Koen Haziël van Dam

Capturing socio-technical systems with agent-based modelling

32

NEXT GENERATION INFRASTRUCTURES FOUNDATION

Capturing socio-technical systems with agent-based modelling

Capturing socio-technical systems with agent-based modelling

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr. ir. J.T. Fokkema,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op 30 oktober 2009 om 10:00 uur
door

Koen Haziël van Dam

doctorandus in de kunstmatige intelligentie
geboren te Delft.

Dit proefschrift is goedgekeurd door de promotor:

Prof. dr. ir. M.P.C. Weijnen

copromotor: Dr. ir. Z. Lukszo

Samenstelling promotiecommissie:

Rector Magnificus	voorzitter
Prof. dr. ir. M.P.C. Weijnen	Technische Universiteit Delft, promotor
Dr. ir. Z. Lukszo	Technische Universiteit Delft, copromotor
Prof. dr. F.M.T. Brazier	Technische Universiteit Delft
Prof. dr. ir. G. Deconinck	Katholieke Universiteit Leuven
Prof. dr. ir. B. De Schutter	Technische Universiteit Delft
Prof. dr. G.V.R. Reklaitis	Purdue University
Dr. ir. R. Srinivasan	National University of Singapore

ISBN 978-90-79787-12-8

Published and distributed by:
Next Generation Infrastructures Foundation
P.O. Box 5015
2600 GA Delft
The Netherlands
T: +31 15 278 2564
F: +31 15 278 2563
E: info@nginfra.nl
I: www.nginfra.nl

This research was funded by the Next Generation Infrastructures Foundation

Keywords: agent-based modelling, socio-technical systems, ontology, benchmarking, oil-refinery supply chain, intermodal freight transport

Cover photography of A. Gormley's Event Horizon by Justin Lynham
Typeset by the author in LaTeX2e in 10pt URW Garamond 8
Printed in the Netherlands by GildePrint on G-print FSC Mixed Sources

FSC
Cert no. CU-COC-811465

e-mail: koen@koenvandam.com
www: http://www.koenvandam.com

Voor mijn zusjes…

Contents

Acknowledgements **xi**

1 Introduction **1**
- 1.1 Problem statement . . . 1
- 1.2 Definitions and scope . . . 2
  - 1.2.1 Infrastructure . . . 2
  - 1.2.2 Complexity . . . 2
  - 1.2.3 Model and simulation . . . 3
  - 1.2.4 Actor and agent . . . 4
  - 1.2.5 Socio-technical systems . . . 4
- 1.3 Examples of new challenges . . . 6
  - 1.3.1 Vertical unbundling of the energy sector . . . 6
  - 1.3.2 High speed rail . . . 7
  - 1.3.3 Carbon capture and storage . . . 7
  - 1.3.4 Commonalities and decision support for socio-technical systems . . . 8
- 1.4 Research objective . . . 9
- 1.5 Audience and relevance . . . 10
  - 1.5.1 Audience . . . 10
  - 1.5.2 Scientific relevance . . . 11
  - 1.5.3 Societal relevance . . . 13
- 1.6 Research questions . . . 14
- 1.7 Overview of this thesis . . . 14
  - 1.7.1 Thesis outline . . . 14
  - 1.7.2 Reader guide . . . 15

2 Modelling socio-technical systems **17**
- 2.1 Introduction . . . 17
- 2.2 Modelling socio-technical systems . . . 17
  - 2.2.1 Research groups and application domains . . . 18
  - 2.2.2 Modelling approaches for socio-technical systems . . . 20
  - 2.2.3 In-depth study of potentially interesting modelling approaches . . . 22
  - 2.2.4 Conclusions part 1 . . . 28
- 2.3 Agent-based modelling . . . 29
  - 2.3.1 Applications . . . 29
  - 2.3.2 Interpretations of the concept “agent” . . . 31
  - 2.3.3 Modelling paradigm spectrum . . . 31

2.4 Agent-based modelling of socio-technical systems . . . 34
2.4.1 Agent-based model for energy systems analysis . . . 34
2.4.2 Multi-agent model predictive control . . . 36
2.4.3 Controlling electricity failures with cooperative agents . . . 37
2.4.4 Agent-based modelling of transport and energy systems . . . 38
2.4.5 BRIDGE agent architecture . . . 39
2.4.6 Modelling the evolution of large-scale socio-technical systems . . . 39
2.4.7 Conclusions on keywords . . . 40
2.4.8 Agent-based approaches in the energy domain . . . 40
2.4.9 Conclusions part 2 . . . 42
2.5 Conclusions . . . 42
2.6 Research questions revisited . . . 44

3 **Framework** 45
3.1 Introduction . . . 45
3.2 Requirements . . . 46
3.3 Ontologies . . . 47
3.3.1 What is an ontology? . . . 48
3.3.2 An example: the girl with a pearl earring . . . 48
3.3.3 Why use an ontology? . . . 49
3.3.4 System decomposition method . . . 50
3.3.5 Software tools for ontology development . . . 50
3.4 Approach to the development of a framework . . . 51
3.4.1 Initial state . . . 52
3.4.2 Iterations . . . 56
3.4.3 Stop condition . . . 57
3.5 Ontology of socio-technical systems . . . 57
3.5.1 Nodes . . . 57
3.5.2 Edges . . . 64
3.5.3 Properties . . . 67
3.5.4 Instances . . . 70
3.5.5 Putting it all together . . . 71
3.6 Modelling steps . . . 72
3.7 Core elements and conclusions . . . 74

4 **Case studies — application and use of the framework** 75
4.1 Introduction . . . 75
4.2 Case 1: Intermodal freight hub . . . 76
4.2.1 Conceptualisation in terms of actors and physical systems . . . 77
4.2.2 Definition of possible disturbances on the system . . . 78
4.2.3 Refinement of the generic ontology with new abstract classes . . . 79
4.2.4 Creation of concrete instances . . . 80
4.2.5 Implementation of the behaviour of the agents . . . 80
4.2.6 Verification and validation of the model . . . 81
4.2.7 Conclusions . . . 81
4.3 Case 2: Oil refinery supply chain . . . 81
4.3.1 Conceptualisation in terms of actors and physical systems . . . 81

4.3.2 Definition of possible disturbances on the system 83
4.3.3 Refinement of the generic ontology with new abstract classes . . . 84
4.3.4 Creation of concrete instances 84
4.3.5 Implementation of the behaviour of the agents 84
4.3.6 Verification and validation of the model 86
4.3.7 Conclusions 86
4.4 Case 3: Chocolate game 87
4.4.1 Conceptualisation in terms of actors and physical systems 87
4.4.2 Definition of possible disturbances on the system 87
4.4.3 Refinement of the generic ontology with new abstract classes . . . 88
4.4.4 Creation of concrete instances 88
4.4.5 Implementation of the behaviour of the agents 88
4.4.6 Verification and validation of the model 89
4.4.7 Conclusions 90
4.5 Cases by others 92
4.5.1 Case 4: Evolution of industrial clusters 92
4.5.2 Case 5: CO2 emission trading 94
4.6 Conclusions 96

5 Framework development trajectory **99**
5.1 Introduction 99
5.2 Analysis tool 100
5.2.1 Design of the tool 101
5.2.2 Data collection 101
5.2.3 Missing values 102
5.2.4 Lessons learnt on the analysis tool 103
5.3 Ontology development 104
5.3.1 Results 104
5.3.2 Conclusions 104
5.4 Re-use of concepts in practice 105
5.4.1 Results 106
5.4.2 Conclusions 107
5.5 Re-use of instances 107
5.5.1 Results 107
5.5.2 Conclusions 109
5.6 Conclusions 111

6 Benchmarking **113**
6.1 Introduction 113
6.2 Benchmarking 114
6.2.1 Definition of the objectives for the study 115
6.2.2 Identification of what is to be benchmarked 115
6.2.3 Evaluation if objects of study are comparable 115
6.2.4 Determination and specification of performance measures 117
6.2.5 Description of scenarios and simulation 118
6.3 Models of the oil refinery supply chain 119
6.3.1 Model E: An equation-based model in Excel 119

6.3.2 Model M: A numerical model in MATLAB 121
6.3.3 Model R: An agent-based model in Repast 122
6.3.4 Mapping the models on the model-space 123
6.4 Benchmarking case study: Oil refinery supply chain 123
6.4.1 Definition of the objectives for the study 123
6.4.2 Identification of what is to be benchmarked 123
6.4.3 Evaluation if objects of study are comparable 124
6.4.4 Determination and specification of performance measures 129
6.4.5 Description of scenarios and simulation 129
6.4.6 Benchmarking conclusions 130
6.5 Conclusions 135

7 Decision support with agent-based models 137
7.1 Introduction 137
7.1.1 Formulate the decision problem 138
7.1.2 Select the decision problem solving method 139
7.1.3 Perform experiments 140
7.1.4 Analyse the results 140
7.2 Decision support for the location of an intermodal freight hub 141
7.2.1 Formulate the decision problem 142
7.2.2 Select the decision problem solving method 143
7.2.3 Perform experiments 143
7.2.4 Analyse the results 144
7.2.5 Conclusions 145
7.3 Decision support for abnormal situation management in a refinery 146
7.3.1 Formulate the decision problem 146
7.3.2 Select the decision problem solving method 148
7.3.3 Perform experiments 149
7.3.4 Analyse the results 149
7.3.5 Conclusions 150
7.4 Conclusions 151

8 Conclusions, discussion and future research 153
8.1 Conclusions 153
8.1.1 Modelling framework for socio-technical systems 154
8.1.2 Categories of modelling paradigms 155
8.1.3 Benchmarking modelling paradigms 155
8.1.4 Advantages of agent-based modelling 155
8.1.5 Ontology development 156
8.1.6 Decision support 156
8.2 Discussion 156
8.2.1 The best approach? 156
8.2.2 Scope 157
8.2.3 Benchmarking 157
8.2.4 Ontology languages 157
8.2.5 Distributed controllers 158
8.2.6 Application in the industry 158

8.3 Recommendations for future research 159
8.4 Final remark 160

Appendices 161

A Literature study on socio-technical modelling 163
A.1 Literature study approach 163
A.2 Application domains and background of authors 163
A.2.1 Completeness of study 166
A.3 Modelling approaches for socio-technical systems 166
A.3.1 Literature review 166
A.3.2 Selection of papers 168
A.4 Agent-based modelling in the energy domain 169
A.4.1 Literature review 169

B Survey on agent-based systems 173
B.1 Introduction 173
B.2 Questionnaire 173
B.3 List of answers 173
B.4 List of participants 179

C Models built with the framework 181
C.1 Brief description of models 181
C.2 Time planning 183

D Classes added to the ontology 185
D.1 List of classes added 185

Bibliography 188

Glossary 209

Summary 211

Samenvatting 217

Curriculum vitae 223

List of publications 225

NGInfra PhD thesis series on infrastructures 229

List of Tables

2.1 Review of modelling approaches for socio-technical systems 22
2.2 Attributes of modelling 32

4.1 Agents and PhysicalNodes for the freight hub case 77
4.2 Agents and PhysicalNodes for the refinery supply chain case 82
4.3 Agents and PhysicalNodes for the chocolate game case 88

5.1 Time periods and number of active projects 109

6.1 Steps required for the extension of models for six scenarios 131
6.2 Estimated efforts to implement six scenarios in Model M and Model R . . 134

7.1 Outcome of the Nelder-Mead simplex optimization 149

A.1 Application domains and background of authors 163
A.2 Detailed review of modelling approaches for socio-technical systems 167
A.3 Agent-based models in the energy field 169

B.1 Answers to the survey on agent-based systems 174

D.1 Classed added to, or removed from, the shared knowledge base 185

List of Figures

1.1 Interaction between physical and social networks 4
1.2 Two main story lines in this thesis 16

2.1 The modelling space 33

3.1 Approach of the framework development 52
3.2 The border between generic and case-specific classes 53
3.3 The relationship between different classes of Nodes 58
3.4 Nodes in the ontology 59
3.5 Agent in the ontology as a subclass of SocialNode 60
3.6 Technology in the ontology as a subclass of PhysicalNode 61
3.7 OperationalConfigurations for the class Technology 62
3.8 Edges in the ontology 63
3.9 Physical edges in the ontology 65
3.10 Contracts in the ontology 66
3.11 Ownership in the ontology 67
3.12 Application cycle for re-use of ontologies and generic building blocks . . . 73

4.1 Candidate sites for a new intermodal freight hub in Queensland, Australia 78
4.2 Geographical representation of the nodes in the freight hub model 79
4.3 Schematic of an oil refinery supply chain 83
4.4 Screenshot of the graphical user interface of Protégé 85
4.5 Students negotiating about trading goods in the chocolate production game 90
4.6 Screenshots of the agent-based implementations of the chocolate game . . 91

5.1 Growth of the ontology in number of classes and slots over time 105
5.2 Instances added to the shared knowledge base and projects depending on it 108
5.3 The growth of classes and growth of instances 110

6.1 Simplified schematic of refinery supply chain as used in Model E 120
6.2 Crude tank masked block in Model M 122
6.3 The three models presented in this chapter plotted on the modelling space 124
6.4 Results from Case 1: no stochastics 126
6.5 Results from Case 2: no stochastics with a step in demand 127
6.6 Results from Case 3: with stochastics 128

7.1 Illustrative results from the intermodal freight hub case study 144
7.2 Assets of the transport company agent as a function of operational costs . 145
7.3 Results from the oil refinery supply chain case study 150

C.1 Time planning for cases that use the shared ontology 183

Acknowledgements

The work presented in this thesis is the result of four years of research as a doctoral candidate at the Energy and Industry group in the faculty of Technology, Policy and Management of the Delft University of Technology under the supervision of prof. dr. ir. Margot Weijnen and dr. ir. Zofia Lukszo. The life of a doctoral candidate is often thought of as being solitary, but I have had the pleasure of working with, learning from, and just being around many people throughout my research project. I am indebted to all of you, but I would like to thank some people in particular.

First of all, Margot, thank you for your dedicated guidance during my PhD research. Your ability to form, lead and encourage a devoted team of researchers from so many backgrounds, as well as your ambition to address both the scientific and the political agenda's have been a strong motivation to start my research in your group. I have really enjoyed our discussions as they were getting more and more frequent over the years and I learned so much from working closely with you these last months. I am more than thankful for everything you taught me.

Zofia, our long and frequent meetings about definitions and formalisations have really changed my way of thinking and I have fond memories of all the times we sat at the table with several heavy books open, trying to find that one perfect word. When you'd sometimes receive a phone call from home during our meetings, I would go back to a slogan you have on the wall: "Most people don't know that there are angels whose only job is to make sure you don't get too comfortable and fall asleep & miss your life". Well, I don't know if it was those angels or all the work you sometimes gave me, but I sure didn't get the chance to fall asleep. And it has been more than rewarding. I will always be grateful that you have inspired me to stay in Delft to pursue a PhD. Thanks to your warm and involved way of supervising I have never regretted that decision. Thank you for everything.

I owe special thanks to the members of my thesis committee for all the time taken to read my work and to patiently provide me with such detailed feedback. Your input has been significant and it has helped me to, hopefully, tie up the last loose ends. I really look forward to continuing our discussions during the defence.

My dear colleagues at E & I. Working with you I have felt at home right from the start. The social cohesion within the group is truly unique and combined with brilliant minds and a strong drive we've got a winning team. You have all inspired me, contributed to my thinking and helped advance my knowledge. I would like to especially thank all the people I shared an office with for helping create a great atmosphere. We never stay in one configuration for a very long time, so over time I've had the pleasure of working in an office with Angelique, Ivo, Laurens, Igor, Chris, Reinier, Alfredas and Amineh, as well as several MSc students, including Robert and Maarten.

Hello E & I agent people! I am grateful for your contributions to the framework presented in this thesis and I fully acknowledge that the final result would not have been possible without team work. I hope a close cooperation has made us all stronger. It has been fun to learn about all the exciting things you are doing, too. Seeing how work could be shared and re-used in practice has been very gratifying. Discussing practical implications with Rob, Gerard and others helped put things in the right perspective.

Michiel, the swimming sessions during lunch breaks were the best way to clear the head (Jeroen, shame you could not join us more often!) and seeing bands from AC/DC to zZz live on stage in the weekends restored the energy levels. Thanks for your friendship. Emile, you set an example when it comes to clean and clear coding and I am glad we share an interest in nice typography and attention to detail. Oh, and then there was Detroit, of course! Igor, with your enthusiasm for complexity, wiki and open-source you change the way people work together and that is a rare and valuable skill. I really enjoyed working with you. Anish, thanks for all the social events you have organised and keep up the hard work, you're almost there. Austine, I'm glad you are doing well out there but I miss you man, the couple of conferences we attended together were the best. Behzad and Layla, all the best for your research and I apologise for having taken so much of Zofia's time these last months. Chris, these things you are doing with open data are really cool and I am curious where it will all go next. Andreas, I've enjoyed the discussions on peak oil and I am very keen to find out how you will proceed with the non-rational agents. Alfredas and Amineh, it is so nice to welcome you, as well as the other new colleagues, to the section. Of course I want to wish you good luck with the projects you are only just starting with, but I should also warn you that four years will be over before you know it so please make sure you enjoy the ride.

Connie, Angelique, Rachel and Inge at the secretariat, thank you thank you thank you, I cannot say this often enough…

My research project was part of the Intelligent Infrastructures subprogramme of the Next Generation Infrastructures Foundation and our regular meetings have been a perfect opportunity to learn about different perspectives, problems and various modelling tools for infrastructures. I would like to thank all other members of the programme, including Hans, Bart, Geert, Jos, Michel, Michiel, Petra and Rudy. Rudy, I also want to thank you for your help with LaTeX and please know that I am always impressed by your structured and efficient way of working.

For the freight hub case study I have worked closely with prof. Luis Ferreira from Queensland University of Technology. Luis, your vast experience in transport modelling as well as your work for the government have been a great inspiration and I would like to thank you for the pleasant cooperation. I would also like to thank the National University of Singapore and prof. Raj Srinivasan for inviting me to spend four months at the Department of Chemical and Biomolecular Engineering to work on the oil refinery case study. Raj, the countless discussions on various modelling paradigms we had in Singapore (and later on a regular basis in London) have really moved my work forward and resulted in some of the key findings presented in this thesis. It was a real pleasure to discuss modelling as well as life in Asia with the members of the lab for Intelligent Applications in Chemical Engineering, including Arief, Kaushik, Iftekhar and Sathish. Arief, I am especially grateful to you for for all the hours during which we compared the most specific details of our models.

I am very happy I have been given the opportunity to develop myself outside of my research work as well. Working closely with highly motivated people in organisations representing young researchers at the local, national and international level has been a great and very rewarding experience. Starting at Promood in Delft, where Ides, Gary, Tuba, Loesje, Rudy and many others made my first active membership of an organisation a real pleasure and encouraged me to continue in this direction. At the national level, at PNN, I have been lucky to have been given the chance to work with Derek-Jan, Renée, Marijke, Leonie, Caelesta, Gertjan, Marijn, Michel, Olof, and so many others who were a member of the board or representative of a local organisation. Eddy, I would like to thank you in particular; you have been an inspiration for me with your strong drive to continue research even when there are plenty of obstacles and with your ability to turn them into the most entertaining stories. Thanks also for your support when I joined Eurodoc and for being there when I needed advice. Within Eurodoc I have really enjoyed working with people from all over Europe and I could fill a couple of pages acknowledging every single one of you, but I just want to say a special thank you to Raquel, Dunja, Armando, Rhys, Dmitri and Judit who did a great job in the board for 2007/2008, to Karoline and her board for taking over the responsibility and to the people in Vienna I want to say good luck with Eurodoc 2010! Working in all these organisations has enriched my life as a young researcher and I am proud of everything we have done together.

To all my friends I want to say this: thanks for being amazing! Martijn, our trips to festivals and gigs around Europe are always so much fun and they were the perfect mini-breaks away from my PhD. Rogier and Nicole, I really value our long-standing friendship, thanks for your support and I'm so happy about the arrival of Amélie! Dear Ides, your encouragement over these last four years has been a tremendous boost and I want to thank you for always believing in me. Marjolein, going to the beach, dancing and enjoying a nice meal with a friend, those are some of the best things in life. Dennis, I'm really looking forward to being able to attend concerts in your living room (and I also mean WATT) again. Claire and Helen, thank you both for our friendship in Singapore! Justin and Stavroula, so kind of you to put me up in London all the time and thanks for this not being needed any more, hehe. I am also very pleased you have donated one of your beautiful pictures for the cover of this thesis.

Lieve Mama, Papa, Justien & Fleur. Het boek is klaar! Ik ben ontzettend dankbaar dat jullie mij al deze jaren zo hebben gesteund, aangemoedigd en geïnspireerd. De continue interesse in de voortgang van mijn werk was enorm motiverend, de telefoontjes, e-mailtjes en smsjes kwamen meestal precies op het goede moment, en mijn bezoekjes aan Almere waren de ideale gelegenheid om weer even op te laden. Thuis is nooit ver.

I would also like to express my sincere gratitude to the Khan family. It means a lot to me that you will all try to attend the viva and of course I really look forward to welcoming you here in Delft.

Rogier and Ides, I am thrilled you have both accepted to be my paranymphs for the defence ceremony. Together we will make it a very special day.

Lovely Rebecca, my thesis is ready: let's celebrate!

Koen Haziël van Dam
Delft, September 2009

Chapter 1

Introduction

1.1 Problem statement

Decision-makers in the *infrastructure* domain, which includes the electricity sector, transport networks, industrial clusters and supply chains, have to deal with capacity limitations, unexpected disruptions, maintenance and investment decisions, as well as other challenges. These problems have always existed (e.g. competition in the developments of the first transcontinental railways in the 1860s in the USA ("Pacific Railroad Act"), capacity problems of dung removal for horse trams in London, UK, until World War I, and the 1965 power blackout in the North-East of the USA and Canada), but they are still difficult to solve, understand and predict due to the *complexity* of the infrastructure systems.

This complexity arises from a world that is more and more connected: infrastructure systems are not independent of each other but have significant dependencies and interactions[1]. Different infrastructures need each other, for example electricity networks and telecommunication networks; one cannot function without the other. Moreover, these systems have grown from small, often local, systems to regional, national, continental and global networks. They were not designed to function like they do today, but evolved to this state.

Furthermore, infrastructure systems are *socio-technical systems*: not only the physical system is complex, but so is the social network to which it is inherently connected. The social network includes *actors* such as the users, network operators, maintenance companies, governmental authorities and regulators. These actors are part of a bigger system and they call for novel solutions to approach the challenges of socio-technical systems.

Decision makers often rely on *models and simulations* for support in the decision process to come to well-informed conclusions. Model-makers design and build models that can be used to test different scenarios and to gain insight in the possible consequences and results of many actions, using simulations. These models can be used for *decision support*. What is a suitable modelling approach for socio-technical systems? The answer to this question is of great importance to strategic, tactical and operational decision makers in large-scale interconnected network systems.

[1] Although, one could argue that this is not new (e.g. the strong link between railways and telegraph systems in the 19th century), such interaction has effects on a much larger scale nowadays.

Challenges for the development of models arise when trying to incorporate both the technical and the social systems in one model. Existing tools to deal with either the physical (e.g. models of industrial processes) or the social network (e.g. economic market models) are available, but these worlds have yet to be brought together in an integrated modelling approach for socio-technical systems. That is the ambition of this thesis.

1.2 Definitions and scope

In the problem statement some key concepts have been used that need to be well specified and defined for use throughout the thesis, before continuing with a number of specific examples of challenges and the research goals.

1.2.1 Infrastructure

The word 'infrastructure' is widely and commonly used in the English language, but still it is open to different interpretations. One would easily agree that a road system of motorways and carriageways is an infrastructure, as is the network that brings electricity from power plants to end users, but how about the stock exchange or an educational system of schools and universities?

A start is to look at a dictionary definition: "*Infrastructure* (noun): the basic physical and organisational structures (e.g. buildings, roads, power supplies) needed for the operation of a society or enterprise" (*Oxford English Dictionary Online* 2009). Interesting enough, this definition already highlights the socio-technical nature of infrastructures: it does not only encompass the physical structures, but also the organisational structures. What makes something an infrastructure lies in the purpose of the system: without it society (or, at a lower level of hierarchy, a company) could not function. Recently the effects of a failing financial sector on society have again become visible[2], so indeed it is an infrastructure. The scope of the concept 'infrastructure' is, therefore, very broad.

For this thesis, however, a sub-set of infrastructures is considered, namely those systems in which mass, energy or information is literally *transported* through a physical network and *transformed* in the nodes. It is an engineered system and the organisational structure is in place to support this transfer or directly use it.

1.2.2 Complexity

Herbert A. Simon, winner of the Nobel Prize in Economics and one of the founders of the field of Artificial Intelligence, already in the year 1962 refused to give a definition of a *complex system*. He realised there are many definitions and many different fields in which the concept of complexity is used. Instead of formulating a definition, he said: "Roughly, by a complex system I mean one made up of a large number of parts that interact in a non-simple way. In such systems, the whole is more than the sum of the parts, not in an ultimate, metaphysical sense, but in the important pragmatic sense that, given the properties of the parts and the laws of their interaction, it is not a trivial matter to infer the properties of the whole. In the face of complexity, an in-principle reductionist may be at the same time a pragmatic holist" (Simon 1962). He then stresses that the concept

[2] See for example Zandi (2008).

of hierarchy (meaning that a system is composed of interrelated subsystems that each in turn can also be hierarchic) is one of the structural schemes in complex systems. This property can be observed in the infrastructure systems discussed in this thesis.

Another complex systems researcher who also goes by the name Simon, is more specific and defines a complex system as a system that has certain well-defined properties. A complex system is a system that has many components that are heterogeneous (i.e. many different types of components), have non-stationary, non-linear dynamics, contains feedback loops (i.e. the output of a component is input to another component), is organised and nested (i.e. contains hierarchies and subsystems which themselves can again be seen as complex systems) and shows emergence (i.e. the behaviour of the system cannot be predicted by looking at the behaviour of the lower level components) (Simon 2006). The socio-technical systems that are the topic of this thesis all have these characteristics.

There are numerous others who try to give a definition of complexity and complex systems, such as Mikulecky (2001) ("Complexity is the property of a real world system that is manifest in the inability of any one formalism being adequate to capture all its properties") and Holland[3] ("[A complex adaptive system is] a dynamic network of many agents (which may represent cells, species, individuals, firms, nations) acting in parallel, constantly acting and reacting to what the other agents are doing. [...] The overall behaviour of the system is the result of a huge number of decisions made every moment by many individual agents" in Waldorp (1992)). While here it is acknowledged that there are many definitions of complexity, it is not necessary to choose a specific one, let alone try to add a new definition: socio-technical infrastructures are considered as complex systems under these different definitions and the bottom-line is that models need to be able to capture these characteristics in order to be useful.

Note that complexity in the way it is used in this thesis is different from *computational complexity* which deals with intrinsic limitations of what can and what cannot be efficiently computed given limited space and time (Borodin 1975). The challenge for this thesis lies in capturing the complexity in the world and not in efficient use of computer power.

1.2.3 Model and simulation

A model is a simplification of reality, designed to learn something about reality. In building a model, choices have to be made as to what is important and to what extent it can be understood and simplified. Here specifically *computational* models are considered: those models that can be implemented in a computer program so that calculations can be made using it. Simulation is then "the activity of carrying out goal directed experiments with a computer program. A distinctive aspect of this program (which is typically referred to as a simulation model) is that it has been developed to capture relevant features of the dynamic behaviour of some 'target system' which is under study" (Birta & Özmizrak 1996). These experiments always have a purpose, for example to optimise a system that is being studied or to gain insight in how the system behaves and responds.

[3] Associated with the Santa Fe Institute which is dedicated to studying complexity theory.

1.2.4 Actor and agent

In the definition of a complex adaptive system by Holland in Waldorp (1992), the term *agent* is used and this concept will play an important role throughout this thesis. A strict distinction between the concept of an *agent* and an *actor* is drawn: an actor is an active entity (be it an individual or a collective) in the real world that makes decisions whereas an *agent* is a *model* of an actor. Furthermore, the concept of an *agent* has a specific meaning in the agent-based modelling paradigm (see Section 2.3). While actors, being human or organisations consisting of humans, can behave irrationally (meaning that even when a certain decision has a known positive effect towards a goal of the person, he or she is not guaranteed to make this choice) their modelled counterparts, the agents, are in this thesis assumed to be *rational*.

1.2.5 Socio-technical systems

A *socio-technical system* consists of one or more social networks and one or more physical networks that interact with each other (See Figure 1.1). One could consider them as different networks where one follows social laws (e.g. legislation, unwritten codes of behaviour, economic contracts) and the other follows the physical laws (e.g. Newton's laws, Archimedes' principle, Einstein's theory of relativity). In a socio-technical system both types of laws influence the system (Ottens, Franssen, Kroes & van de Poel 2006).

In a similar fashion, but coming from a different perspective, the technical system can be considered as a problem-solving system, usually concerned with the reordering of the material world. It is "a means to an end" (Hughes 1987). However, perhaps the social network should also be considered as a means to an end? Can it be designed like one would design a technical system or are other approaches necessary?

Hughes, in his frequently cited work 'The evolution of large technological systems' (1987), never uses the word socio-technical system, but instead uses a different word for this: *technological systems*. Technological systems are "socially constructed and society shaping systems" and consists of

- Physical artefacts;
- Organisations;

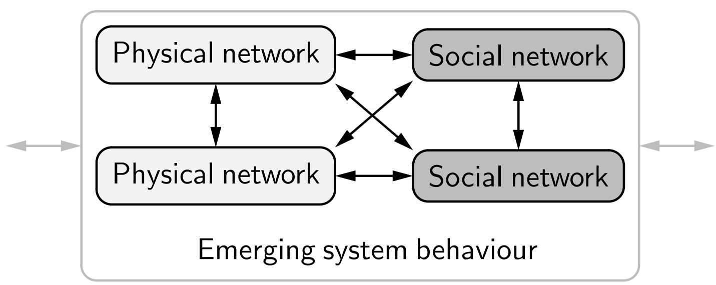


Figure 1.1 – *Interaction between physical and social networks*

- Scientific components and
- Legislative artefacts.

Both the social and physical artefacts are designed, and this is often done by different people. Engineers usually limit technological systems to technical components, with the mistaken impression that system-growth and management are neatly circumscribed and they consider "politics" to be separate. However, technological design is part of the system that inherently includes the social organisations, too. One can say that in a technological system organisational form follows technical function, but also that technical function follows organisational form (Hughes 1987). The term socio-technical system, however, captures this better than the term technological system (Weijnen & Bouwmans 2006).

The socio-technical concept is used in many different research fields. The top five disciplines (based on number of papers as found in Scopus[4]) using the term socio-technical[5] are

- Engineering;
- Social Sciences;
- Computer Science;
- Business, Management and Accounting and
- Medicine,

with 'arts and humanities' and 'agricultural and biological sciences' at the bottom of the list. Furthermore, 'chemical engineering' and 'energy' (two fields that are addressed in this thesis) score low[6], for example. In each of these five disciplines listed above slightly different definitions are used:

Engineering: In a technological system, organisational form follows technical function, but technical function also follows organisational form (Hughes 1987).

Social science: In the social sciences a common view of socio-technical systems is that of humans operating in a technical world: systems that comprise of the interdependencies between persons especially the mutually dependent activities of multiple persons (those dependencies include social aspects like communication and cooperation structures, formal organisational structures, personal expectations and interests or qualifications) and also have a technical side where artefacts are relevant (Herrmann & Loser 1999).

Computer science: In computer science the technical system consists of the hardware and software that make an information system, while the users of this system and the organisation in which it is embedded form the social system. The main challenge lies in the specification of requirements (Sutcliffe, Chang & Neville 2007) as well as human-computer interaction.

[4]http://www.scopus.com/

[5]Including variations thereof, such as social-technical and socio-technological.

[6]Note that these fields are in general smaller than those in the top five (and could even be considered as part of 'engineering'), which could explain the low number of published papers with this keyword.

Business, Management and Accounting: In business, management and accounting, a socio-technical approach means observation of work group configurations and different ways of supervision, combined with informal discussion with workers so that knowledge is collected not only about the mechanical (or technical) aspects of a job, but also of the social aspects (Trist 1981). These findings can then be applied to real organisations and used in consultancy (Pasmore & Khalsa 1993).

Medicine: In medicine the concept of socio-technical system is mostly used in relation with *telemedicine*, which is providing healthcare services on a distance through information and communication technology (LeRouge, Hevner, Collins, Garfield & Law 2004). The social aspects typically consist of patients and healthcare providers and the technical aspect are the computer systems used.

These definitions all choose a different focus point, but in essence they are not much different. The challenges that arise from the interaction between physical and social elements (regardless of *what* is considered to be part of those networks) are comparable. Following the choice for a sub-set of infrastructure systems in Section 1.2.1, the view of socio-technical systems used in this thesis is that of a system that includes both social and technical elements that can both be considered as nodes in a network. Specifically, the social nodes make *decisions* about the physical nodes (which they own, control, manage, etc.) and the physical nodes *convert* mass, energy or information. This view is based on tasks and functions and it is a subclass of the systems used in the research fields mentioned above.

1.3 Examples of new challenges

Next, a number of recent cases are presented briefly to illustrate the wide scope of challenges that have to be dealt with, and to identify common aspects which a new approach has to be able to cope with.

1.3.1 Vertical unbundling of the energy sector

The energy sector has radically changed over the past years. Driven by liberal beliefs in the power of the market to come up with the best solutions to problems and to make systems the most efficient and at the lowest price, competition has been introduced. Traditionally the energy sector has been one dominated by monopolies, often state-owned. The main infrastructure, the physical transportation network for electricity consisting of the high voltage lines (the *grid*) is a natural monopoly (i.e. it is not efficient to build a second power grid) so here competition is not suitable. However, the companies responsible for generation of power and sales to consumers were also responsible for the transportation and network operation: the whole chain was vertically integrated. This meant there was no real market.

To encourage more competition, mainly in electricity production and sales, many governments in Europe (following policies from the European Commission) decided to start the process of *vertical unbundling*, meaning that companies are not allowed to be active in different layers of the chain at the same time. This required a complete restructuring of the energy sector and reorganisation of companies, as well as the installation of a regulator.

While the social and economical system completely changed, the physical energy infrastructure has remained the same: the same cables are in use, the same power plants and transformers are active, etc. The new electricity sector has the same physical network, but a different social network. To make models of this process it is necessary to have the flexibility to make changes in the social network, while keeping the physical elements the same.

1.3.2 High speed rail

A new high speed train connection between the Netherlands and Belgium ("Hogesnelheidslijn" or "HSL-Zuid" in Dutch) has been developed between 2000 and 2006, with the aim to create a faster connection between Amsterdam, Schiphol Airport, Rotterdam and Antwerp, reducing not only international but also local travel time. Plans for this new rail connection were already made in the 1970s, but among other reasons because of the high investments and risks attached to the project it was postponed, until it was finally realised using a *private public partnership* (PPP) construction in which the parties executing the work will be responsible for design but also long term maintenance and they are involved in the exploitation, for which they receive funding from the government.

Vertical unbundling is a reality in the Dutch railway sector too (meaning that an actor is responsible for operation and maintenance of the tracks but several other actors can operate trains on these tracks) so there are many different actors involved. For the new high speed rail this includes a new consortium of (already existing) players which together accepted the PPP project, infra-management organisations and the government. Also for the exploitation of the line a new alliance was formed between an existing railway operator and an airline.

On the trajectory some existing tracks are also used (for example between Amsterdam and Schiphol Airport) which adds extra complexity. On top of that, a new security system (European Train Control System, or ETCS) was installed which is hoped to become a European standard, but has not been widely used yet[7]. Furthermore, the trains that were ordered were not ready in time. The consequence is that slower trains still operate on the old tracks, even though the new infrastructure is ready.

Whereas in the example of the vertical unbundling of the electricity sector the physical system stayed the same and only changes in the social system were made, here the changes are mostly in the physical network: new rail routes and a new security system have been installed which are connected to the old physical system, while the same actors are involved. A model of such a system should be able to connect with already existing elements, be flexible in the physical layer to allow the new network elements to be included and should allow existing actors to play the same role, as well as include other responsibilities to be shared between actors.

1.3.3 Carbon capture and storage

As a third example let us consider *carbon capture and storage* (CCS). This is a technique considered as a possible solution for minimising the amount of carbon emissions into the

[7]The ETCS system has multiple security levels, but the ETCS Level 2 installed in this case relies on a digital radio-based system and no longer uses track-side signaling, so trains that do not support the security system cannot be used on the tracks.

atmosphere. The capture technique, for example installed in a fossil fuel power plant, filters (part of) the CO_2 that is produced. This CO_2 can then be stored, for example in empty natural gas fields, so it does not end up in the atmosphere with all negative effects that are linked to this, such as climate change.

To install CCS, a number of investments have to be made in the physical system. Capture devices are only part of it, but more importantly a new transport infrastructure has to be created together with storage facilities. To manage this new infrastructure, it is conceivable that a new social infrastructure has to be designed to regulate this new market.

A part of the actor network remains the same (e.g. energy producers), but new technologies and a new infrastructure have to be added that are connected to an already existing infrastructure (e.g. power plants of said energy producers). Also, new actors are needed for this new infrastructure, with new responsibilities and actions. A model, for example to experiment with the feasibility of CSS or compare alternative designs, has to cope with these challenges. It should connect to existing models of, for example, the energy sector or the petrochemical industry. This is the case for both the physical and the social network.

1.3.4 Commonalities and decision support for socio-technical systems

The cases presented in this section have one key element in common: certain aspects of the system remain the same, while others radically change or completely new elements are introduced. Sometimes the social network changes while the physical layout of the network remains the same while in other cases it is the physical network that changes for a given social network. The last examples showed that elements of both the social and physical network are altered, while other parts of both networks stay the same after the introduction of new actors or physical elements.

Challenges in the infrastructure domain as considered in this thesis are characterised by the fact that they are multi-actor, multi-criteria and multi-level problems. This means that there are multiple stakeholders with their own goals (which may or may not be conflicting with those of other stakeholders), who have multiple objectives and values (which may or may not be conflicting) and who may operate at different levels of hierarchy. These characteristics make it hard for actors to take the well-informed decisions, but models can support them in the decision-making process.

It should be stressed that there is not just one stakeholder in large scale socio-technical systems. With many different actors cooperating or in competition with each other, there will be different interests. The fact that there are multiple actors in the system is one of the characteristics of socio-technical systems. For a decision support tool, however, there is usually only one problem owner for whom the tool is designed. Models of other actors should be included in the system model, but it is assumed that only one is responsible for the assignment to create a model and from this perspective the problem owner is unique. In this thesis different roles of problem owners being supported by models are included, such as governments with a supervisory role or the management of a company.

In general in many of today's infrastructure systems, it is not possible for any one problem owner to directly influence the whole system. For the model to be an effective support tool for the problem owner, it needs to give insight in exactly how changes at lower levels impact the emerging system behaviour. This way of modelling is close to how it works in the real world: the collective decisions made by more or less autonomous actors at various levels of a hierarchy together result in an overall system behaviour.

To help decision makers with a decision support tool, models have to deal with the type of changes that occur (or are being considered) in the real system. The result of this requirement is that social and physical aspects have to be described separately and the model should allow changing them in a modular fashion. Furthermore, new elements in the model should be able to connect to existing parts, just like in the real system new additions to an infrastructure connect to already existing ones.

The goal of decision support models is not always to find a system *optimum* (which is a common goal for many existing models) or *predict* the future. Epstein (2008) lists "sixteen reasons other than prediction to build models". Models can, for example, be used to improve *understanding* of the dynamics of the whole system and subsystems, explore possible futures, find states that have to be avoided or that are desirable, and, most of all, to provide a tool for decision makers to experiment with "what-if" scenarios, etc. Which degrees of freedom are there? What are the possible consequences of certain decisions? What are successful configurations of either physical or social networks? It should be stressed that this requires a wider view than traditional engineering: the systems under research are considered as part of a larger system; the view is one of a *system of systems* (Hansman, Magee, Neufville, Robins & Roos 2006).

1.4 Research objective

The objective is to develop an integrated modelling approach for socio-technical infrastructure systems that can "capture" both the physical and social reality of the system, their interactions with one another and the external dynamic environment. The additional challenge is to meet this objective not just for one specific domain, such as energy or industry, but to develop up a modelling framework that is able to deal with today's reality of socio-technical network systems that are interconnected across domains.

From the examples from Section 1.3 it can be deduced that a successful model has to be able to deal with

- different configurations of the social network with same physical network;
- different configurations of the physical network with same social network and
- different configurations of *both* social and physical networks.

Furthermore, the models should be able to connect new parts of the model with existing elements. This is the case both when making extensions of models and when dealing with the interactions between infrastructures, meaning that (elements of) models of infrastructures have to be connected.

The first step is to analyse what existing models do, how they are designed and implemented and what their aims are. Can they cope with the challenges stated in Section 1.1? And if not, the second step is to find what needs to be done to make models that *can* deal with these challenges. How can existing models and approaches be improved? A new framework to modelling socio-technical systems can be developed and applied, to help modellers build better models[8] and ultimately provide better decision support to actors involved in regulating, operating or otherwise using these systems.

1.5 Audience and relevance

The target audience and key problem owner(s) are discussed in Section 1.5.1. Furthermore, the relevance of this thesis, both from a scientific viewpoint (Section 1.5.2) and the viewpoint of society (Section 1.5.3) is addressed in this section.

1.5.1 Audience

There are two different "problem owners" who form the main target audience for this thesis. Where the "decision maker" is the real problem owner of problems such as listed in Section 1.1, the modeller has to cope with the challenge of how to build a model of the system. This thesis is mainly targeted at this second group: people who build models of socio-technical systems. However, decision makers might also find interest in this research work because it shows how lessons learnt from one domain can be translated to another domain and it can support them in defining the model requirements. Furthermore, others working on agent-based modelling, ontologies or re-usable software could benefit from this thesis. Sections 1.5.1.1, 1.5.1.2 and 1.5.1.3 address these three groups.

1.5.1.1 Modeller

Modellers are the main target audience of this thesis. Their work has become more difficult with new requirements to cope with the socio-technical complexity. To meet the requirements of their assignments, *better models* are needed. Models that can deal with social as well as technical components and that can describe how they interact. Models that are flexible enough to perform experiments in which parts of the system, either only social or only physical or both, change while others stay the same. Models that can easily be connected with other simulation models that include some of the important interdependencies that can be observed in the real world and that can be the cause of extra complexity. In other words, models that satisfy the needs of the decision maker.

When on an assignment to build a model, one has to make a number of decisions, such as:

- What is the best modelling paradigm for the problem?
- How can the use of the paradigm be justified, compared to other options?
- Does this paradigm meet the requirements from the assignment?
- What is the quickest way to build new models to test different decisions?

These questions have to be answered before a model can be created. This thesis has the aim to help answering them. Moreover, a modelling framework is presented that can help a modeller to quickly set up new models by re-using building blocks. Following this framework, the models are flexible to perform simulations with variations in the

[8]Note the emphasis on building better *models*, which is not necessarily the same as getting better (e.g. more reliable) *results* from models.

system structure (both social and physical) and to connect it to other models of other infrastructure systems.

Following the approach presented in this thesis does not lead per se to better predictions, more accurate results, or more valid simulations, but it does mean that models can be built that meet the requirements from the increased complexity of infrastructure systems. In other words, the thesis can help building better models.

1.5.1.2 Decision maker

Decision makers, such as policy makers, regulators, infrastructure managers, investors, designers, planners, contractors, service providers and operators, have to be aware of the opportunities offered by models and simulations as a decision support tool. When writing an assignment for the development of such a decision support tool one should not be held back by limitations to model complexity that may have been the case in the past. It is important to stay up-to-date on the latest developments, even when not executing the modelling work. This thesis assists the decision maker to hire a modeller with the appropriate skills for the problem and to ask the right questions about the model. Furthermore, reading about the possibilities can open up new ideas for applications.

Furthermore, the approach developed in this thesis makes it possible to learn from other applications in other domains. An important aspect of the framework is a shared language. This is not only a language between the elements in the model, but is also suitable for people in different disciplines to talk to each other as well as for decision makers to talk to modellers. If concepts can be expressed in this language, it should be clearer to others what is meant and common grounds between problems in different domains can be discovered so that solutions can be found jointly.

1.5.1.3 Others

In addition to these two main audience groups others, such as software engineers or ontology developers, may be interested in ontology development in general, the use of ontologies in agent-based models or re-use of source code in software engineering. Ontologies play an important role in many software systems, knowledge-based systems or for example in the *semantic web*. The ontology presented in this thesis can also be used in other types of applications, outside the agent-based modelling framework.

1.5.2 Scientific relevance

As a PhD thesis, this work aims to advance and contribute to the realm of science. There are four different fields of science where this thesis hopes to make an impact:

Technology, policy and management The field of Technology, Policy and Management deals with the difficult link between engineered physical systems and the policies that relate to this engineered system as well as the management of organisations that use and depend on such technical systems. The view of the world is one of multi-actor systems. This research work aims to contribute by offering an approach to create models where these two processes (physical and social) converge.

Knowledge engineering Artificial intelligence is a broad research field with a confusing name that is open to different interpretations. Here the sub-field of *knowledge engi-*

neering is considered only. In this field it is the aim to "understand" and "capture" human intelligence, learn from it and use it to create so called *knowledge based systems*. In this thesis agent-based modelling, a widely used approach from the field of artificial intelligence, is applied to capture the socio-technical system. Furthermore, knowledge acquisition and ontologies, both concepts from the knowledge engineering discipline, are used. No fundamentally new developments are made that contribute to this scientific domain, but this thesis applies an ontology in a novel way to setup agent-based models. By combining existing knowledge it contributes to this field of research.

Computer engineering Computer science looks at the development of computer hardware and software and their interaction. Re-use of software and modularity receive a lot of attention in this field. Again, existing knowledge is combined to create something new, but moreover new insights into how computational modelling paradigms differ from each other are presented. Finally, a structured approach for benchmarking paradigms is proposed.

Process systems engineering The process systems engineering field, traditionally concerned with (chemical) production processes and manufacturing, now aims at the integration of system elements (Grossmann & Westerberg 2000), developing a multi-disciplinary approach (Gani & Grossmann 2007) and acknowledges the need for systems thinking and shows a growing interest for applications in the infrastructure domain (Klatt & Marquardt 2009). The focus in models of process systems is, however, mostly on the physical aspects and the social layer is often ignored. In this thesis it is stated that the social elements are critical and they also need to be included in models. Doing so creates more opportunities for performing experiments with these models. Through different applications in the process systems engineering domain, lessons are learnt about the use of this approach (and agent-based modelling in general) that could be of benefit to this field of science.

The main scientific contributions of this thesis are the following:

Framework A modelling framework for agent-based models that can be re-used in different infrastructure domains.

Ontology An extensible ontology for the domain of socio-technical systems.

Benchmarking approach A structured and well-defined benchmarking approach offering techniques to compare different modelling approaches and modelling paradigms.

Categories of modelling paradigms An approach to the categorisation of modelling paradigms and a way to visualise the differences and similarities.

Rules of thumb A set of rules of thumb for the applicability of agent-based modelling of socio-technical systems: when to use this approach or when another approach may have more advantages.

Literature overview A literature overview of different approaches for modelling socio-technical systems, application domains of agent-based models as well as a literature study on comparisons between equation-based models and agent-based models.

1.5.3 Societal relevance

Alongside this list of contributions to *science*, the thesis aims to offer several practical implications. Firstly, the societal relevance can be summarised in the huge importance of infrastructures on society. One often speaks of *critical infrastructures*, meaning that their failure has major impact on society. Infrastructures are everywhere in all parts of our daily life and disturbances are deeply felt. While, of course, it cannot be claimed that this thesis will lower the risk of failures in the energy sector, increase the capacity of motorways and improve the efficiency of transport networks, still it might be a small part of the puzzle towards more intelligent use of existing infrastructures (See Box 1) and smarter designs for the next generation of infrastructures.

Specifically, this thesis offers the following:

Case-specific lessons This is a methodological thesis that studies a modelling approach rather than a specific system from one of the infrastructure sectors. Still, the framework is applied to a number of cases and, from these, case-specific recommendations may follow. The framework, when applied for a specific problem, can be helpful to decision makers, as will be illustrated with the case studies in this thesis.

Co-learning The framework offers an approach for decision makers to try out different scenarios and to learn about the effects, for example, of hierarchical or distributed control in one domain and compare that with another application domain. Lessons learnt from comparable problems in different domains can be useful this way.

Modelling recommendations Recommendations are given to decision makers in the infrastructure domain (or other socio-technical systems) who may want to use (agent-based) models for decision support tool on what requirements could be set for new model assignments.

Box 1 — Intelligent Infrastructures

The *Intelligent Infrastructures* research programme is part of the Next Generation Infrastructures Foundation.

The operation and control of existing infrastructures is failing: too often we are confronted with capacity problems and a lack of safety, reliability and efficiency. The aim of the Intelligent Infrastructures programme is to develop advanced methods and tools for the operation and control of existing infrastructures. A wide range of problems in various infrastructures are studied.

The Intelligent Infrastructures programme has a focus on the short-term, aiming at developing new, intelligent modes of operation for existing infrastructures. The problems of different infrastructure sectors are comparable: How to maximise the use of available capacity? How to do this in the most efficient way? How to prevent congestion, without neglecting the proper safety precautions?

There are no easy solutions to these problems, because large infrastructure systems have many components and levels, involving different parties, all primarily pursuing their own local performance objectives.

See http://www.nginfra.nl/ for more information.

1.6 Research questions

Following from the problem description, a number of research questions are posed here that will be answered in this thesis. The main research question is the following:

> ***What is a suitable modelling approach for socio-technical systems that allows the user to make changes in both social and physical networks and which can support strategic decision makers to experiment with "what-if" scenarios in a dynamic, multi-actor, multi-objective and multi-level world?***

To help answer this main question, several sub-questions are formulated:

- What does a suitable modelling approach for socio-technical systems look like?
- How can such an approach support decision makers?
- What are different categories of modelling paradigms?
- How can different modelling paradigms be compared in a well-defined way?

These questions will be refined in Chapter 2 after the hypothesis that agent-based modelling is a suitable approach has been tested in a literature study.

1.7 Overview of this thesis

Next, the thesis outline (Section 1.7.1) and readers guide (Section 1.7.2) are presented.

1.7.1 Thesis outline

The rest of this thesis is structured as follows:

Chapter 2 This chapter presents an overview of options for modelling socio-technical systems. The hypothesis that there is no integrated modelling approach yet that can handle the full complexity of socio-technical system and that agent-based modelling is a suitable paradigm to *base* such an approach on is tested. Furthermore, a systematic method to visualise the position of models in a *modelling space* is presented, followed by a discussion on the use of *labels* such as equation-based model or agent-based model. Finally, the research questions as formulated in Section 1.6 are refined based on the findings from this chapter.

Chapter 3 As one of the cornerstones of this thesis, a modelling framework for the development of agent-based models of socio-technical systems is presented. This chapter provides a practical approach to quickly set up modular models, founded on re-usability and a shared language in the shape of an ontology. The concepts formalised in an ontology for socio-technical systems are presented, along with the steps that have to be taken to expand this formalisation for new case studies and new domains. This chapter aims at being a manual for the development of models following the framework.

Chapter 4 The framework has been applied to a number of case studies, some of which are described in this chapter. The focus is on models of an oil refinery supply chain and an intermodal freight transport system. Numerous models developed by others in various infrastructure domains are also briefly addressed. This chapter provides an overview of possible application domains but also illustrates the power of modular and re-usable model components and serves as an argument that the approach is useful to solve real problems.

Chapter 5 The development of the framework from Chapter 3 is an iterative process, and the framework itself has been developed keeping in mind lessons learnt from various case study applications by a number of researchers. In this chapter the development of the framework is analysed, the question whether the framework is ready or needs more iteration cycles is addressed and lessons learnt during the development phase are shared.

Chapter 6 As highlighted in Chapter 2, agent-based modelling is not the only paradigm that can be used for socio-technical systems. This chapter provides a structured and well-defined approach for comparing different modelling paradigms, based on a literature study of other comparisons and methods. A benchmarking study is performed of oil refinery supply chain modelling, and the agent-based model as presented in Chapter 4 is used to compare the approach with equation-based modelling. The conclusions of this exercise can then be used to write down rules of thumb about the applicability of the framework and the advantages of agent-based models.

Chapter 7 The framework (Chapter 3) and the models (Chapter 4) presented in this thesis can be used to support decision makers, as will be demonstrated in this chapter. First a decision problem for the selection of the location for a new inter-modal freight hub is discussed. Next, as an example of abnormal situation management, the disruption in ship arrival in the oil refinery supply chain model is used to illustrate the applicability of the decision support system. The decision support system derives a suitable course of action for a given situation based on the outcomes of a number of simulation runs according to the Nelder-Mead zero-order optimisation method.

Chapter 8 The final chapter returns to the problem statement and the research questions posed here, and shows how the framework presented in this thesis answers these real problems and where the scientific questions have been answered. The conclusions include a critical evaluation of the agent-based approach as well as of the framework.

1.7.2 Reader guide

This thesis covers two different story lines (shown systematically in Figure 1.2), which will be addressed below. The first starts with an illustration of the problems and challenges in socio-technical systems and the need for a flexible, re-usable, bottom-up approach to modelling, resulting in a modelling framework that fulfils these criteria. The framework can then be applied to a number of case studies, each case study again contributing to the generic nature of the framework. For this purpose the *agent-based* mod-

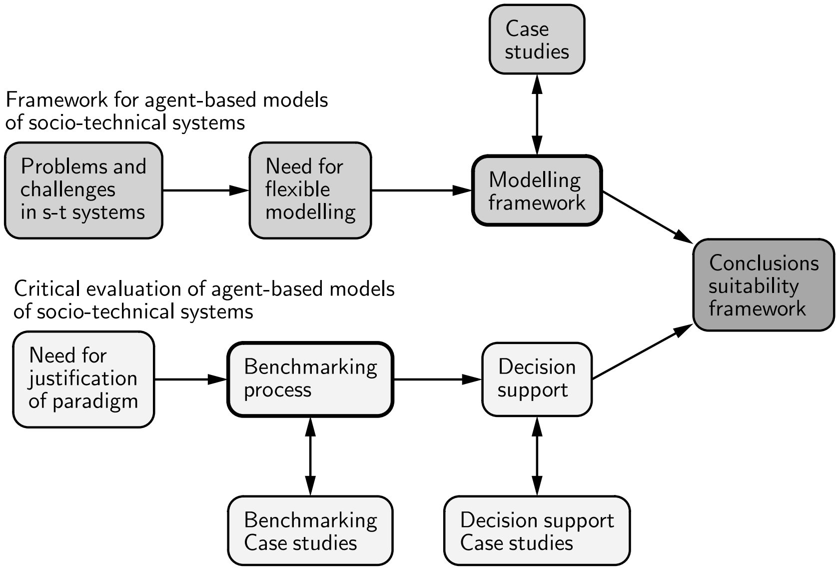


Figure 1.2 – *Two main story lines in this thesis*

elling paradigm turned out as most promising. This story line could be denoted as the 'framework for agent-based models of socio-technical systems'.

The second story line starts with the need of modellers to justify the choice of the selected modelling paradigm, as well as with the scientific challenge to objectively analyse the framework developed in this thesis. After a methodology for systematically performing such a comparison is given, a benchmarking exercise of modelling paradigms is done on a number of case studies. The evaluation of the framework results in rules of thumb for the applicability and its usefulness. Two models developed with the framework are then deployed to support a problem owner, demonstrating how real-life decision problems can be solved with agent-based models. This second story line could be labelled the 'critical evaluation of agent-based models of socio-technical systems'.

Finally, the framework and the conclusions from the benchmarking study converge to lead to the scientific and practical conclusions and implications resulting from this thesis.

Chapter 2

Modelling socio-technical systems

2.1 Introduction

In Chapter 1 the need for modelling socio-technical systems was highlighted. In this chapter different modelling approaches are studied and their applicability to the domain of socio-technical systems is discussed. A literature study of modelling socio-technical systems is conducted to show that an approach that meets the requirements from Section 1.4 does not yet exist and to find approaches that have comparable aims and meet these requirements to large extent, so that lessons can be learnt from them.

First, in part 1 of the literature study, a broad search for modelling approaches for socio-technical systems and research fields and groups that deal with socio-technical systems is conducted in Section 2.2, starting with a brief view and then zooming in on a number of promising approaches. It is concluded from the first part of the literature study that the papers that come close to meeting the requirements use the *agent-based modelling* paradigm, which seems to be a promising one. Section 2.3 then discusses what agent-based modelling is, when it can be applied and how different people use the agent paradigm.

Next, in part 2 of the literature study on modelling, Section 2.4 deals with how agent-based models are used for modelling socio-technical systems by looking at a handful of case studies and frameworks that use agents in a domain that is closely related that the scope of this thesis, that of infrastructure systems. Furthermore, a study of agent-based approaches in one specific field of study, that of *energy*, is conducted.

Finally, this chapter concludes in Sections 2.5 and 2.6 by stating what can be added to the existing body of knowledge and revisiting the research questions from Section 1.6. They are updated with the new findings from this chapter.

2.2 Modelling socio-technical systems

As said in Section 1.2.5, different research fields use slightly different definitions of the concept *socio-technical system*, but the similarities and connections are stronger than the differences. This leads to the hypothesis that tools and techniques from one field may be applicable to socio-technical systems modelling in another field. This hypothesis is tested by studying literature on different modelling approaches and different models in all fields where “socio-technical” systems exist. The quotes are used here because whatever the definition used in a field, papers will be included in the review as long as the term “socio-technical” is used.

The results of this study are threefold:

- An overview of application domains of socio-technical system modelling;
- An overview of research groups or clusters of authors that deal with socio-technical systems modelling; and
- An overview of approaches for modelling socio-technical systems.

The work in this first part of the literature study is done in three steps. The first step is to analyse in which application domains the term socio-technical systems is used and which research groups are involved in making models of such systems. The results are presented in Section 2.2.1. From the first list of papers obtained with a broad search, a selection is made of papers that present a modelling approach or are talking about modelling of systems that lie close to the definition of socio-technical used in this thesis. These papers are discussed in Section 2.2.2. A new selection from these papers is made, with modelling approaches that potentially meet the requirements from Section 1.4. They are analysed in more detail in Section 2.2.3 before coming to conclusions about the availability and existence of a modelling approach suitable for our purposes in Section 2.2.4. The literature review was executed using the Scopus[1] databases. All papers of which the author of this is thesis is one of the authors have been removed from the selection.

2.2.1 Research groups and application domains

A first search for simulation and modelling of socio-technical systems resulted in about eighty papers (see Appendix A). The results for the query are shown in Table A.1 and the application domain as well as a short description of the background of the authors are listed for each paper. The following key application domains can be identified[2]:

- Human-computer interaction;
- Requirement engineering (for software engineering);
- Embedding computer systems in organisations;
- Crisis management;
- Infrastructures;
- Military;
- Medicine/health;
- Manufacturing and
- Evacuation.

[1] See http://www.scopus.com/.

[2] Note that research may fall under more than one application domain (e.g. embedding a computer system in an infrastructure organisation to support evacuation).

The large majority of work that is *unrelated* to the type of socio-technical infrastructure systems that are the scope of this thesis are about *software engineering* and related disciplines. See for example the work of Fischer (e.g. Ye & Fischer 2007), who is most cited author for *socio-tech**[3], and the group of Gregoriades, Shin and Sutcliffe (e.g. Gregoriades & Sutcliffe 2006, Gregoriades & Sutcliffe 2008, Shin, Sutcliffe & Gregoriades 2005, Sutcliffe et al. 2007) who are working together in Manchester in the centre for human-computer interaction at the department of computation.

The application domains 'human-computer interaction', 'requirement engineering' and 'embedding computer systems in organisations' are highly related to each other. The first is about humans users (social) operating computer systems (technical), the second deals with identifying the needs[4] of a client (social) for the development of a software programme (technical) and the third with changes in an organisation (social) after the introduction of a new information system (technical). Again, already in this subset of disciplines that are highly related to one another because they all deal with information systems, different use and meaning of the social and technical aspects of the term socio-technical can be found. Research in these disciplines, especially on human-computer interaction, is often in the domain of the military, which can be explained both by the challenges that occur in this field as well as with the funding opportunities for research (Geiger 1992).

When looking at the affiliation of the authors (see Table A.1), one can see that the background of authors is diverse with people working in different countries, universities and institutes, and, most importantly, different disciplines. Authors are mostly from the field of computer science, but this can be explained not only by the dominant application domain of computer science (as highlighted above), but also because computer scientists employ the simulation and modelling skills from their field to all other domains. Furthermore, a strong number of authors are from technical universities or engineering schools, which fits well with the other application domains identified in this study.

The authors can be grouped in the following way by affiliation. In addition to the group from Manchester mentioned above, Basnyat, Palanque, Schupp & Wright (2007) and Qudrat-Ullah (2008) work at the University of York and Carley (2002) and Yahja & Carley (2005) work at Carnegie Mellon University (but in different groups). Ottens & Marchau (2005), Houwing, Heijnen & Bouwmans (2007) and Thissen & Herder (2003) are all from Delft University of Technology, faculty of Technology, Policy and Management. Govindaraj (2008) and Shah & Pritchett (2005) work at the School of Industrial and Systems Engineering, Georgia Institute of Technology. No other "clusters" can be found in this data set (see also Appendix A.2.1).

The backgrounds of the authors as well as the application domains reveal that, in contrast with what one would predict from the usage of the term socio-technical as identified in Section 1.2.5, the social sciences domain is under-represented in this study. A reason for this might be that the keyword *simulation* does not fit the social science well.

Despite inclusion of *simulation* as keyword, not all papers present an actually implemented model that can be used for scenario-based testing and answering "what if" questions etc. Those papers might not be helpful in meeting our requirements at first

[3] The asterisk (*) is used as a wildcard symbol to represent a group of characters. In this case the search matches socio-technical, socio-technological and other variations.

[4] Requirements engineering is about making specifications for functions or services that become implemented in software as algorithms, and non-functional requirements that express performance and quality criteria for the system as a whole (Sutcliffe et al. 2007).

glance, but it is worth looking deeper into some of them because they can help with theories of socio-technical simulation. There are a number of papers that present frameworks or generic approaches and those are particularly interesting because finding such an approach is the aim of this study.

Papers that meet at least one of the following selection criteria continue to the next stage of the study in Section 2.2.2:

- The paper describes a framework or structured generic approach to build simulation models that can support decision makers;
- The domain is closely related to the socio-technical infrastructure systems that are the scope of this thesis: socio-technical infrastructure systems for example the energy and transport domain; or
- The paper discusses how social and technical aspects and models are combined in one model.

2.2.2 Modelling approaches for socio-technical systems

With a subset of papers from Table A.1, the next step is to analyse the work and specifically searching for papers presenting approaches to modelling socio-technical systems, aiming to find an answer to the objectives from Section 1.4. For each selected paper a more detailed study of the following attributes is done (the bold keywords refer to the labels used in Table 2.1):

S-T Domain: Which application domain(s) does the paper describe? Does the application domain fall within the scope of this thesis?

S-T Definition: Is the domain considered as a socio-technical domain by the authors, or can the view that was taken be seen as a socio-technical perspective? Does the definition of socio-technical system match with the one uses in this thesis?

Simulation: Does the paper present simulation results? Does it present an approach that could lead to experiments and generation of data, as well as recommendations to decision makers?

Reproducible: Are these results reproducible, based on the content of the paper? Is the level of detail of the approach enough so that other models can be developed based on it?

Generalisable: Can the approach presented in the paper be generalised to domains other than those presented in the paper? Is it easy to do so?

Extendable: Is the model presented easily extendable, both for social and physical aspects of the domain? To what extent?

Conclusion: Is the answer to most of these issues positive and do they warrant a more detailed study of the approach?

The aim is then to find papers for which the domain falls within the scope of this thesis (or is generalisable beyond the domain presented in the paper), whose definition of a socio-technical system matches the one used here and that present simulation results. Furthermore, as was established in Section 1.4, it is a key requirement that the approach can be generalised and that social and physical aspects of the system can be adjusted independently from each other. The results of this study can be found in Table 2.1. See Table A.2 for an expanded version of Table 2.1, in which the details of different domains and definitions of socio-technical systems are also included.

The first conclusion that can be drawn is that 19 papers do not match the domain that is the scope of this thesis and only 11 do. Almost all papers claim to present a generalisable approach though, so a negative match for the application does not have to be an obstacle. For the definition of socio-technical system, most (15 against 11) papers use a comparable interpretation of the concept and indeed see the system they work on as socio-technical. The papers that use a different concept are typically in the human-computer interaction domain. In some cases the use of the term socio-technical is not clear as it is mentioned but never stressed why this plays a role. If both the domain and the definition of socio-technical system do not match, the paper is not useful here irregardless of whether the approach presented can be generalised.

Still many papers do not present any simulation results, even though *simulation* was included in the search keywords instead of *model**. Few of the papers that have simulation results provide enough detail to replicate the model or to build models based on the framework. This is unfortunate because model replication is an important part of model validation and comparisons between approaches. However, without exception the approach used is made clear in the paper and with further information (perhaps found in other publications by the same authors) it should be possible to replicate[5] the experiments.

As said, by far most of the papers claim to offer a generalisable approach, even though it remains unclear to what extent this approach can be used in other domains. Because this was the first criteria for selection of papers from those included in Table A.1, it was to be expected that generalisability would score high. It was hard to determine from reading the papers only (instead of studying the models themselves) if the models would be extendable in both social and physical aspects of the system description (and independent of each other). Only Shah & Pritchett (2005) explicitly talks about this possibility. For papers that use a different definition of socio-technical system, answering this question was not meaningful, because of a different interpretation of what comprises the social and technical aspects.

From the papers shown in Table 2.1, five were selected for a more detailed study. See A.3 for a more detailed discussion about this selection.

[5]Note that to 'repeat' an experiment means that the same experiment is performed again with the same model, where 'replicate' indicates that the model is re-built (possibly by others) to carry out the same experiments.

Table 2.1 – *Review of modelling approaches for socio-technical systems. Papers included here were selected from Table A.1. '–' means no match, '+' means a match and '?' means that it is not clear. No answer indicates that the full paper was not available online so it could not be determined. A '√' in the Conclusions column indicates that the paper has been selected for more detailed study*

Paper	S-T Domain	S-T Definition	Simulation	Reproducible	Generalisable	Extendable	Conclusion
Basnyat et al. (2007)	–	–	+	+	+	?	√
Bergman, Haxeltine, Whitmarsh, Köhler, Schilperoord & Rotmans (2008)	+	+	+	?	+	?	√
Carley (2002)	+	+	–	–	+	–	
Donzelli, Setola & Tucci (2004)	–						
Eliasson & Persson (1996)	+						
Govindaraj (2008)	+	+	+	–	+	?	√
Gregoriades & Sutcliffe (2006)	–	–	+	–	+	?	
Gregoriades & Sutcliffe (2008)	–	–	+	–	+	+	
Iivari & Hirschheim (1996)	–	–	–	–	+	–	
Jarman & Kouzmin (1990)	–	–	–	–	+	–	
Johnson (2008)	–	+	–	–	+	–	
Little (2005)	+	+	–	–	+	?	
Liu, Yoshikawa & Zhou (2005)	+						
Maciol & Stawowy (1993)	–						
Masys (2007)	+	?	–	–	+	–	
McIntosh, Jeffrey, Lemon & Winder (2005)	+	+	–	–	+	–	
McNeese, Perusich & Rentsch (2000)	–	+	–				
Moscoso, Wäfler & Windischer (1999)	–	+	–	–	+	–	
Nikitaev (1991)	–	–	–				
Qudrat-Ullah (2008)	+	+	–	–	+	?	
Ramanna, Skowron & Peters (2007)	–	–	–	–	?	–	
Ramaswamy, Thulasidasan, Romero, Eidenbenz & Cuéllar (2007)	+	+	+	+	+	?	√
Saeed (1987)	–	+	–	+	–	–	
Shah & Pritchett (2005)	–	+	+	–	+	+	√
Shin et al. (2005)	?	–					
Simone (1989)	–	–	–	–	+	–	
Smajgl, Izquierdo & Huigen (2008)	?	?	–				
Sutcliffe et al. (2007)	–	+		–	?	–	
Thissen & Herder (2003)	+	+	–	–	+	–	
Yahja & Carley (2005)	–	–	+	?	+	?	
Yilmaz (2007)	–	–	+	–	–	?	
Zarboutis & Marmaras (2007)	–	+	+	+	–	–	

2.2.3 In-depth study of potentially interesting modelling approaches

Next, the five approaches selected in Section 2.2.2 are discussed and conclusions are drawn about their suitability to answer the problems addressed in this thesis. The following five subsections address the multi-scale integrated information and telecommunication system, a modelling approach for socio-technical transitions, modelling of socio-technical barriers for safety critical system design, and, finally, the work analysis framework. For every approach the domain and problem, solution, extension and generalisation are addressed before coming to conclusions in Section 2.2.4.

2.2.3.1 Multi-Scale Integrated Information and Telecommunication System (MIITS)

At the Los Alamos National Laboratory in New Mexico, USA, the MIITS (Multi-Scale Integrated Information and Telecommunication System) modelling approach has been developed (Waupotitisch, Eidenbenz, Smith & Kroc 2006). Ramaswamy et al. (2007) present a case study using this framework.

Domain and problem The approach is applied by Ramaswamy et al. (2007) to simulating the national telephone network, which is considered to be a socio-technical system because its dynamics depend on the interaction between technology and human behaviour. The human behaviour in this case consists of calling patterns, of which data is collected through the means of surveys. The main problem addressed is that of communication networks under stress during emergency situations, such as hurricanes or a terrorist attack. The approach is then used to rank the economic values of each "wire-centre" (where circuit switching takes place by routers) in the infrastructure and assess the effects of emergencies on the availability of the telephone network.

Solution Ramaswamy et al. subscribe to the idea that large scale simulation tool might be the only practical way to analyse the complexity of such systems. This complexity is caused by the real-world dynamics of usage, even though the system itself can be reasonably well understood in isolation. The approach consists of three building blocks: 1) a network generator, which creates a model of the infrastructure (with the wire-centres as nodes) based on industry data, 2) a session generator, which generated individual calls using algorithms of calling patterns based on the survey data, and 3) the simulator which routes the calls over the network in a realistic fashion.

Extension and generalisation The approach is easily extendable for other telephone networks, because the network generator and the session generator are independent of the simulator itself. Hence, different case studies can be done by supplying new industry data for the infrastructure and new caller patterns based on actual use of the network. The MIITS suite has also been applied to other information and telecommunication systems such as the Internet.

Conclusions While the separation of the network generator (physical infrastructure) and the session generator (social infrastructure) is a very useful idea in general and is proven to be useful (Ramaswamy et al. 2007), the approach is not directly suitable for the challenges posed in Section 1.4. The definition of the social system in the MIITS suite is too limited as it can only include generation of traffic over the network (which, even though this has not been done, probably could also be applied to road traffic, public transport, etc). Thus, the MIITS approach appears to be useful beyond the scope of information and telecommunication systems for which it was originally designed, but the social system cannot encapsulate other social aspects such as regulators, operational and strategic decisions and as such it does not meet our requirements.

2.2.3.2 Modelling approach for socio-technical transitions

Bergman et al. (2008) present a modelling approach for socio-technical transition patterns and pathways, developed at Oxford University in cooperation with Erasmus University Rotterdam and as part of the MATISSE project[6] on transitions (Tàbara, Elmqvist, Ilhan, Madrid, Olson, Schilperoord, Valkering, Wallman & Weaver 2007).

Domain and problem Integrated sustainability assessment should be part of policy making (e.g. at the level of the European Union) and models are required to support decision makers on choosing technological and behavioural solutions. A modelling approach is not meant to help predict (i.e. give numerical values) events (both the occurrence and the consequences of these events) or systems (e.g. the end result of an evolutionary process), but — in interaction with users — as a tool to generate insights in the dynamics involved. The systems studied are in various infrastructures domains, including a sewer system, road transport and steam ships. These systems are considered as socio-technical systems because of the need for both technical and behavioural solutions that address changes, with a focus on innovations towards sustainability (and how this may be fostered). The authors claim that numerical models are not good enough for generating such insights and that the political and cultural aspects, among others, cannot be captured in economic models alone. A new approach is therefore needed.

Solution A framework is presented, based on both transition theory[7] and social theory[8] (Haxeltine, Whitmarsh, Bergman, Rotmans, Schilperoord & Köhler 2008). It uses an agent-based approach (building on the "Mason" library[9]) and agents can have an internal system dynamics model. There are three types of agents: niche agent, empowered nice agent and regime agent (all concepts from transition theory). There can only be one regime at the time and the niches represent individuals or technologies outside the dominant set of practices and rules, all looking for "resources". Together agents form a landscape in which they try to survive, while individual actions change the landscape and at the same time the society again influences individual actions. Niches can be so powerful (for example through clustering with other similar niches) that the regime is changed. With the modelling approach one can study, for example, lock-in effects.

Extension and generalisation The approach focusses entirely on transitions, changes in the system and in particular on "radical" changes that go beyond the ordering of the current system. It is not developed for one particular domain but was intended for transition studies in various applications and it has been successfully demonstrated that the approach can be useful in various domains. As such it is a generic framework. It is, however, focussed entirely on "radical" transitions and that idea has been embedded in the core of the framework with the different types of agents for niches and regimes.

[6] Methods and Tools for Integrated Sustainability Assessment. See http://www.matisse-project.net/.

[7] The field of research studying long-term technological developments and changes, see for example Geels (2002).

[8] Theoretical work on understanding and explaining the causes and consequences of social change, see for example Noble (2000).

[9] An open-source library for agent-based simulation for discrete events. http://www.cs.gmu.edu/~eclab/projects/mason/.

Conclusions This approach is a potential alternative way to handle some of the problems discussed in this thesis (for example the evolution of industrial clusters, see Section 4.5.1). However, its strong focus on transition theory makes it less flexible for analysing what-if scenarios with lower-level changes (such as experimenting with different behavioural rules of an agent) or experimenting with new technologies for the same social network, for example. The approach does have matching assumptions (e.g. models of such complex systems are mostly useful to gain insight instead of make predictions) and goals (support decision makers on both technological and organisational changes).

2.2.3.3 Modelling framework for nuclear power

Govindaraj (2008), from the School of Industrial and Systems Engineering, Georgia Institute of Technology, USA, presents an approach to modelling organisational issues for a nuclear power plant.

Domain and problem Safety and reliability are key characteristics of electricity production and, especially in the case of nuclear power, serious accidents have major consequences. Managing such systems in a way that prevents such accidents is critical and models are needed for this. The aim of the model presented is to predict undesirable events in nuclear power plants and this can, according to Govindaraj, only be done using a *variety* of modelling approaches[10]. There is a zero tolerance for critical events so problems have to be detected before they occur. However, not only the technology but also cultural and organisational issues could lead to disasters so socio-technical models are called for. The aim is visualising relationships that are not apparent from traditional analysis.

Solution In socio-technical systems modelling the technological components is generally not a significant problem, because analytical or computational models are available. The technological aspects of nuclear power generation are well understood and a lot of data has been collected. Furthermore, the organisational aspects are only of a moderate degree of complexity as well, but still it is not possible to build an analytical model of the whole system. The paper briefly looks at relevant methodologies, including network models, statistical and probabilistic methods as well as examples from finance and climate prediction, before presenting a graph based approach. The key attributes (collected from surveys among experts e.g. not following operations standards, high rotation in position of operations manager, loss of key personnel) are represented as nodes and edges connect related nodes that influence each other. These nodes are clustered, for example in operations and engineering, leadership or plant events groups. Simulations are then used to compute the plant performance (in four steps from least desirable to most desirable) and to rank the nodes in order of importance so their impact on the plant performance can be elucidated.

Extension and generalisation The approach is first developed for the nuclear domain but the results should be more widely applicable to other critical infrastructures. The Columbia space shuttle disaster is given as another illustrative example (but with higher degree of complexity) as well as studying invasive species in ecosystems (again even more

[10] The latter is also in line with Mikulecky's (2001) definition of complexity.

difficult). The model has not been fully implemented, so it is difficult to draw conclusions about its extension and generalisation.

Conclusions The organisational issues mentioned in the paper are for example political pressure and insufficient emphasis on leadership skills in human resource management. This is different from the social aspects considered in this thesis. Also, the model is not a simulation model of the socio-technical system itself but of factors that have an impact on the performance of the system. Furthermore, the aim of the model to *predict* the occurrence (not the consequences) of *events* seems impossible because of the high complexity of the systems. Still, such models can be used to gain insight in the sensitivity of attributes of certain problems.

2.2.3.4 Modelling socio-technical barriers for safety critical system design

Basnyat et al. (2007), working at Université Paul Sabatier and the University of York, presents an approach to modelling barriers for safety critical systems.

Domain and problem The approach presented by Basnyat et al. (2007) aims to improve the safety in interactive systems. Safety-critical systems, such as infrastructures, have dedicated risk reduction systems that need to prevent escalation of incidents. These risk reduction systems, called barriers, are often socio-technical: not only a technical element is needed (e.g. a fire extinguisher) but also the human elements (the person using the fire extinguisher, but also training needed for this and the chosen location of the device). The case study used is about an incident in a mine where a fault in the waste fuel system occurred. Basnyat et al. (2007) describes how this case is analysed and which safety systems can be installed to prevent such a disaster from happening in the future.

Solution For the analysis a group of experts use a common hazard identification method and afterwards these are modelled with the following approach. In the first step the Safety Modelling Language (SML) is used to define the relationships between "hazards" (something that could potentially have a negative effect on a target) and the rest of the system. These are causal links. The second step is to analyse, design and model each possible barrier and to describe its behaviour and function. Finally, in the third step these barriers are included in the system model so that their effect can be simulated and conclusions can be drawn about the effectiveness of the barriers before they are introduced in the real system. The special focus in this approach is on the description of the barriers (which are the socio-technical elements in the system). The Interactive Cooperative Objects formalism (a Petri-net based model to describe, for example, the states of the system and state changing operators (Navarre, Palanque, Dragicevic & Bastide 2006)) is used for the barrier descriptions. A formal system description is made using well-defined concepts and links between them. The barriers and the system are both described using the same formalism, allowing the models to be connected.

Extension and generalisation The approach is described in a generic way and could be used for any system were barriers can be identified. It has also been applied to interactive cockpit software and a cash machine system and the authors plan to use it in additional case studies. Because of the use of a shared formalism additional components can easily be added to the system.

Conclusions While the barriers in these safety-critical systems are considered as socio-technical, the system for which the barriers are designed is also seen as a socio-technical system. In the case study human operators are included and they have to be protected from making the wrong decision. However, in the model the waste fuel system is considered as a purely technical system of pipes and valves. Therefore the approach does not seem to be suitable for including models of human actors in the system and their behaviour. However, the idea to use a shared formal language for different parts of the system to they can be integrated in one model is important and useful.

2.2.3.5 Work Analysis Framework

Shah & Pritchett (2005), at the School of Industrial and Systems Engineering, Georgia Institute of Technology[11], USA, created a framework for describing humans working in their environment.

Domain and problem Shah & Pritchett use the definition of socio-technical systems as a system comprising people, technologies, physical surroundings, processes and information, but state the term was used in the past to merely identify technology in its social setting. The focus in Shah & Pritchett's work is on humans working in the system and their environment and the impact the environment has on their work. The design variables in such a system are humans, elements of the work environment and how they are connected and influence each other. The key questions are then: What is a suitable technology mix? How well do technologies perform in a different organisational set-up? Which training is needed? Which design can cope with variation in human behaviour? Other approaches do not support answering all these questions, but perhaps concentrate on one or two. The case study employed is that of air traffic control and the selection of different procedures for routing air planes towards the runway.

Solution An agent-based approach is proposed, in which the agents represent the workers. In the case study these are the pilots and air traffic controllers. The agents are defined by a set of capabilities and objectives by the modeller to fit the work environment that is being simulated. The model can produce emergent behaviour at the system level caused by the behaviours at the lower levels, but Shah & Pritchett are not clear what this behaviour is and how it would be measured. When the work environment changes (e.g. automating of manual work) the workers have to be trained so agents have to be tuned for each change in the work environment too and new simulations have to be run. Experiments with the multi-agent simulation model can be done to try different procedures, regulations and technologies in the work environment.

Extension and generalisation The framework is designed to fit any domain and any scale. While the paper focusses on air traffic control, other examples given are transportation systems, military organisations and corporate enterprises. A modular approach to

[11] Note that this is the same group as Govindaraj (2008).

defining the work environment is used, where each element is defined as a separate component. The same can be said about the agents, built-up from pluggable executable components. This makes it possible to perform simulations with different configurations and to adjust the model to match new domains: the framework is re-configurable and extensible.

Conclusions The questions addressed by Shah & Pritchett (2005) are similar to the ones that are the motivation for the work in this thesis. The idea of a re-configurable and extensible framework is essential to address the challenges posed in Section 1.4 and to deal with changes in the structure of the system modelled. Also, the physical environment can be described separately, which looks promising. However, the approach seems limited by the fact that the work environment is not represented as an agent and that there seems to be no support for modelling a physical infrastructure and its behaviour.

2.2.4 Conclusions part 1

From the papers on simulation of socio-technical systems a subset was selected with papers that describe a framework or structured generic approach, are on a domain closely related to the socio-technical infrastructure systems that are the scope of this thesis or that discuss how social and technical aspects and models are combined in one model. Looking at these potentially interesting modelling approaches, there is no approach yet that fully meets the requirements.

The main reason for this is that a different view of socio-technical systems is used. Socio-technical system are studied in a wide range of scientific fields, from anthropology to mechanical engineering. The approaches and tools needed for — and resulting from — research in these fields are used on a wide range of application domains, ranging from nuclear power generation to evacuations in metro stations. Because of the wide range of fields and application domains many different interpretations of the concept *socio-technical system* exist that are often only partially overlapping. This means that the approaches and tools, even when designed to be generic, cannot always be used outside the frame in which they were created.

As said in Section 1.2.5, the view of such systems in this thesis is that of physical installations and their owners or controllers who make decisions about the physical installations. None of the approaches discussed in Section 2.2.3 offer a natural and straightforward way to model such systems and to support the decision maker in experimenting with different configurations of either the social, the physical, or both networks, as well as of different decision making rules for the actors or different characteristics of the technical nodes.

Still, lessons can be learnt from these papers. Basnyat et al. (2007) model different aspects in the same language so they can be combined in one system level model. That is a useful approach towards combining different model elements. Ramaswamy et al. (2007) generate the social and technical network separately from one another. Different sources are used to do this (e.g. survey information for social network generation) and as such the social and physical networks are separate and can be changed independently from each other, which is not only valuable but also feasible.

The aim of the work by Bergman et al. (2008) is an important one: the models are not designed to predict the future (as they cannot) but are meant as a tool to generate insights about the dynamics, in interaction with the user. Their approach to modelling transitions could be used to answer some of the challenges of this thesis, but not all. Most importantly, their idea of clusters of comparable technologies that compete with each other is interesting.

Govindaraj's (2008) model has not yet been fully implemented, but its approach of combining different modelling paradigms is a promising one. Again, it is however not suitable to model the behaviour of the physical and social components in the network, but a more distant perspective is chosen where factors that impact the performance of the system are modelled rather than the actors and equipment that causes these factors. That makes it a useful approach to analyse the links between the factors but unsuitable for the aims set out in this thesis.

Shah & Pritchett (2005) say that design changes occur at a lower level and that the system level changes emerge from this, showing the importance of bottom-up modelling. The agent-based approach that is proposed seems suitable for the problem. The work environment is, however, not modelled as a collection of agents, but as something outside the agent. This seems counter-intuitive where the environment exists of other workers and when the behaviour of other actors plays a role. The fact that their architecture is dynamically re-configurable is, however, a strong point.

Finally, the approaches studied here that come closest to reaching the goal (Bergman et al. 2008, Shah & Pritchett 2005) use the agent-based modelling paradigm, which seems to be the most appropriate approach to handling socio-technical complexity and create the flexible models needed to perform "what-if" studies. It is therefore interesting to continue the literature study with a stronger focus in this direction and to explore how agent-based models have been used in socio-technical systems. The next section briefly discusses the agent-based modelling approach, before continuing with the literature study on applications using precisely this paradigm to answer the question if this could be the right paradigm to use in the model framework presented in the next chapter.

2.3 Agent-based modelling

In agent-based modelling, a model of an actor, or a group of actors (e.g. a company, a governmental institute, a community of citizens), is called an *agent* (see also Section 1.2.4). An agent can be seen as a software entity that is autonomous, reactive, pro-active and capable of social interaction (Wooldridge & Jennings 1995, Jennings 2000). The behaviour of an actor can be formalised using algorithms with, for example, *if-then* rules: the so called behavioural rules. The key distinguishing element, that sets agent-based models apart from other models, is a focus on modelling *individuals* who can make decisions. For an introduction to agent-based systems see for example Weiss (1999) and Wooldridge (2009). Luck, McBurney, Shehory & Willmott (2005) present a "road map" for agent-based systems with trends and views on how agent technology will likely develop over the coming years. This includes technological developments (e.g. architectures, and standards) as well as possible application domains.

2.3.1 Applications

By modelling components rather than the entire system, the structure of the system is not pre-defined. Because agents can communicate and link with other agents, different networks can be created by changing the behavioural rules without explicitly defining which relationships are to be made. This way different set-ups of a control system (e.g. hierarchical or coordinated, see van Dam, Verwater-Lukszo, Ottjes & Lodewijks (2006)) can be tested in a simulated environment and the agents' response to the emergent system behaviour can be monitored.

If system behaviour is modelled explicitly, as is common in numerical approaches for example, making changes in the model would require the modeller to adapt the system structure. That way it is possible to compare different configurations of the system, but it is not clear how the most desirable situation can be obtained by influencing lower levels of the system. Agents (like the actors they represent) can exist in several levels of hierarchy, for example if one actor supervises the activities of one or more other actors creating subsystems (van Dam & Lukszo 2006). Agent-based models, due to their bottom-up nature, are suitable for simulating dynamic systems where the structure can or should change during a simulation run, or where experiments with different configurations have to be done.

In general, the agent-based approach is applicable for (conceptual) modelling of complex systems if the following conditions are satisfied (van Dam & Lukszo 2006):

- The problem has a distributed character;
- The subsystems operate in a highly dynamic environment;
- The subsystems have to interact in a flexible way; and
- The subsystems are characterised by reactivity, pro-activeness, cooperativeness and social ability.

Agent-based modelling seems to be a suitable approach to create models of socio-technical systems.

The agent-based formalism has started to receive much attention and is being used in a wide range of domains. Katare & Venkatasubramanian (2001) use agent-based learning to model the dynamics of microbial growth. Eo, Chang, Shin & Yoon (2000) and Davidsson & Wernstedt (2002) illustrate the suitability of agent-based systems for process monitoring and control. The use of agents for three different problems in chemical process engineering (intelligent search, process design and configuration of team members) is explored by Aldea, Bañares Alcántara, Jiménez, Moreno, Martínez & Riaño (2004) and a number of other process systems engineering areas where the formalism is beneficial are highlighted. One such was further investigated by Siirola, Hauan & Westerberg (2004), using multiple optimisation agents to derive the Pareto front for a multi-objective optimisation problem. Bussmann, Jennings & Wooldridge (2004) explore the possibilities for agent-based systems in manufacturing control. Agent-based models are now widely considered to be a promising approach for decision support in supply chains (Gjerdrum, Shah & Papageorgiou 2000, Julka, Srinivasan & Karimi 2002, Julka, Karimi & Srinivasan 2002, Siirola, Hauan & Westerberg 2003, Ydstie 2004, Mele, Guillen, Espuna & Puigjaner 2005). In addition to applications in more technical systems, agent-based systems are also frequently applied in social sciences to study a broad range of phenomena and human behaviour (Epstein 2007, Billari, Fent, Prskawetz & Scheffran 2006, Terna 1998).

2.3.2 Interpretations of the concept "agent"

Different researchers use different definitions of *agent-based systems*. To identify commonality among the various perspectives on agent-based modelling a small survey was designed and sent to a group of researchers with a strong interest and contribution to the agent-based systems area (see Appendix B for a list of questions used as well as for the list of researchers who responded). It became clear that the concept "agent" has a different meaning when used in "agent-based models" and "multi-agent system". When talking about multi-agent systems, characteristics following, for example, definitions by Wooldridge & Jennings (1995) (as also listed above) are common while no positions leaning towards Dennet (1987) are taken. When used in an agent-based modelling context, it appears to be more a metaphor — or way of thinking — towards modelling the behaviour of individuals, rather than a strict definition with minimum requirements.

As such from this small survey two different "schools" can be identified. Agent-based modelling deals with making a model to *simulate* an actual system where the agents are models of decision makers that exist in the system under study. A multi-agent system, on the other hand, involves *creating* distributed decision makers to perform a certain task, such as in a distributed (control) system. Both use the same vocabulary and sometimes even the same tools, which can cause confusion. However, in any case they share a way of thinking in terms of distributed elements with a focus on individuals.

Luck et al. (2005), after an eighteen-month consultation involving a large number of experts on agent-based modelling, suggest that agent technologies can be considered from three different perspective: agents as a design metaphor (i.e. the agent paradigm offers software developers a way to structure an application around autonomous components), as a source of technologies (i.e. agents can be used as the key elements in a problem solving algorithm such as for resource allocation) and finally as a simulation (i.e. agents can represent real-world domains, for example because the domain is too complex to be modelled otherwise). Using this distinction, it can be said that the agent-perspective used in this thesis is that of the third type: agents as simulation.

2.3.3 Modelling paradigm spectrum

One other conclusion that can be drawn from the fact that different modellers have different views of the concept of an agent and that different schools can be identified is that there is not a clear line between agent-based models and models *not* based on the agent paradigm. The concept is not black-and-white, rather there is a continuous scale, or a spectrum in the modelling space, where a model can be more agent-based or less so. There are two main axes in which models can differ: The *model elements* and *system description elements*. The former deals with *what* is modelled and the constituents of the model, the latter with *how* their structure and behaviour is formally described (van Dam, Adhitya, Srinivasan & Lukszo 2008).

First, it should be considered *what* is being modelled. The model elements can range between *system observables* and *individuals*. System observables[12] are the flows and states that can be observed in the real system, without taking into account who or what caused them (and, most important: why). These are the *results* of actions. On the other end of the spectrum, a focus on individuals means that the system is modeled by capturing the

[12]The concept "observable" is used here from the perspective of the *modeller* and should not be confused with observations done by the elements in the *model*.

Table 2.2 – *Attributes of modelling. The arrows illustrate that there is not a strict division between the rows but a continuous scale*

Model label	Model elements	System description elements	Implementation platform
Equation-based model	System observables (flows and states)	Equations	Mathematical software tools
⇳	⇳	⇳	
Agent-based model	Individuals (decision making entities and executing entities)	Algorithms	Agent-based software tools

behaviours of exactly these *decision making* and *executing entities*. The behaviour of an individual leads to actions that, together with the actions of all other individuals, cause system level behaviour, which can be observed in the model.

Next, there are different ways to formalise the structure and behaviour (or in other words, *how* the model is built). Various description elements such as *equations* or *algorithms* can be used. An equation is a mathematical statement that two terms on either side of the equals sign are equivalent. Algorithms are well-defined sequences of instructions. One could also use different names for the *how* axis, such as mathematics-based on one side, and logic-based on the other.

Agent-based model and equation-based model are *labels* used to describe a model and they can be characterised by their use of these model elements and system description elements. As for agent-based models, in general they are characterised by a focus on individuals as model elements. Equation-based models, on the other hand, focus on system observables modelled predominantly using equations[13].

Table 2.2 shows the model labels, their predominant model and system description elements, as well as and the commonly used implementation platform, for both equation-based models and agent-based models. There is no strict division between the rows. Equations, for example, are system description elements that can be used to describe certain effects or observed behaviour, but are not exclusive to non-agent based systems. Even though they may be predominant in models built up from system observables, they may very well be applied in individual-based models too to model the behaviour of these individuals.

Where agent-based models are mostly identified by the model elements (second column), equation-based models are mostly identified by the system description elements (third column), resulting in a space in which it is not clear how to *label* a model (first column). This also means that the use of equations is not the opposite to using an agent-

[13] An essential point to be noted in this context is the following. Once any model has been constructed it has to be simulated or solved. The presence of algorithms in the model description is qualitatively different from those being used in the solution procedure. Both agent-based models and equation-based models would require algorithms in the solution procedure, however, only agent-based models would contain algorithms in the model description itself.

based model nor is it an alternative per se, as is often stated. Rather, *agent* and *equation* are concepts of a different order.

Figure 2.1 illustrates the modelling space and plots possible implementation platforms that can be used to create the models. A point *on* the x-axis illustrates that both equations and algorithms are used approximately to the same extent and are equally important. A point *on* the y-axis means that individuals and system level observables are equally predominant. A point on the extremes means the model only uses one type of model or language elements. Other points in the space highlight the predominant, but not exclusive, characteristic.

In general, what is called an agent-based model can be found in Quadrant II and traditionally equation-based models are in Quadrant IV, but this is not exclusive as will be demonstrated in Section 6.5. Other examples of models in Quadrant III could be purely continuous physical systems such as liquid flow or molecular dynamics where models describe the behaviour of a large number of system constituents (individual molecules) using equations. Quadrant I, in which models would use algorithms to model system observables, appears to be an uncommon modelling style.

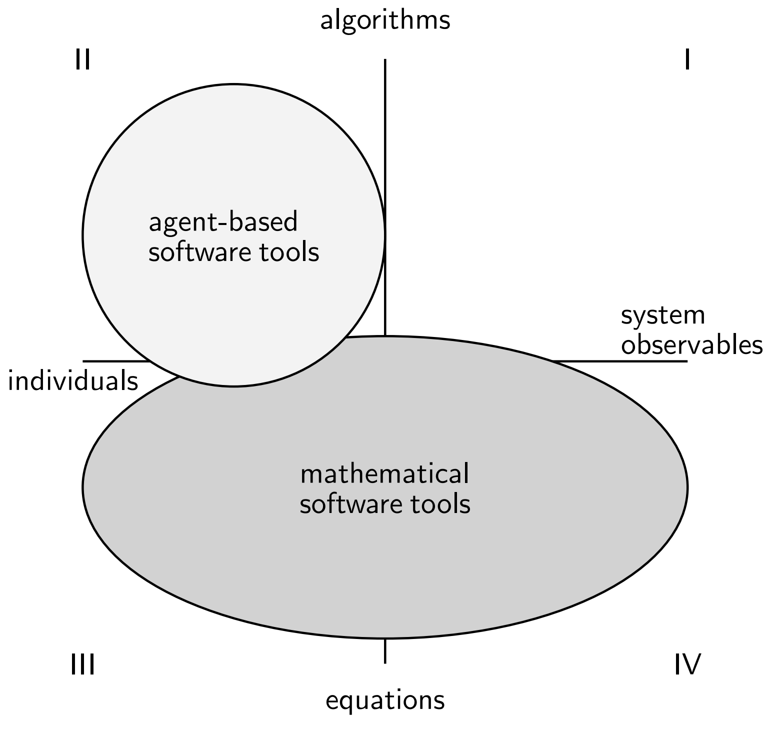


Figure 2.1 – *The space of models based on algorithms or equations, and focus on individuals or system observables. The areas indicating which software tools are most applicable are without clear borders (based on van Dam, Adhitya, Srinivasan & Lukszo (2008))*

Having emphasised that there is no “black” or “white” when it comes to the label *agent-based model* or *equation-based model*, models can be mapped to indicate their essential characteristics. This illustrates not only how the various models are different but also to what extent they are similar. This formulation, by acknowledging the absence of a clear dichotomy, makes stark contrasts more difficult, but, for a fair comparison, the similarities between models should also be fully captured (van Dam, Adhitya, Srinivasan & Lukszo 2008).

Several models presented in this thesis are later (in Figure 6.3) placed in the space of Figure 2.1 to show their differences and similarities.

2.4 Agent-based modelling of socio-technical systems

A short literature study specifically of agent-based approaches (both agent-based models and multi-agent systems) applied to socio-technical systems is conducted. These are approaches that did not surface in Section 2.2 because the keyword *socio-technical* is not used but they *do* deal with systems that are considered as socio-technical. These papers not only give insight in additional approaches to modelling, but can also help find new keywords to continue the search and to better position the work in this thesis.

Therefore, for each of these papers not only the domain and problem, solution, extension and generalisation are presented in a similar fashion as for the papers in Section 2.2.3, but also attention is paid to the keywords: which keywords does the paper use and which terms could be used to find other, similar papers? Based on the conclusion that no keywords are used except domain specific ones, a study for agent-based approaches in the field of energy systems is conducted and presented in Section 2.4.4. Finally, in Section 2.4.9, conclusions on this second part of the literature study are drawn.

2.4.1 Agent-based model for energy systems analysis

Hodge, Aydogan-Cremaschi, Blau, Pekny & Reklaitis (2008), at Purdue University, present an approach to modelling energy systems.

Domain and problem In the electricity sector new technologies emerge but it not immediately clear how they can be incorporated into the existing electricity system. New research makes innovations available and can help make existing technologies cheaper, and (political) choices have to be made about subsidies and taxes. Models can create insight in the current state of the system and how it evolves and provide decision support on investments and subsidies. Top-down models, however, only describe the markets in which energy technologies operate but they do not explicitly represent the technical potential of such technologies. Bottom-up models, on the other side, do not consider their market adaptation. A *systems* modelling perspective is therefore needed, which can be considered as a socio-technical perspective. The energy system of the state of Indiana in the USA was taken as a case study.

Solution A framework for agent-based simulation of energy systems is presented (Hodge et al. 2008). The goal is to develop a large-scale energy modelling framework which can be used as a tool for evaluating the effects of energy policies on new technology growth and integration into the current energy system, show the mechanisms by which changes occur in energy systems, portray new technologies accurately while accounting for market adoption and, finally, examine the role of research in technological improvements.

Agents in the framework make independent decisions based on information they receive and the system structure used for the communication of the agents is therefore essential. A network view of the world is taken, in which the agents are the nodes and the edges represent lines of communication. There are six classes of agents: raw material agents (responsible for extracting raw materials from the ground and selling them), producer agents (convert raw materials into end-use energy products such as diesel of electricity), consumer agents (who have a demand for energy), research agents (used to model the advancement of technologies), government agents (who can influence the system with taxes and subsidies) and finally an environment agent (to model the effects on and from the world outside the system boundaries).

Agent behaviour is modelled with a set of rules for each agent class, for example for a production agent to decide on the amount of energy to produce and the initial asking price as well as on the amount of raw materials needed and the price that can be paid for them. Interaction between agents mostly concerns negotiation (modelled as "take it or leave it") about products and price.

Extension and generalisation The framework is set up in a modular fashion and new agents can be added to the system, thus creating a new model of a different energy system. It appears the existing case study of the state of Indiana could easily be adjusted to model a different state in the USA, for example, just by creating the appropriate producer and raw material agents based on new data (e.g. geographical information and available raw materials and technologies). This means the approach can be extended. According to the author the description of the technologies can also conveniently be adjusted for new case studies.

The framework has been developed specifically for the energy domain and the learning curve used to represent the state and cost of technologies is based on research in the energy domain too. However, even though not specifically designed for this purpose, the approach appears to be more widely applicable to other related domains such as the petro-chemical industry in which actors buy and sell products from each other.

Keywords The title, abstract and list of keywords (i.e. 'Multi-agent systems', 'energy systems analysis' and 'learning curves') do not include terms to indicate that the system is considered as a socio-technical system, but it is. The fact that it is a socio-technical system can only be deduced from the name of the application domain and the way the model was implemented.

Conclusions The approach is generic and can be extended easily, making this a powerful modelling framework for socio-technical systems. The illustrative case study shows that approach can be validated on a real system by replaying an historical scenario. No explicit distinction between the physical and the social elements of the system is made, and the agents represent both the technology and the decision maker (note that some agents, such as the research agents, do not operate a technology themselves). In the paper no new keywords were found that could help forward the literature study on approaches to model socio-technical systems and there are no references to other agent-based models in the energy domain that capture the same dynamics and interactions.

2.4.2 Multi-agent model predictive control

Negenborn (2007) developed a multi-agent model predictive control method at the Delft Centre for Systems and Control at Delft University of Technology.

Domain and problem Transportation systems, in particular power distribution and transmission networks, need to operate in a safe and reliable way and current control strategies do not suffice any more. With the changing structure of power networks (due to distributed generation and increasing integration of national grids, etc.) a central control is no longer possible and purely local control does not guarantee global optimisation.

Solution A multi-agent approach is suggested, in which agents are responsible for controlling a local segment of the network and, through communication with other agents, cooperate to find the best overall performance. Each agent uses model-predictive control to incorporate available information and anticipate on undesired behaviour to make optimal decisions. Different configurations are possible, including multi-layer control and overlapping sub-networks. The model used in the predictive controller deals with the behaviour of the part of the network controller by one specific agent.

Extension and generalisation The model can easily be extended by adding more agents that control other sub-networks or by re-dividing the network and the control tasks. As such it is extendable for both social (i.e. controllers) and technical (i.e. the network to be controlled) aspects. Controllers can be re-used between models. The approach has also been applied to the water domain (Negenborn & De Schutter 2008, Negenborn, van Overloop, Keviczky & De Schutter 2009) and is generic for any transportation system where distributed controllers are or can be used. Other listed examples of suitable application domains include railway networks and autonomous guided vehicles.

Keywords The term socio-technical is not used, even though it does highlight that different actors are involved in the power systems at different levels of decision making (but in the multi-agent approach the decision makers are not models of human actors). No different keywords are used in the abstract that could be used to find similar approaches, except by searching for both the method or the domain.

Conclusions The multi-agent model predictive control approach is a good example of a multi-agent system approach, where the agents are not used to *model* units that exist in the real world, but where they are controllers that can be placed in a real system. It is designed for the control of transport systems in a generic way and can be re-used and applied beyond the domain of the energy sector. Finally, the approach has a clear way of visualising the negotiation process between agents and as such can be useful to explain what the actions taken by the agents.

2.4.3 Controlling electricity failures with cooperative agents

Hines & Talukdar (2007) worked on a multi-agent control approach for electricity networks at department of Engineering and Public Policy, Carnegie Mellon University in Pittsburgh, USA.

Domain and problem Cascading failures in electricity networks start with a failure of equipment which triggers additional outages because new operating constraints are introduced, possible leading to a large black out. The challenge is to design controllers by eliminating violations before they cause outages. Objectives of the operation of power systems, including maximising economic benefit, minimising risk of service disruption and infrastructure damage, are often conflicting and especially during a situation of cascading failure it is difficult to balance between these objectives.

Solution A distributed control approach is used, with autonomous cooperative agents each responsible for load and generator control to avoid voltage and current violations. Existing controllers operate with local information and simple rules, but with advances in communication it becomes possible to design cooperative agents that together can solve complex network problems. The control problem is written as an optimisation problem using equations, and the global problem is decomposed into sub-problems that can be assigned to one agent. Each agent uses model-predictive control to optimise its actions based on the predicted actions of others. Agents cooperate by telling neighbours what they intend to do and to pass on information that other agents may not be able to sense otherwise.

Extension and generalisation A generic decomposition of problems in sub-problems (that can be assigned to agents) is presented, where each sub-problem is simpler and based on a unique view of the network. The scheme proposed by Hines & Talukdar (2007) can also be used for different types of control problems. The size of the physical network, and therefore the number of agents, is variable so the network model can be expanded easily. Furthermore, one can assign a smaller part of the network to individual agents, increasing the number of controllers for the same given physical network.

Keywords The paper uses the following keywords: cascading failure, autonomous agents and electrical power networks. The keywords highlight the problem, the solution and the application domain. The abstract mentions the social consequences of power failures and in the introduction the fact that, next to electro-mechanical controllers, many human operators are involved, but this is not expressed in the keywords.

Conclusions The approach does not use agents to model elements in the real world but to design software controllers that can be used in a distributed fashion ("place a software agent at each load and generation bus"), as was also the case in the work of Negenborn (2007). Hines & Talukdar (2007) use a stronger focus on cooperation methods and there are differences in the implementation of the control system, which are not relevant for this study. Again, an electricity network is used as application domain, which emerges as the predominant field of research for agent-based systems in socio-technical systems.

2.4.4 Agent-based modelling of transport and energy systems

At the Imperial College London, Keirstead, Samsatli & Shah (2009) are working on agent-based modelling of urban transport and energy systems.

Domain and problem Urban areas have an enormous energy demand and, with rising energy prices and awareness of environmental issues, it is a challenge to try to reduce this (Keirstead & Leach 2008). Different aspects of urban life are highly linked (e.g. transport for work, electricity use of offices) and to meet this challenge work on these individual domains has to be integrated. Clearly the system is a socio-technical one, including human behaviour (travellers and other energy consumers) with the physical infrastructure of the built urban environment. The goal of the project is "to identify the benefits of a systematic, integrated approach to the design and operation of urban energy systems, with a view to at least halving the energy intensity of cities". Another challenge is the scale of the urban area with possibly millions of heterogeneous individuals.

Solution A modelling platform called SynCity[14] is being developed, with the aim to bring together different city representations (layout, transport, resource flows and energy networks). The model contains everything from citizens to the city's infrastructure and resources and processes, which means that different software has to be combined. An architecture using an ontology and a set of shared tools can bring together software models from different domains (for example land use, as well as transport demands). The individuals in the urban area are then modelled as agent with their own properties and behaviour. In other words, the agents are a model of how the urban space is used when it comes to energy.

Extension and generalisation SynCity is based on an earlier project, UrbanSim, which focussed on urban planning (e.g. macroeconomic simulation and travel demands). The modelling platform is extendable to different sizes and, because the ontology brings helps to merge descriptions for the various aspects of the system, it is straightforward to adjust either the social entities or the physical reality of the system. As such it is full extendable. It appears[15] that the approach is fully targeted at citizens in an urban area so it might not be re-usable in a model studying other socio-technical system.

Keywords Keywords for this work include integrated modelling and holistic. For the domain the keywords urban energy system or eco-town can be found, which are more widely used terms.

Conclusions This is promising work, but it is still in the prototype phase and no models have been built beyond a number of proof-of-concept models. It is a modelling approach targeted only at urban energy systems and it is not the aim to develop a generic approach for other socio-technical systems. The main lesson that can be learnt from this work is the use of an ontology to bring models from different disciplines together.

[14]Short for Synthetic City.

[15]It is at this stage hard to draw stronger conclusions on this as the work is still in full progress.

2.4.5 BRIDGE agent architecture

Dignum, Dignum, & Jonker (2008) present an approach towards the use of agents for supporting policy makers.

Domain and problem Models that can support decision makers need to capture individual decision making given subjective social norms, individual preferences, and policies. The elements in such a model are humans with own personalities and cultural backgrounds, etc. Realistic social interaction has to be included to be able to evaluate policies and traditional agent-based models do not allow this level of complexity. No specific application domain is mentioned.

Solution The Belief, Desire and Intention (BDI) model of human reasoning (Bratman 1999) which has been applied to agent architectures, is expanded to include Beliefs, Response, Intentions, Desires, Goals and Ego (BRIDGE) so that human behaviour can better be modelled more realistically. Different personalities (e.g. extraversion vs. introversion, feeling vs. thinking) can determine how the agents deal with stimuli and perform their reasoning. An architecture with three layer of descriptions (macro, micro and meso) is proposed, with the middle (meso) level coordinating between the micro (e.g. the characteristics of individuals and groups) and the macro (e.g. an abstraction of the overall system) levels.

Extension and generalisation The architecture should work for any domain where policies are designed and human responses can accurately be modelled. There are no limitations to generalising. For the extension of either social and technical entities it depends on how the different levels will be implemented (e.g. the macro level for the physical system) but the aim is to develop an architecture that is fully modular.

Keywords Keywords used in the introduction include public policies, multi-cultural composition, BDI, decision support. No focus is set on the physical system or the socio-technical interactions.

Conclusions Dignum et al. (2008) have presented a position paper and the research is still at an early stage[16]. It is a fascinating direction because of the specific challenges of including more realistic human behaviour and characteristics, something that has not been seen in other papers in this literature study. It is clear such behaviour plays a key role in how successful policies are. The approach is not explicitly socio-technical (even though the difference between micro and macro level could be seen as a step in this direction) and there is no explicit way to model a physical transportation system or process.

2.4.6 Modelling the evolution of large-scale socio-technical systems

Nikolic (2009) performed a literature study of modelling efforts for large scale socio-technical systems, such as the petro-chemical industry cluster in a large harbour. The search was limited to agent-based modelling and included 171 papers. He concluded that there is currently no modelling framework that meets the requirements to study

[16]Dignum et al. (2008) themselves say "the question is how this framework should be implemented".

co-evolution of industry and infrastructure, but that there are a number of papers that deal with the evolution of large scale socio-technical systems.

Boero, Castellani & Squazzoni (2004), selected as most relevant model in the study, describe an agent-based model of a supply chain in which the agents are modelled as rational decision making entities. A key limitation, however, is that in this approach there is no explicit way to model the technological units and the network structure. This means it is not possible to perform experiments in which the physical network is adjusted independently from the social network.

Finally, Nikolic (2009) concludes that there is not yet a modelling framework suitable for studying the co-evolution of large scale socio-technical systems. Still, the study confirmed the idea that agent-based modelling is the way forward and that the theoretical components needed to create a multi-formalism approach are already present. He then proceeds to describe an evolutionary approach to modelling the evolution[17] of such systems, also based on Nikolic, Dijkema & van Dam (2009).

2.4.7 Conclusions on keywords

From the texts discussed so far in Section 2.4, only Nikolic (2009) uses "socio-technical" explicitly as a keyword[18]. Still, the other papers *do* deal with socio-technical systems without qualifying them as such: the reference to this fact is only clear from the way the application domain is interpreted. No alternative terms to indicate the true nature of these systems (and the fact that they include both technical and social elements that are closely intertwined) have been found.

To continue the search for an approach that meets the requirements set in this thesis, it is worth exploring the spectrum of modelling approaches used in one specific application domain. The large majority of the relevant papers listed above deal with *energy systems*. Not all models used in this field will include both social and technical views[19], but the field can be considered as a typical socio-technical domain. A search for agent-based approaches limited to this domain only, might therefore result in finding one that meets the requirements.

Next, the results of this search are presented.

2.4.8 Agent-based approaches for socio-technical systems in the energy domain

A search for *agent-based model** limited to the scope of *energy* as a research domain, resulted in exactly fifty papers (see Table A.3). From this selection, nearly half (23 out of 50) deal with energy or power markets. Different trading arrangements can be tested (e.g. Bower & Bunn 2000), the effects of different levels of market concentration (e.g. Frezzi, Garcés & Haubrich 2007), showing different strategies of monopolies (e.g. Tellidou & Bakirtzis 2007) and different market rules (e.g. Liu, Yang & Gan 2005). Also the effects

[17]Note that it is not just a model of the system but of the evolution of the system, thus following Epstein's (1999) quote "If you did not grow it, you did not explain its emergence".

[18]It should be stressed, however, that he is a colleague from the same research group as the author of this thesis and that the choice for the keyword "socio-technical" is directly linked to a shared vision on such systems within the faculty of Technology, policy and management.

[19]However, since agent-based approaches in general are useful for including socio-technical elements it is not unlikely that an agent-based application will actually include these aspects in the model.

of CO_2 emission trading on power markets can be studied (e.g. Weidlich & Veit 2008a). Overview papers of agent-based applications in energy can be found in Yu & Liu (2008) and for agent-based models in power markets and computational economics in Lincoln, Galloway, Burt & McDonald (2006), Weidlich & Veit (2008b) and Yuan, Ding & Hu (2005).

Not only power markets are studied. Other examples include the modelling of a DC motor for fault identification (Awadallah & Morcos 2006), the specific situation of restoration of a power network after a black-out (Liu, Chen, Shen & Fan 2005) and investment decisions for generation expansion (Botterud, Mahalik, Veselka, Ryu & Sohn 2007, Ortega-Vazquez & Kirschen 2008). From the papers listed in Table A.3, a majority considers the system as socio-technical system. In papers on power market models the market behaviour of the various actors is modelled, but no detailed model of the technical components (e.g. power plants, switches or transmission cables) is needed and often not included.

Modelling platforms for energy systems are presented in Bunn & Martoccia (2008), Morais, Cardoso, Khodr, Praça & Vale (2008), Ortega-Vazquez & Kirschen (2008) and Thimmapuram, Veselka, Koritarov, Vilela, Pereira & Silva (2008). These four approaches are discussed below.

Bunn & Martoccia (2008) describe a market simulation platform. The agents represent companies who seek profit through interaction with the market and by learning through adjusting offers based on the previous day, with simple computational learning algorithm. The model can be used to experiment with strategic behaviour and market power (control of large market share), for given pricing and demand profiles.

Morais et al. (2008) present the Multi-Agent Simulator of Competitive Electricity Markets (MASCEM). It is used to model an energy market, more specifically that of virtual power producers composed of multiple households with distributed generation technologies. Agents represent sellers, buyers, system operator and regulators, among others, but again no explicit representation of the technical system even though technical characteristics influence the behaviour of the agents. A virtual power producer is seen as a coalition of agents and it acts both as buyers and sellers.

Ortega-Vazquez & Kirschen (2008) look at a toolbox for modelling generation expansion. Investment decisions are made by each generation company (an "agent"), based on available market data (e.g. demand and fuel costs) with the goal to maximise its own profit. Equations are used for the decision problem but the cornerstones of the model are the individuals. These equations describe the market clearing process, bidding functions, etc. and the model is implemented in MATLAB.

Finally, Thimmapuram et al. (2008) presents the Electricity Market Complex Adaptive System (EMCAS[20]) which is extended with a hydro-thermal coordination model. The hydro-thermal model optimises the operation of the plant and reservoirs. The integration between the two models is done through information exchange of projected "hydro conditions". It is a very specific model, needed to handle the complexity of hydro power plants but it makes it not suitable for other domains. The market model is similar to the market models presented above.

These four modelling platforms or architectures fully concentrate on energy markets and economic decision making and cannot contain the physical infrastructure. This

[20]See also Cirillo, Thimmapuram, Veselka, Koritarov, Conzelmann, Macal, Boyd, North, Overbye & Cheng (2006).

means that they cannot be used to do experiments with disturbances in the physical network and responses to this, or with the evolution of the network and are limited to the electricity domain.

2.4.9 Conclusions part 2

The papers discussed in part 2 of this literature study all deal with systems considered to be socio-technical following the definition from Section 1.2.5 and they are all applied on a domain closely related to that of this thesis. Search was performed through work of peers, known conferences and within a specific application domain, but no modelling approaches were found that meet all requirements, yet some come close and foundations of those can be used.

The first conclusions that has to be drawn is that agent-based approaches are becoming more and more widely used. Where the papers on modelling socio-technical systems (Table A.1) still included many with a publication date more than ten years ago, the papers on agent-based modelling of electricity systems (Table A.3) are heavily concentrated on work done over the last three years only. The same can be said about the other work discussed in the rest of Section 2.4. Almost all relevant work that was found was published in 2007 or 2008. This also means it is a field still under development with new ideas, theories and models being promoted.

Many researchers are working on models that can support decision makers to deal with challenges in socio-technical systems comparable to those posed in this thesis. Specifically Dignum et al. (2008), Keirstead et al. (2009) and Hodge et al. (2008) have similar aims and similar suggestions for solutions. The use of ontologies to bring together multiple fields, as used by Keirstead et al. (2009), for example, is extremely valuable and something that will also be used in the framework presented in the next chapter. However, none of the approaches found are explicit in their definition of a socio-technical system as consisting of social and physical subsystems that are interrelated and defined independently from one another.

Most work focusses either on the definition of a multi-agent control system (e.g. Negenborn 2007, Hines & Talukdar 2007) or on the modelling of behaviour of individuals (e.g. Dignum et al. 2008, Keirstead et al. 2009), but Dignum et al. (2008) is too detailed in trying to replicate realistic human behaviour, as for agents representing organisations an introvert character does not play a role. It might be a possible direction for future work when including different management styles of companies, etc. Hodge et al. (2008) comes closest with his research questions, and his ideas closely match those discussed in Chapter 1. However, it is not clear how well this approach would work in other domains and when combining models of different infrastructure systems.

Finally, Ortega-Vazquez & Kirschen (2008) demonstrate that equations can also be used to model decision making of individuals in MATLAB (see also Figure 2.1). Approaches using specific agent software should be compared with this paradigm too.

2.5 Conclusions

There are many modelling approaches for socio-technical systems, however none fully satisfy the conditions set out in Chapter 1 even though some come close, as discussed in Sections 2.2.4 and 2.4.9. The following key lessons are learnt from this study:

- Socio-technical systems modelling is done in various applications domains, ranging from software requirements engineering to evacuation planning and from crisis management to sustainable development. The backgrounds of researchers is incredibly diverse, but with a strong emphasis on software engineering. This is because it is a popular application domain but also because software engineers use their skills on other domains.

- Agent-based models are suitable for modelling socio-technical systems, with applications in various domains. From all approaches that can be used, agent-based ones are predominant and most promising.

- The distinction between agent-based and non-agent-based approaches is often not clear and people use different interpretations of the concept 'agent', leading to different types of models. Models that focus on individuals as modelling elements and that mostly use algorithms as language to define behaviour are considered as agent-based models in this thesis, but this is not a black-and-white distinction. Models built up from equations can be — and have been — used to model individual decision making too.

- Different types of models will have to be combined to capture the full complexity of real systems.

- A shared language, formalised in an ontology, is needed to bring different aspects of the system together and to connect different models or modelling approaches. This shared language also helps when communicating with people from different domains.

- System level changes caused by individual behaviour is important and can be found in all work studied here.

- Even for very different application domains, similar challenges can be observed. Most work done in this field aims at being generic beyond the original field and only few people (mostly in the energy domain) target the work at one specific domain only.

- All relevant work is very recent and many papers present work-in-progress or conceptual models only. At the start of the research published in this thesis there were almost no publications on the subject yet.

There is still an open challenge in dealing with the socio-technical complexity. The distinction between social and technical elements in the model should be made explicit, so experiments making variations in either one of the networks can be performed. This thesis aims, by building upon the existing body of knowledge, to contribute to finding an approach that meets the criteria set in Chapter 1 so strategic makers can successfully be supported when dealing with the challenges that arise from the socio-technical complexity in the system. Furthermore, the advantages of agent-based systems, as listed by many authors, should be critically viewed and compared with other modelling paradigms to enable modellers to make the right — well informed — choice when being faced with a new problem.

2.6 Research questions revisited

Following the conclusions of this chapter three additional research questions and a refinement of one of the questions presented in Section 1.6 are added to better focus the work of this thesis:

- How can an ontology be created that describes the relevant elements of socio-technical infrastructure systems, that can be applied to different domains and refined for specific cases?
- Which concepts should an ontology for socio-technical systems contain?
- What are the advantages of agent-based modelling compared to other computational modelling paradigms?
- How can agent-based models support decision makers?

These questions deal with the two open challenges: fully capturing the socio-technical complexity by combining models of these subsystems and gaining insight in the real advantages of agent-based modelling when compared to alternative approaches.

For the first challenge, the concept of an *ontology* will prove to be a key concept in the rest of this thesis. It enables the modeller to create a formal description of the concepts in a domain and share these. The ontology is not only machine readable, but also machine understandable (i.e. the computer can reason about the concepts and how they relate to each other). Section 3.3 explains this in more detail. A framework has to be developed that brings the social and physical systems together using these concepts. The second challenge requires a structured comparison of different modelling approaches and their advantages and use. The hypothesis here is that agent-based modelling is the best choice for modelling socio-technical systems, but this has to be critically evaluated afterwards.

Chapters 3 to 7 address the research questions formulated in this section as well as those from Section 1.6.

Chapter 3

Framework for the development of agent-based models of socio-technical systems

3.1 Introduction

Following the conclusion of Chapter 2 that agent-based modelling is a promising approach for dealing with the challenges that arise from socio-technical complexity and that, to support modellers, a framework for the development of agent-based models of socio-technical systems is preferred, such a generic framework has been developed. This framework aims at supporting the modeller in quickly setting up new applications by re-using building blocks as well as supporting connecting existing models to one another.

Kaelbling (1991) defines an *architecture* as a "specific collection of software (or hardware) modules, typically designated by boxes with arrows indicating the data and control flow among the modules. A more abstract view of an architecture is as a general methodology for designing particular modular decompositions for particular tasks". His "abstract view" of an architecture covers a key element of the work presented in this chapter, but the aim is not, however, to develop a new *agent architecture*. An agent architecture is defined by Maes (1991) as a "particular methodology for building [agents]. It specifies how [...] the agent can be decomposed into the construction of a set of component modules and how these modules should be made to interact. The total set of modules and their interactions has to provide an answer to the question of how the sensor data and the current internal state of the agent determine the actions [...] and future internal state of the agent. An architecture encompasses techniques and algorithms that support this methodology". An already existing agent architecture is re-used[1] instead of building a new architecture to handle the basics of scheduling and message passing, for example.

When speaking about software or model development, a *framework* is often defined as a set of classes and code libraries. Or in the words of Gamma, Helm, Johnson &

[1] In this thesis the Repast agent simulation toolkit (North, Collier & Vos 2006, Nikolai & Madey 2009) is used to develop models, but the framework does not depend on it.

Vlissides (1995)[2]: A "framework is a set of cooperating classes that make up a reusable design for a specific class of software". Furthermore, they say that "a framework provides architectural guidance by partitioning the design into abstract classes and defining their responsibilities and collaborations. A developer customises the framework to a particular application by subclassing and composing instances of framework classes".

The framework presented in this chapter is a *software framework* following Gamma et al.'s (1995) definition with a set of *modelling steps* to build models using this software framework. Furthermore, the *approach* to develop such a framework is addressed. In other words, this chapter presents an approach as well as the result of this effort, together with guidelines on how to use the software framework and modelling steps to develop models.

The development of the framework is an iterative process involving modellers from different disciplines and backgrounds. The framework evolved over time through experience gained from application of the framework to various case studies. In addition to a description of the procedures followed, this chapter presents the current result of this work. In Chapter 5 the process itself will be analysed to evaluate the current state of the development (how generic the framework is and if it is finished or still needs more development) and to learn lessons from the work done on the framework so far.

This chapter describes the process towards the framework, but also the results of the development: it aims to provide a practical framework that can be used for model development for socio-technical systems, as well as an approach to build a similar framework in other domains.

The rest of this chapter is structured as follows. First, in Section 3.2 a set of requirements for the design of the framework is formulated, to make it applicable to solving the problems posed in Section 1.1. The cornerstone of the framework consists of an ontology for socio-technical systems. First, ontologies in general are discussed in Section 3.3. Next, Section 3.4 describes the approach followed to come to the framework. The ontology for socio-technical systems, developed using the aforementioned approach, is presented in Section 3.5. Section 3.6 deals with the steps required to build a model with this framework. Finally, in Section 3.7, concluding remarks about the framework are made, including the core elements, applicability and rules of thumb for usability.

3.2 Requirements

In Section 1.4 the requirements for the framework were presented, based on the need for decision support for socio-technical systems as identified in 1.3.4. Firstly, the main requirement is that both the physical and social reality of the system can be captured, including their interactions with one another and the external dynamic environment. To be able to support modellers and decision makers a flexibility is desired to experiment with:

- different configurations of the social network with same physical network;
- different configurations of the physical network with same social network and
- different configurations of *both* social and physical networks.

[2]Sometimes referred to as the *Gang of Four* of object-oriented programming.

They can be summed up as *interoperability* and *inter-connectivity*. This type of flexibility can be obtained with a modelling framework based on modularity and shared interfaces between the modules[3]. Modular models can be seen as consisting of *building blocks* that can be connected and re-used. To be able to define and use these building blocks, a language to describe the components and a language for the components to interact is required.

Furthermore, the models should be able to connect new parts of the model with existing elements. This is the case both when making extensions of models and when dealing with the interactions between infrastructures, meaning that (elements of) models of infrastructures have to be connected. Modularity also allows the modeller to connect models of different infrastructures via shared interfaces and re-use model components in other projects. This style of modelling should explicitly be supported by the framework as it is a key challenge for the problem owners.

The functional requirements can be summed up as follows:

- Support a wide range of socio-technical infrastructure systems including petrochemical clusters, energy networks, freight transport and supply chains.
- Flexibility for experiments with varying configurations of the social and technical networks, either one or both.
- Full modularity, which results from the requirement of flexibility, but also offers re-usability.
- Easy to use by modellers, including those not involved in the development of the framework.
- Extendibility without losing backward compatibility, so that case-specific aspects can be added without causing older models to stop functioning.
- Easy to explain to new modellers, but especially to the problem owner and other stakeholders in the case studies.

3.3 Ontologies

As said in Section 3.2, a language to describe the components and a language for the components to interact is required. Agents not only need a communication language and a standard interface, but, in order to interact, they also need a shared model of the world (Aldea et al. 2004, Garcia-Flores & Wang 2002). In the framework proposed in this chapter an *ontology* is used for both the interface and as a shared world model, forming the cornerstone of the framework. Before presenting the ontology for socio-technical systems in Section 3.5, this section discusses what an ontology is (Section 3.3.1), how to decompose a system to create a useful description and ontology (Section 3.3.4) and, finally, which tools can be used in the development process (Section 3.3.5).

[3]Following Bradshaw's (1996) statement that agent-based systems should be constructed in a modular way so that all parts are replaceable.

3.3.1 What is an ontology?

When two agents in a model communicate about certain concepts, it is critical that they give the same interpretation to the meaning and use of these concepts[4]. Therefore it is of the utmost importance to unambiguously specify each concept and its meaning. In the artificial intelligence community *ontologies* are developed as a useful means of knowledge representation. Ontologies are formal descriptions of entities and their properties, relationships, constraints and behaviour, that are not only machine-readable but also machine-understandable. Communication between agents, be it in the form of messaging using well specified protocols (see van Dam 2002) or directly calling methods of the other agent, can only be meaningful when the *interface* is clearly defined. An ontology contains explicit formal specifications of the terms in the domain and the relations among them. In other words: it is a formal specification of a conceptualisation (Gruber 1993).

The foundation of an ontology is most frequently defined by a number of 'is a' relationships and 'has a' relationships. 'A is a B' means that A and B are both classes of Things and that all things that are A are also a member of class B. 'B has a c' means that c is a property of B. Because A is a B, all properties that a B can have are also applicable to an A, so automatically c is also a property of A. An example in Section 3.3.2 should clarify this.

An ontology consists of classes (abstract specification of concepts, with their possible properties) and instances (concrete specification of concepts with specific properties). Classes provide the abstract[5] specification of the concepts and their properties. An instance is a single identifiable object within the limits of the scope of the model, belonging to a class that is formalised in the ontology.

In this view a *class* is nothing else but a generalisation of a number of instances that the modeller chose to put together. An *instance* is a single identifiable object within the limits of the scope of the model, belonging to a class that is formalised in the ontology. The class is abstract, whereas the instance is concrete. To illustrate what these definitions mean in practice an example is presented next.

3.3.2 An example: the girl with a pearl earring

To give an example of a simple ontology, the domain of *art* is considered here. There are many different types of art works: A Painting 'is a' WorkOfArt, a Sculpture 'is a' WorkOfArt, one may also consider that Furniture 'is a' WorkOfArt, etc. Additionally, it could be defined that a Painter 'is a' Human, a Sculptor 'is a' Human, a photographer 'is a' Human, etc. Once the concepts of 'art' and 'human' have been defined, a 'has a' relationship can be formalised: A WorkOfArt 'has a' artist, and the artist 'is a' Human. Because every Painting has been defined as a WorkOfArt, it can be deduced that every Painting also has an artist. In similar fashion, one can say about a WorkOfArt that it 'has a' location and that it 'has a' yearOfCompletion, etc.

In the abstract language formalised so far only classes of art in general have been considered, but no specific piece of art. With the class definitions one can now, for example, talk about the painting the "Girl with a pearl earring". This is a specific object of which

[4]This also applies to the actors which the agents represent, even though it may be less strict.

[5]Note that when speaking about classes only, one can also distinguish *abstract* and *concrete classes*, where abstract classes are those that cannot have instances and concrete classes can have instances. This same terminology is used in the Protégé tool discussed later.

there is only one in the real world[6]. When two art-lovers talk about this specific painting they have a shared understanding that the other means the work painted circa 1665 by Johannes Vermeer in Delft, and which currently hangs in the Mauritshuis museum in the Hague. They would not confuse the painter for the film maker when refering to the artist, thanks to a shared ontology.

The key to building ontologies is describing concepts using already defined concepts, such as in the example above: because the concept of an Artist was defined, it can be used to define one of the properties of a WorkOfArt.

3.3.3 Why use an ontology?

If there is a standard way of building an agent system, developers can use this proven approach. It is also important to standardise the environment in which agents are running so that agent platforms can pass on messages and share information about agents (e.g. descriptions of agents, agent locations). This allows communication between agents running on different types of platforms (van Dam 2002).

Ontologies are not only useful for communication between agents, but also for sharing knowledge between modellers, domain experts and users. No misunderstanding should be possible so a shared language is needed. Which concepts play a role depends on the goals of the research and the problem owner should specify what type of questions should be answered by the simulation.

When it comes to implementation of a model, an ontology forms the basis of the class structure for object-oriented software implementation. The 'is a' relationship is coded as the subclass relationship in class descriptions and the 'has a' provides information on the properties of the class and the possible values. This is similar to, for example, the Java programming language.

An ontology provides an *interface* definition between objects in the model implementation. If properties defined in the ontology for a specific class are known, it means this information can be exchanged. In the case that two parts of the model communicate about a painting, to revisit the example from Section 3.3.2, it is known that since it has been defined that a painting is a work of art and a work of art has an artist who created it, that the artist can be requested. The knowledge rules (i.e. the decision making rules of agents) to implement the behaviour of the agents can then also be expressed in these formalised concepts.

Finally, but most importantly, the definition of a system that is being modelled can be expressed in concepts from an ontology and stored in a *knowledge base*, enabling automated generation of models from the instances in the knowledge base.

To summarise, in the model implementation phase the ontology offers:

- A class structure;
- An interface and
- A language for system definition.

Ontologies are even more powerful when they can be re-used. To be able to do this, it is important to use a generic description as much as possible. This does not

[6] Note that there is also a book and a film (based on the book) with the same name, making clear that the context is important for how things are understood.

only make it possible to re-use domain and expert knowledge, but also to re-use source code. This is essential in the approach presented in this chapter: by specifying a problem using previously formalised generic concepts, already implemented building blocks can be re-used. In other words: ontologies facilitate re-use, sharing and interoperability of agent-based models.

3.3.4 System decomposition method

The goal of system decomposition is to identify the internal structure of a system to be observed in such a manner that the analysis of the system becomes possible. This means that a system is not only considered to be a collection of actors and interactions that exist in the current situation of the system, but that it is also taken into account which elements might change over time and at what speed, with system level emergence as a result. The output of this approach is as follows:

- A list of important system elements.
- A list of concepts that describe the relationship between the system elements.
- A specification of the communication language used between the actors, which includes semantics.
- A specification of which elements are variable over time.

The process decomposition method used here consists of three phases which are introduced in van Dam, Nikolic, Lukszo & Dijkema (2006): inventory, structuring, and formalisation. In Nikolic (2009) the social processes of the first two phases are dealt with in more detail. The system decomposition method has also been applied to a different domain than that of infrastructure systems, namely that of development aid (Rammelt, Nikolic, Boes & van Dam 2005). This illustrates the wide applicability of the approach, but also demonstrated that different viewpoints from different perspectives result in different conceptualisations of a system. This is a key challenge when incorporating both social and physical realities as well as different infrastructural sectors in a shared system specification.

The final phase of the system decomposition method is the formalisation in ontologies. After the inventory and structuring phases of the system decomposition have been completed, an ontology can be created in order to strictly formalise the domain and to enable the system description to be generalised (Noy & McGuinness 2001). A body of formally represented knowledge is based on a conceptualisation: the objects, concepts, and other entities that are assumed to exist within the system boundaries and the relationships that hold among them. This is exactly what follows from the system decomposition process.

3.3.5 Software tools for ontology development

Ontology development is, as said before, a shared effort. By definition people from multiple disciplines have to work together in order to create an ontology that can be widely used. To support this development process a number of software tools are used.

The ontology development tool used for the research performed for this thesis is Protégé, which is developed by Stanford Center for Biomedical Informatics Research at the

Stanford University School of Medicine in California, USA (Gennari, Musen, Fergerson, Grosso, Crubezy, Eriksson, Noy & Tu 2003). It was originally designed to be used in the medical domain, but it is being widely used outside this field too, in disciplines ranging from art to engineering. The tool has a strong community and a closely involved user base, providing both clear examples and case studies as well as support in practical problems new users may encounter. Protégé supports many different standards languages for storing ontologies, including W3C's Web Ontology Language (OWL)[7] and Resource Description Framework Schema (RDFS) (Gennari et al. 2003) which are both based on XML. For the work in this thesis a *Frames* ontology was used (Wang, Noy, Rector, Musen, Redmond, Rubin, Tu, Tudorache, Drummond, Horridge & Seidenberg 2006).

Protégé uses a Graphical User Interface (GUI) for entering class definitions. In addition it provides a GUI for knowledge acquisition using user-defined forms for entering information about instances. It is free and open-source software, making it ideal for scientific research purposes as well as integration with other tools (as will be demonstrated in Section 3.5.5). Alternative tools for ontology development and maintenance of knowledge stored in the knowledge base include OilEd (Bechhofer, Horrocks, Goble & Stevens 2001) and OntoEdit (Sure, Erdmann, Angele, Staab, Studer & Wenke 2001), but Protégé was chosen for ontology development: the strong user base and full community support together with the fact that it is open-source make it the preferred tool for this thesis.

To share the latest version among all users and developers of the ontology as well as to keep track of the different versions over time, version control software is used. A central Subversion (SVN) server[8] hosts the files while those that need access to the ontology install an SVN client on their computer to be able to checkout the recent version and commit updates back to the server.

On the user-end, client software called *Tortoise* (for MS Windows) or *svn* (for Linux) is used combined with Subclipse for use in Eclipse IDE, but any SVN client can be employed to access the repository[9]. Finally, a web interface can be used to access the latest version of the ontology from any web browser. The use of version control software and storing the ontology on a central server make it possible to access, use and edit a shared ontology for those who have authentication. While there are other means to establish this, such as using a shared database, SVN has proven to be easily incorporated into the work flow. Which tool is used is irrelevant, but for joint development of a shared ontology some form of cooperation software is required. The version control facility is also used in Chapter 5 to analyse the development of the ontology.

3.4 Approach to the development of a framework

In this section a structured approach to modelling socio-technical systems, with a focus on infrastructure systems, is presented. The approach can be used to set up new models of infrastructures by following a number of steps and re-using already existing building blocks from other models: it can be seen as a "model factory" (van Dam & Lukszo 2009).

[7]See http://www.w3.org/TR/owl-guide/.

[8]Until 2007 a Concurrent Versions System (CVS) Server was used for the ontology development, but it was replaced by the more advanced and easier to use SVN.

[9]Note that Protégé does not include support for version control, but other tools can be used to synchronise the files with the repository, for example from the file manager or from the IDE.

It aims at offering support to modellers in building and connecting models, but also to help other parties (including the problem owner) to be involved in the modelling process and to better understand the results.

A typical rule of thumb in software development, or good old programmer's wisdom, is "if it has not been tested, it does not work". This can be interpreted as "software that has not been reused, is not reusable". A framework can only be called re-usable when it has actually been re-used in practice.

A generic framework for modelling should be developed through gradual changes and iterations instead of making one definitive design and implementing that design. Each iteration follows an application of the framework to a new domain and the lessons learnt from this feed back into the generic framework. This is repeated several times before a state can be reached in which the framework can be called generic (to a certain extent). Each new case study that is executed provides the opportunity to add new re-usable components to the library (see Figure 3.1).

Generic classes are those classes that can be shared between various models. However, it is always possible to extend the class structure by adding new subclasses to already existing classes or all new classes can be created. The structure of an ontology can be expanded for specific purposes, but changing the structure is problematic because it means models based on earlier versions have to be adjusted as well.

See Figure 3.2 for an illustration of a number of classes for a case study that are added as a subclass to generic classes. When these case-specific classes are in turn shared by future models, they can become part of the generic classes, as illustrated in Figure 3.1. It should be stressed that in practice there is not a clear line between what is generic and what is specific, as all classes are in principle shared. However, there are many classes that have been developed for one specific model only and in this thesis they are not considered to be part of the set of generic classes. The generic ontology as presented in this chapter therefore focusses on those elements that two or more models use. This provides the main structure that future models are based on, as well.

Before the development cycle from Figure 3.1 was started, an initial state was created. This step is explained in Section 3.4.1. Afterwards the current state of the framework is presented.

3.4.1 Initial state

The initial state of the framework was created based on experiences of building a proof-of-concept model of an industrial cluster. In this test-case a *chocolate production chain*

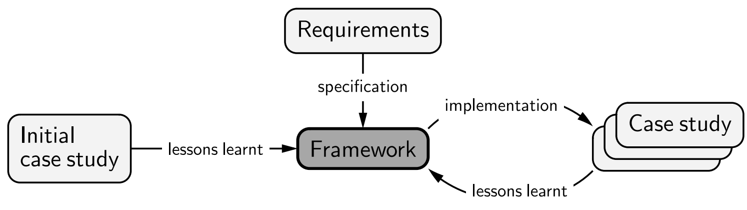


Figure 3.1 – *Approach of the framework development*

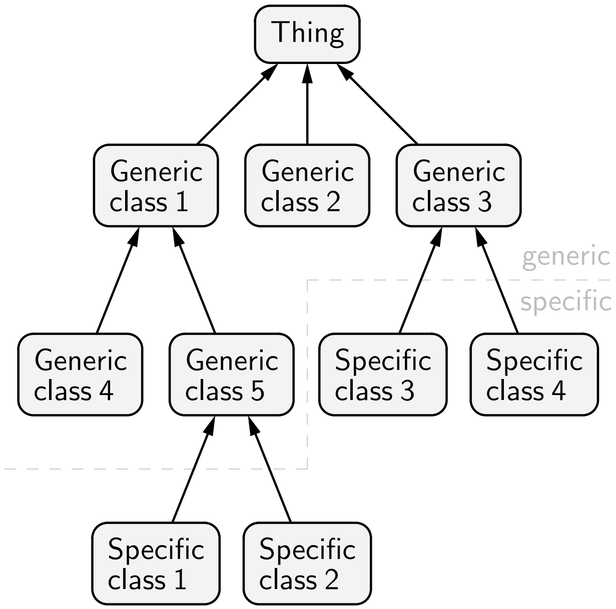


Figure 3.2 – *The border between generic and case-specific classes (based on van Dam & Lukszo 2006)*

was imagined as representing clusters in the (petro-)chemical industry. Instead of trading chemicals such as crude oil, ethylene, chlorine or naphtha and selling plastics or diesel, the factories here need to buy cocoa and raisins and sell chocolate bars to the world market. This model was originally designed as a game to play with a number of students or domain experts to raise awareness of the strong interdependency between their decisions (See Box 2).

Box 2 — Chocolate game description

The goal of the *chocolate game* (van Dam, Nikolic, Lukszo & Dijkema 2006) is to make the game players experience the complexity of the system and to create awareness about the need for system-level thinking in the path towards sustainability. Players act as individuals, representing an industrial plant or company. The game-leader manages the changes the in the "external world". The actors represented by the players may be hierarchically organised, with explicit cooperation or competition goals. The model of reality is a large system formed by the participants themselves with some upper and lower boundaries and external influences.

In the game different roles can be identified: producers of certain intermediates, producers of end products, transporters and a world market. The world market can be used to steer the behaviour of the players by changing the prices of products, and is controlled by the game leader. All producers can buy any product from the world market and sell anything to the world market, but everything has its price. Producers can also trade products with other players for which they have to negotiate the price and several other conditions, such as the duration of the contract and what happens if one of the players does not hold to his or her part of the deal. Players were free to come up with their own conditions as long as they were described using the terminology formalised in the game.

Producers buy products (either raw materials or intermediates, depending on their role) and they have a certain technology that allows them to turn these into another type of product (e.g. make a bar of "raisin dream" out of "processed dark chocolate" and "processed raisins", or transform "raw peanuts" into "processed peanuts"). One of the most important concepts of the game (and another main reason for using chocolate production as an analogy for chemical production) was the fun factor. Players have to enjoy themselves to feel involved in the game. To do this the game uses real ingredients that have to be processed by the players. A player buying a batch of "raw peanuts", for example, receives peanuts that still have to be peeled in the production step, resulting in a number of "processed peanuts".

Transport players are responsible for transporting goods from one player to another as well as to and from the world market. Contracts were also needed to ensure delivery. One of the main game rules is that no products can leave the tables except in the transport unit of a transport player. This emphasises the dependency of industry and infrastructure. The waste that is the result of the processing step is also in fact a good that has to be transported to the landfill (i.e. the garbage bin in the corner of the room). Again, the rule applies that no products can leave the table without a transporter.

Before the start of the game each player receives a personal manual with the specific game rules for his or her role. After a short introduction and the initial loading, the game is played for ninety minutes with a group of about fifteen players. In this game, one scenario was played: after a certain amount of time the prices on the world market of the "raisin dream" bar will double ("because of increasing demand after a new marketing campaign"). When the hype is over, the prices will drop again. The goal of this scenario was to see the change of behaviour of the players and make them realize that they are dependent on each other. Players acknowledged that they became more aware of the interdependencies between the different actors in the supply chain, including the link between production and transportation.

After the game had been played a number of times it was decided to implement an agent-based model based on the same system (van Dam, Nikolic, Lukszo & Dijkema 2006). The model, based on an analogy with an industrial cluster, was developed to serve as the initial state of the modelling framework for agent-based modelling of socio-technical systems. The software infrastructure, the strategy components (in this case a model of the players' decision making) and the technology components (i.e. the transition processes from one product to another) were developed independently of the domain description, with the aim of allowing re-use and cross-sectoral modelling. A data structure was created for the agent-based implementation of the chocolate game. This data structure is presented in Box 3.

The proof-of-concept model, while simple, demonstrates the use of agent-based models in this domain and illustrates the occurrence of emergence, which could be seen in the prices for the different traded products. Analysis of the model and its output resulted in a deeper understanding of the impact network lay-out has on system behaviour. The agent-based model is revisited in Section 4.4.

Box 3 — Data structure for the chocolate game model

Each bullet in the class structure presented below represents a new class and an indent means the class is a subclass of the class one level higher. Between brackets the properties of each class are listed, with the data type of the property after a colon. A data type listed in square brackets behind a property means multiple values are allowed for this property.

- Objects (classname:string)
  - Goods (ID:Goods ID, amount:int, owner:Agent ID, location:Agent ID)
    - Raw Materials (usedToMake:Classname)
      - Raw Dark Chocolate
      - Raw Peanuts
      - Raw Raisins
    - Intermediates (processedBy:Agent ID, processedFrom:[Goods ID], madeFromType:[Classname], usedForType:[Classname], consistsOf:[Goods ID])
      - Processed Dark Chocolate
      - Processed Peanuts
      - Processed Raisins
    - End Products (producedBy:Agent ID, madeFromType:[Classname], consistsOf:[Goods ID])
      - Dark Dream
      - Peanut Dream
      - Raisin Dream
      - Dream Everything
  - Storage (capacity:Units, amountInStorage:Unit, typeInStorage: Classname)
  - Truck
  - Contract (price:int, signTime:double, startTime:double, endTime:double, nonDeliveryFee:int, buyer:Agent ID, seller:Agent ID, signedByBuyer:boolean, signedBySeller:boolean)
    - Trade Contract (toSell:Classname, amount:int)
    - Transport Contract (from:Agent ID, to:Agent ID, transportCapacity:int, maximumTripDuration:int)
  - Agent ID
  - Goods ID

3.4.1.1 Lessons learnt from the initial state

This model was built without directly re-using elements from other models, but during the design the aim was to allow for the main classes to be re-used when moving on to create more advanced and realistic models for the type of industry on which the proof-of-concept model was based.

For this initial application, some decisions and modelling choices were made that soon proved not to be the most flexible and very limiting. There is no real functional difference between intermediates and end products, and the distinction should be removed. Furthermore, it was learnt that that using discrete products (each with their own unique good identification number) was not the way forward as it resulted in many difficulties related to splitting up discrete items into smaller pieces (e.g. how should the unique product identifier be treated after part of a batch of a certain raw material was processed?) and having to decide which is the smallest unit that will play a role in the model. This turned out to be workable for the chocolate model (and perhaps even preferred, as it allowed product tracking through the value-adding chain), but is not useful when translating the analogy back to, for example, crude oil instead of raw chocolate. The issue of discrete versus continuous manufacturing becomes even more strongly a problem when talking about energy infrastructures such as electricity networks.

The general structure of agents and contracts can be re-used as it proved to be useful both as a concept in the game as well as in the agent-based model. It was mostly in the field of the physical network where changes had to be made. After this initial case the concept of a Flow (See also Section 3.5.2.1) was introduced (van Dam & Lukszo 2006) to deal with these problems. The new way of dealing with material and energy transport proved to be very versatile and can be re-used in all applications since. The use of analogy of a real system proved to be a powerful way to facilitate out of the box thinking of domain experts and to increase understanding of a generic approach (van Dam, Nikolic, Lukszo & Dijkema 2006).

3.4.2 Iterations

To improve the framework and guarantee genericity and broad applicability, a wide variety of cases is conducted over the development cycle (Figure 3.1). In Chapter 4 a number of case studies that helped shape the framework are discussed in more detail, but here it is sufficient to say that the cases are selected based to widen the spectrum of cases for which the framework can be used[10]. At the same time cases were executed by others that were building upon earlier work in a more incremental fashion.

For each new case study, generally executed in series but in a few cases (especially towards the end of the development) in parallel, adjustments were made to the framework based on lessons learnt during the model design and implementation efforts. If required, changes to the generic building blocks were made. Each new application can result in two types of adjustments to the framework:

- Addition or
- Modification

Additions are concepts added to the framework without changes to any of the existing parts of it are required. Modifications include those adjustments that do change the existing part of the framework (e.g. replacing goods that a unique identification number with a transport flow). Both types of adjustments are necessary to further develop the framework, but modifications come at a price. Modifications can have serious consequences for backward-compatibility of the framework. If certain parts of it are changed (including renaming or moving part of the conceptualisation to another class structure) earlier models, and their source code based on the previous version of the framework, might break when they were using this aspect of the ontology. It is therefore important that for closed projects the version number of the ontology that is required for the model is listed.

[10] Recall the motto "software that has not been reused is not reusable".

3.4.3 Stop condition

Since the approach is iterative, an important question is when to terminate this loop. When is the work complete? The answer is possibly that the work is never finished. However, when the framework can be demonstrated to be applicable and useful in the domains mentioned in Section 3.2 then the requirements are said to have been met. This is done in Chapter 4. Furthermore, conclusions about the completeness of the framework can be drawn from studying the development process itself and monitoring the changes required for the development of new models. This is done in Chapter 5 when the growth and evolution of the framework are analysed through its development trajectory.

3.5 Ontology of socio-technical systems

To support the development of models of socio-technical systems, an ontology for this domain has been developed. It contains concepts that are generic to socio-technical systems. Figure 3.3 shows a small fraction of this ontology in which agents (social nodes) and physical systems (physical nodes) are both considered as nodes, with different properties. It shows, for example, that an Agent 'is a' Social Node and that it 'has a' Technology.

The ontology contains formalised concepts, including different types of edges, properties, configurations and labels, which can be used to define a wide range of elements from socio-technical systems. This section introduces the most important concepts that are generic for use in socio-technical systems modelling. The concepts are introduced in graphical format and discussed in more detail in the text.

In figures with class definitions, each class is a box and the arrows show how the classes relate to each other (within the context displayed in the figure). One or more boxes for a class that are shaded grey indicate that they are the focus of the diagram and these classes and their properties are discussed in more detail in the section referring to the figure.

The properties for each class are shown in the left column and for each property the allowed classes are listed in the right column, with the value type in the middle. If the value type in the middle column is singular, it means only one value is allowed for this property (e.g. 'primitive' for 'label' in the class Node in Figure 3.3), while a value type in plural (e.g. 'instances' for 'physicalProperties', also in Figure 3.3) means that multiple values are possible in the class definition. In Box 4 notes on naming conventions used in this section are given.

Box 4 — Notes on naming conventions

The name of an abstract classes start with a capital letter, while the first letter of a property is printed in lower case. If the name of a class consists of more than one word, the so called "CamelBack" notation is used (i.e. there are no spaces in between words and each new word starts with a capital letter). For properties, starting in lower case, the first letter of each next word is capitalised, when applicable.

3.5.1 Nodes

Since socio-technical systems such as infrastructures can be viewed as networks, the main concept is that of a *Node*. Nodes are connected to one another by Edges (See Section

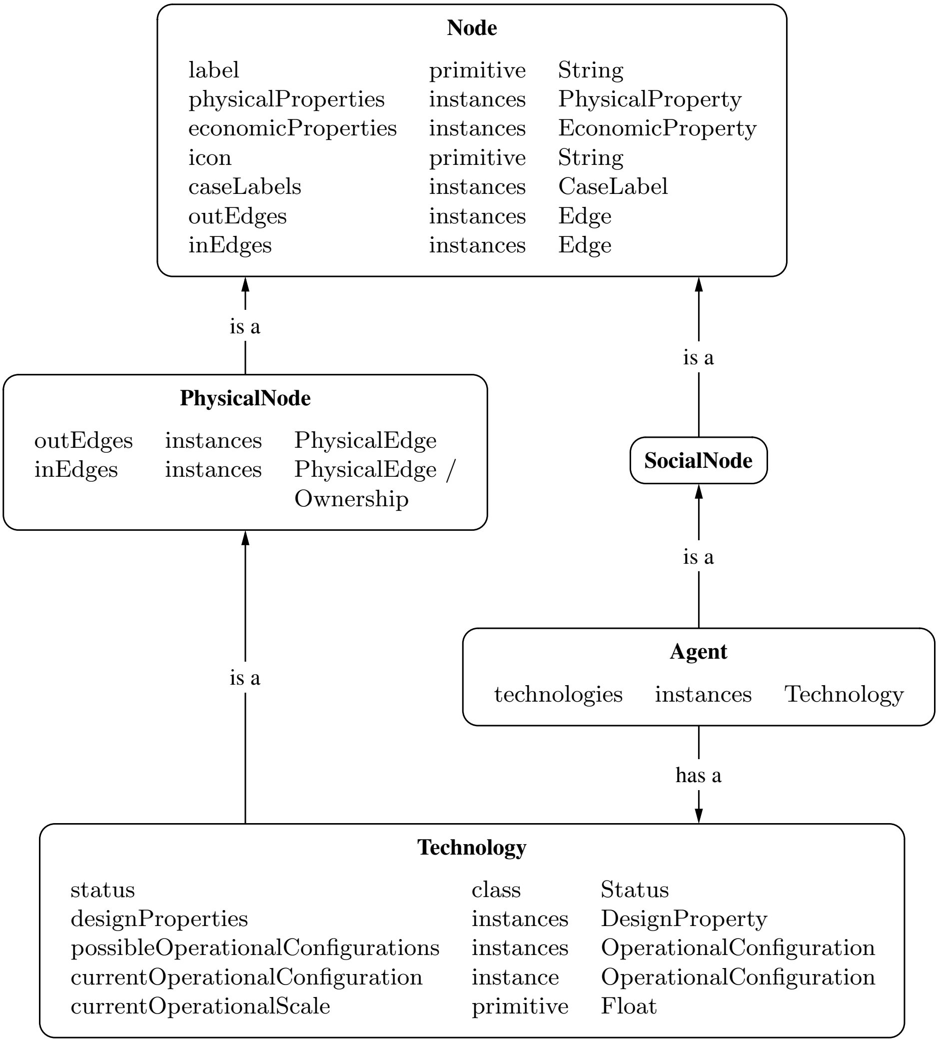


Figure 3.3 – *A fragment of the ontology for socio-technical systems, showing the relationship between different classes of Nodes (Social and Physical) and some of their properties. Agents and Technologies are both nodes that, together, form the socio-technical network. Classes inherit properties from their superclass through the 'is a' relationship where the properties can be seen as 'has a' relationships*

3.5.2). The first distinction of classes of Nodes is that of *SocialNode* and *PhysicalNode* (See Figure 3.4). This distinction is based on Figure 1.1 and follows the requirement that social and technical aspects of the system can be modelled independently of each other.

All Nodes share the following properties:

label A name for this Node, used for identification or display purposes. Labels are of the type String.

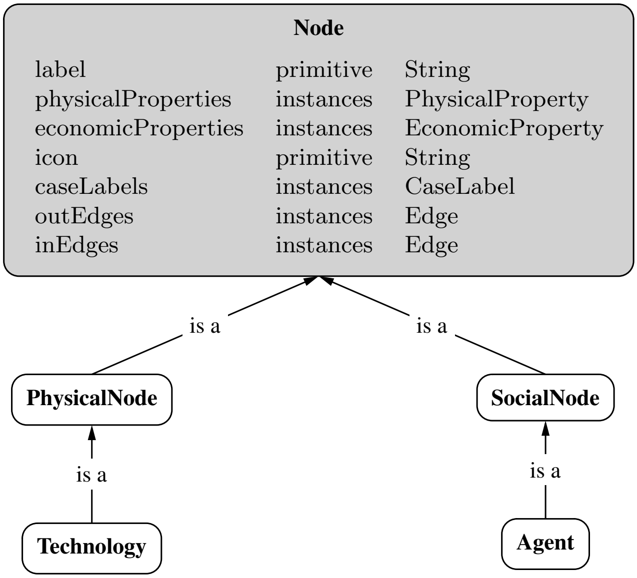


Figure 3.4 – *Nodes in the ontology*

economicProperties Properties related to the economics of the Node. See Section 3.5.3.1.

physicalProperties Properties related to the physical aspects of a Node. See Section 3.5.3.2.

icon The name of an icon used for visualisation of this node during a simulation.

caseLabels One or more labels to indicate for which case studies this node is used. These labels are of the type CaseLabel.

outEdges links going out of this Node. They are of type Edge (See Section 3.5.2)

inEdges links coming into this Node. They are of type Edge (See Section 3.5.2)

Of these properties, label is singular (meaning that a Node can have only one label) while others are plural (meaning that a Node can have multiple caseLabels, economicProperties, etc.).

The main subclasses of Node, SocialNode and PhysicalNode, are introduced in the following two sections.

3.5.1.1 Social nodes

A SocialNode is a Node capable of making decisions about PhysicalNodes. SocialNode inherits all properties of Node, of which it is a subclass and the same applies to the class Agent (see Figure 3.5). This includes the property PhysicalProperties, which is generic for a Node because a SocialNode can also have a Location (which is a PhysicalProperty).

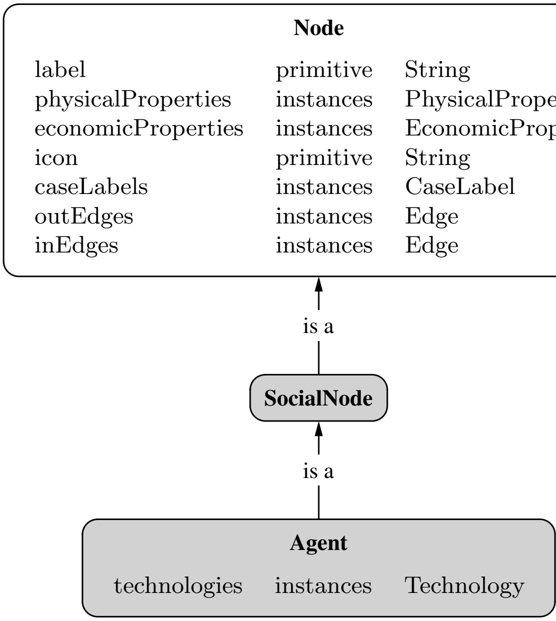


Figure 3.5 – *Agent in the ontology as a subclass of SocialNode*

SocialNode has a subclass Agent, representing an actor in the system. This can be a single person (e.g. an owner of a photovoltaic panel), a group of people (e.g. the operations department) or a whole organisation (e.g. the government). Moreover, Agent has one class-specific property that distinguishes it from its super classes:

technologies a list of Technologies that the Agent owns, controls, maintains, etc.

Strictly speaking the technologies property is not needed because the same information is also stored through OwnershipEdges (see Section 3.5.1.2), but it is still useful to include it in the ontology because it allows easier referencing to the objects owned by the Agent through the editor. In the implementation of Java classes this property is not used directly, but instead it is calculated based on the OwnershipEdges that are in the outEdges property of the Agent. However, because the property is defined in the ontology, it means it is provided as interface for the Agent objects so, independent of how this is implemented, a definition in the ontology allows direct access to this property.

The class SocialNode itself does not have any additional properties compared to the class Node, but it is included to created flexibility to add other types of SocialNodes that are not Agents. One can think of the concept of a Player in an educational game, for example. Because changing the structure of the ontology at a later stage is difficult (as it is not backwards compatible any more after a structural change) it is preferred to already include this flexibility by introducing the concept of a SocialNode. Furthermore, it provides a clear parallel to PhysicalNode, which is at the same level in the class hierarchy as SocialNode.

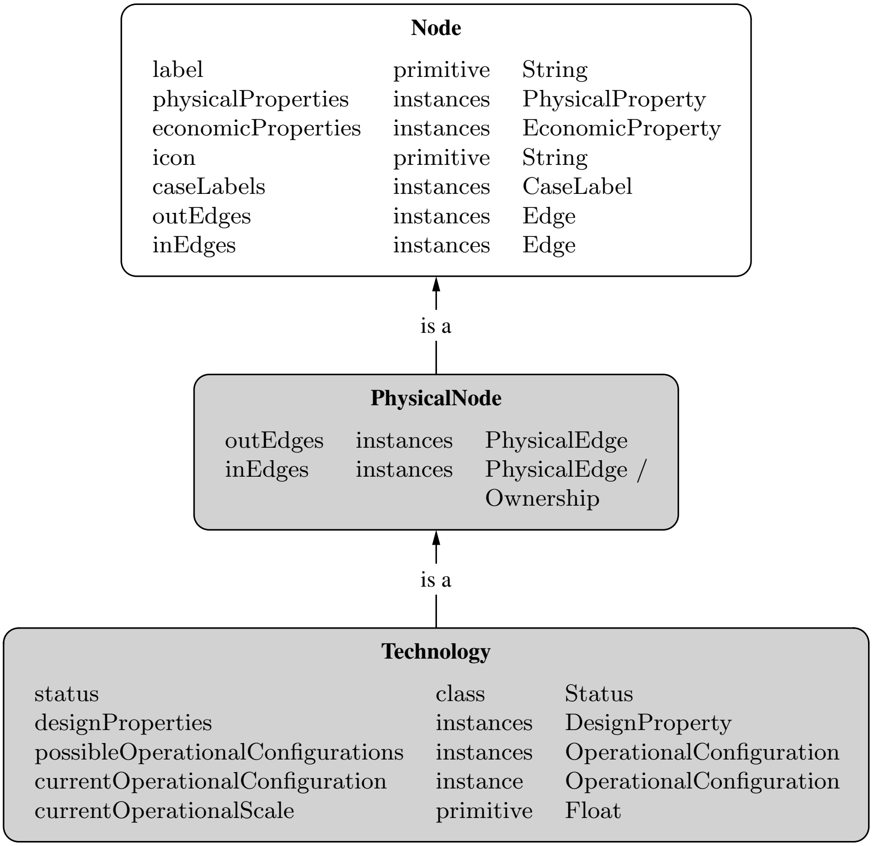


Figure 3.6 – *Technology in the ontology as a subclass of PhysicalNode*

3.5.1.2 Physical nodes

A PhysicalNode represents an element in the physical world, such as an engineered system. For PhysicalNodes a *process system* perspective is followed: in the node a transformation takes place. A PhysicalNode can either be a small unit (e.g. a battery) or a very large system (e.g. a power plant). A subclass of PhysicalNode is Technology (see Figure 3.6). As with Agent as a subclass of SocialNode, this allows later inclusion of different type of PhysicalNodes in addition to what is called Technology here.

The properties InEdges and OutEdges are refined for the class Technology. OutEdge is restricted to the class PhysicalEdge, because PhysicalNodes can only connect to other PhysicalNodes via the physical network (See Section 3.5.2.1). For InEdge the restriction is also PhysicalEdge (for incoming physical flows) but in addition to that it also allows Ownership as InEdge to connect the physical and social network to each other. Ownership is a subclass of SocialEdge (See Section 3.5.2.2). Again, this is done to guarantee that the social and physical networks are separated by making sure a SocialEdge such as a Contract cannot be created between two PhysicalNodes.

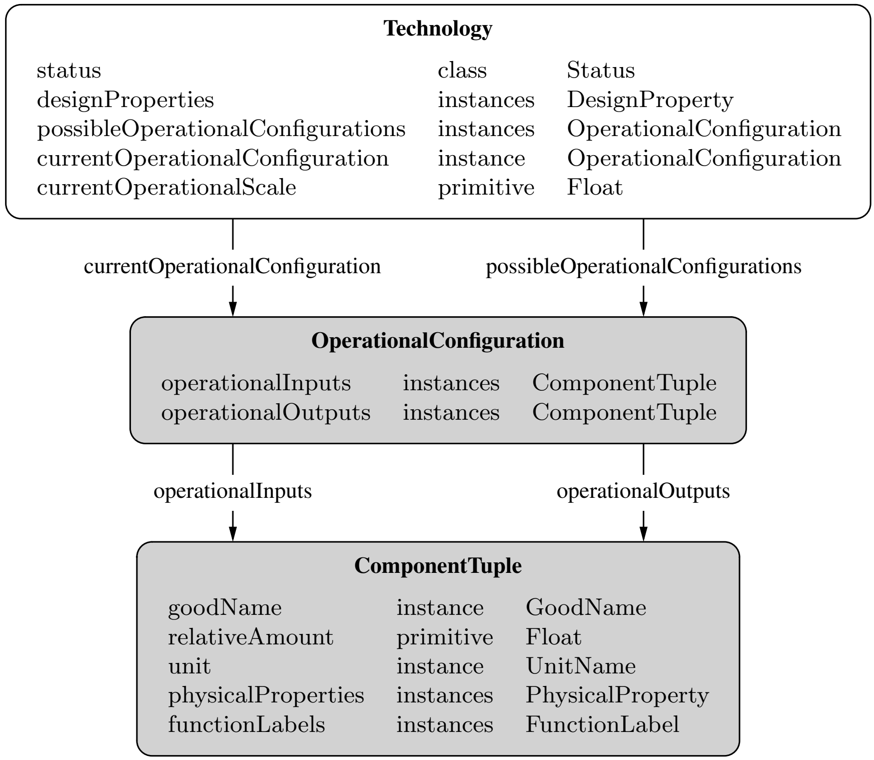


Figure 3.7 – *OperationalConfigurations for the class Technology*

The following properties are defined for the class Technology:

possibleOperationalConfigurations A list of possible configurations for the operation of this Technology. Type is OperationalConfiguration (explained below).

currentOperationalConfiguration The current choice for configuration for the operation of this Technology. The configuration set here should always be one of the set of PossibleOperationalConfigurations.

currentOperationalScale The scale of the operation (i.e. the throughput at which the Technology operates). The relative numbers from the OperationalConfiguration are multiplied by this integer.

designProperties A list of properties related to the design of a physical system. See Section 3.5.3.3.

status The status indicates if the Technical system is under construction, under maintenance or operational.

Of these properties, currentOperationalConfiguration, currentOperationalScale and status are singular while the others are plural.

A key concept in the definition of a Technology is that of an OperationalConfiguration (see Figure 3.7). An OperationalConfiguration is the specification of the connection between input and output of a technical system, precisely defining which GoodNames are at the input (OperationalInputs) and how they are transformed to the output (OperationalOutputs). For each of these inputs or outputs, the following properties can be defined in the form of a ComponentTuple:

goodName The name of the product (from class GoodName) used at the input or output.

relativeAmount The amount of this product needed in relation to the other inputs or the amount of this product produced in relation to the other outputs.

unit The Unit in which the RelativeAmount is expressed.

physicalProperties Other properties of the input, such as required pressure or temperature.

functionLabels Labels to for example indicate if the input is a scarce resource or a ubiquity, or if the output is a primary

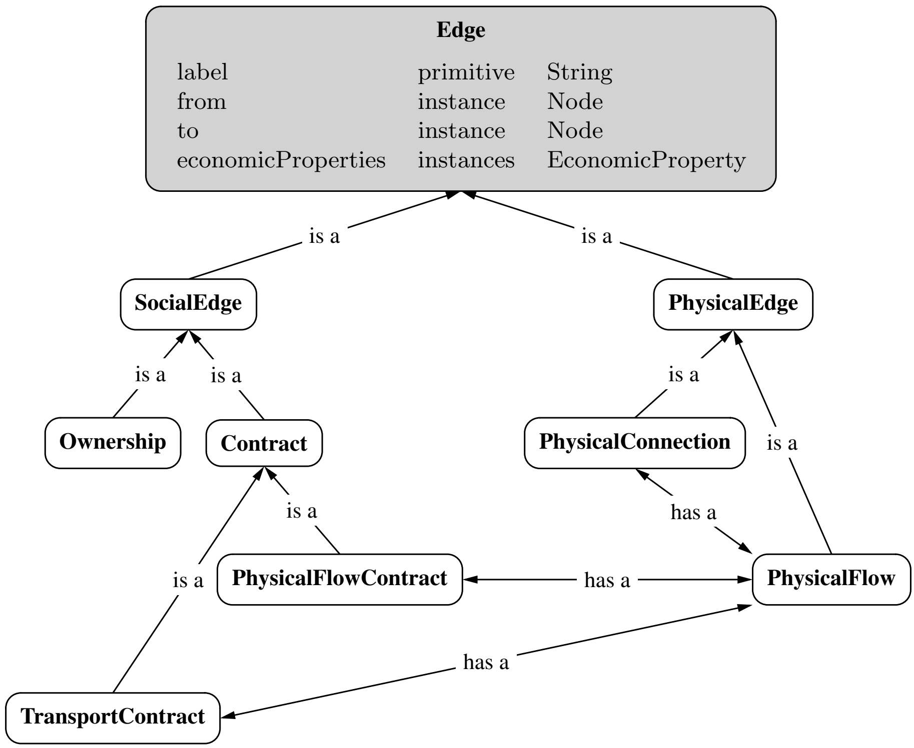


Figure 3.8 – *Edges in the ontology*

3.5.2 Edges

The concept of an Edge is key for the (abstract) definition of networks (See Figure 3.8). Edges connect Nodes to each other, forming the backbone of the infrastructure to be modelled. The following properties are defined for the class Edge:

label A name given to the Edge.

from The Node from which the Edge is connected.

to The Node to which the Edge is leading.

economicProperties Properties that are related to the economics of the Edge.

An Edge can only have one from and one to Node defined, but multiple economicProperties.

To create both physical and social networks, two types of Edge are required: PhysicalEdge and SocialEdge. In the following two sections these two types of Edges are introduced.

3.5.2.1 Physical edges

PhysicalEdges are Edges between PhysicalNodes. Figure 3.9 illustrates the concept of a PhysicalEdge and those concepts directly related to it.

A PhysicalEdge is a container-class without own specific properties. It is useful to define it as a separate class though, so that it can be used to define restrictions for Nodes to only allow PhysicalEdges, for example. Two different PhysicalEdges are considered: PhysicalConnection and PhysicalFlow. PhysicalConnection is the "hardware" and real links of the infrastructure connecting two Nodes. One can think of a pipeline, a power cable or a road connecting two Nodes, making transport of mass or energy possible. The following properties are defined for the class PhysicalConnection:

from The Node the connection is originating from, refined to Technology.

to The Node the connection is going to, refined to Technology.

designProperties A list of properties related to the design of a physical system.

physicalProperties Properties related to the physical aspects of this link.

transportModality The modality of the transport going through this connection, such as pipe, road or sea.

content The flow going through this PhysicalConnection.

A PhysicalFlow is the actual flow of mass or energy between two Nodes. PhysicalFlow has the following properties:

from The Node the flow is originating from, refined to Technology.

to The Node the flow is going to, refined to Technology.

goodName The name of the good in the flow.

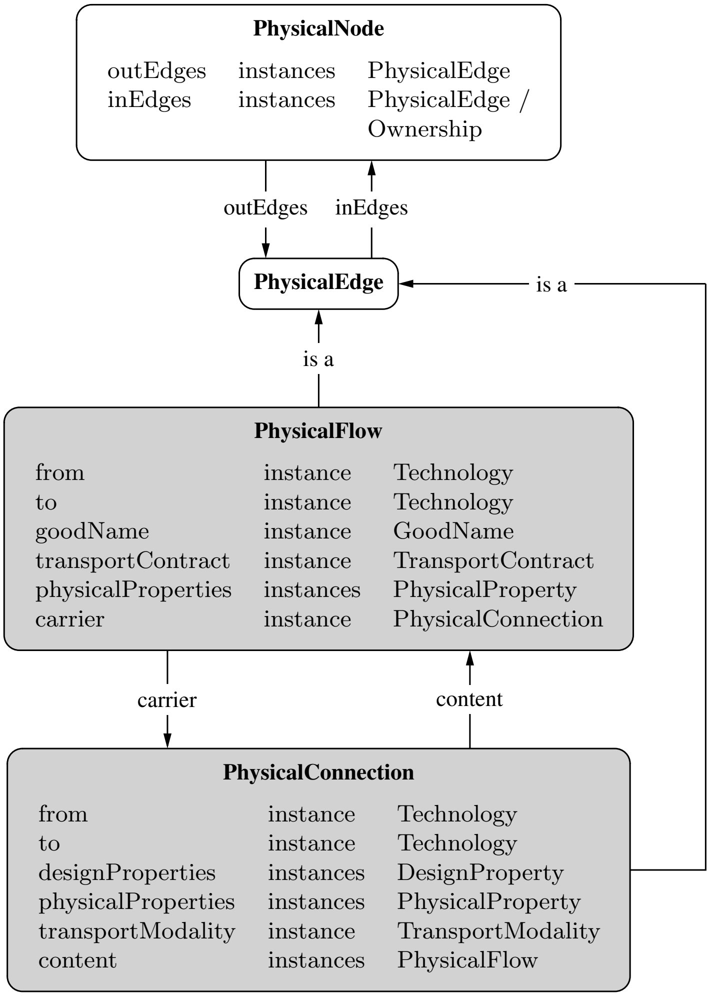


Figure 3.9 – *Physical edges in the ontology*

transportContract The contract with the transporter responsible for connecting the flow between the from and the to Node.

physicalProperties Properties related to the physical aspects of this link.

carrier The PhysicalConnection that carries the flow.

Together, the PhysicalNodes and PhysicalEdges form the physical infrastructure.

3.5.2.2 Social edges

Where PhysicalEdges model real connections, mass and energy flows, a SocialEdge is a social construct and it helps to form the social network of a socio-technical system as well as establish the link between the physical and social networks. There are two subclasses for SocialEdge in the generic ontology: Contract and Ownership.

Figure 3.10 shows the Contract class and its subclasses PhysicalFlowContract and TransportContract. A PhysicalFlowContract between two Agents arranges the trading of a PhysicalFlow between two Technologies. A TransportContract between two Agents arranges the delivery of a flow traded with a PhysicalFlowContract.

Contracts share the following properties:

to Inherited from Edge, refined from Node to Agent.

from Inherited from Edge, refined from Node to Agent.

startTime The time for which the Contract is first valid.

endTime The time at which the Contract is valid for the last time.

signTime The time at which the Contract was signed by both parties.

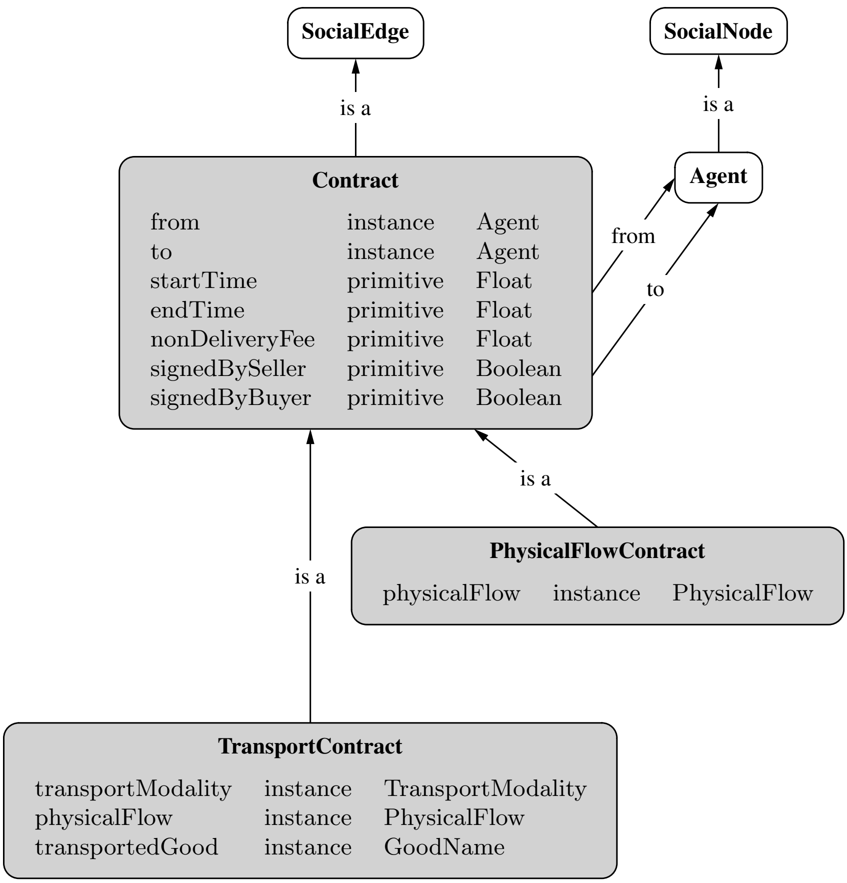


Figure 3.10 – *Contracts in the ontology*

signedByBuyer Indicates if the buyer has signed the contract.

signedBySeller Indicates if the seller has signed the contract. Only a contract signed by both parties is valid.

nonDeliveryFee A penalty fee arranged by the buyer and seller in case the terms of the contract are broken, for example because a delivery could not be made.

Another subclass of SocialEdge can be used both between SocialNodes and between a SocialNode and a PhysicalNode. This is the OwnershipEdge (See Figure 3.11). An Ownership edge can connect a SocialNode and a PhysicalNode, forming the link between social and physical networks. This way it can be defined that an Agent is the owner of a Technology or another PhysicalNode. An Ownership edge can also be used between an Agent and another SocialNodes (e.g. a company that consists of several departments, or to define that a department consists of specific people).

With the most important Nodes and Edges defined, the next section deals with data objects to refine the system specification.

3.5.3 Properties

In the discussion of the Nodes and Edges above, various properties have already been mentioned. These are EconomicProperty, PhysicalProperty, DesignProperty. These

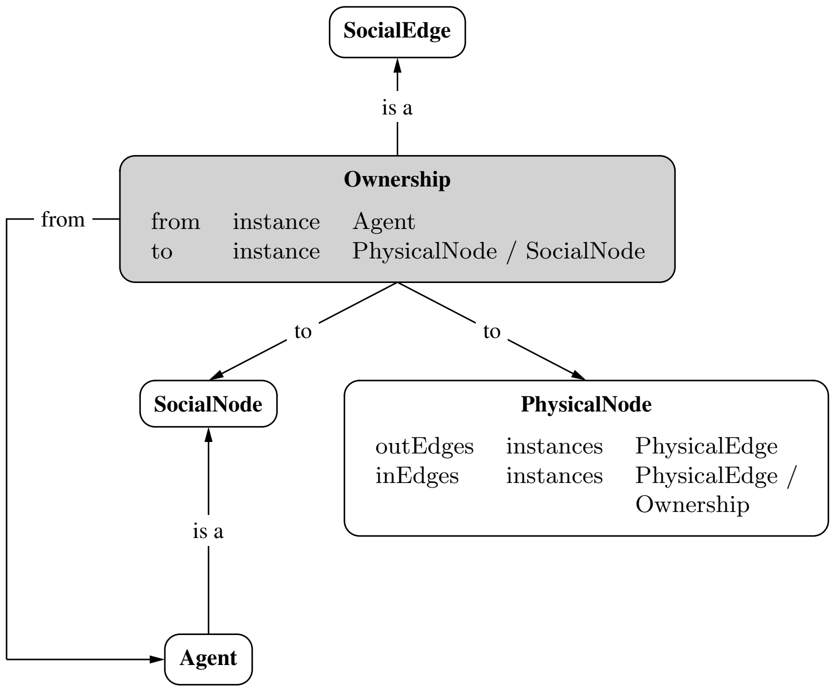


Figure 3.11 – *Ownership in the ontology*

three classes are presented next, before looking at a number of other properties. All subclasses of Property inherit the following property:

literatureReference A source where the value of a specific property comes from, for example a link to a website[11] or a journal paper where the facts are stated.

3.5.3.1 Economic Properties

A large number of subclasses of EconomicProperty have been defined. The most common and widely applicable ones are detailed below. All properties can be used in various other classes (such as the EconomicProperty of an Agent, or an Edge, or a PhysicalFlow) but not all combinations might make sense to use in practice.

Price The price, for example of a product or service.

DevelopmentCost The costs associated with the development of, for example, a PhysicalNode or PhysicalConnection. This has to be paid once, before construction starts.

ConstructionCost The costs associated with the development of, for example, a physical node or physical connection. These have to be paid before it becomes operational.

OperationalCost The price to keep a technical system operational. Can be associated with a fixed price per time period, for example, or with a certain throughput based on the Unit chosen.

MaintenanceCost The costs paid for maintenance, for example of a physical installation.

DeconstructionCost The costs associated with the demolition of, for example, a PhysicalNode or PhysicalConnection. To be paid before taken out of operation and before it is removed from the system.

LiquidAssets The capital of an Agent, from which financial transactions can be paid.

As always, the set of classes can be expanded for specific case studies. Examples of other EconomicProperties include VariableCost, CapitalAssets, Loan, PriceTrend, DemandTrend, SupplyTrend and RiskAttitude.

3.5.3.2 Physical Properties

In the same fashion as EconomicProperties, PhysicalProperties can be used as a property for many classes, both social and physical. They are, for example, used to describe the contents of a PhysicalFlow or an input *2-tuple*[12] of an OperationalConfiguration, but some are also applicable to the social aspects such as the location of an Agent.

Mass The weight, for example to indicate the amount of a product traded.

Temperature The temperature, for example the value required as input for a process.

[11] The date on which the source was checked should be included.

[12] *Tuple* (noun): With preceding algebraic symbol: (an entity or set) consisting of as many parts or elements as indicated by the symbol (*Oxford English Dictionary Online* 2009).

Pressure The pressure, for example of a produced product.

Location The location, for example as a set of x, y coordinates.

Volume The volume, for example the contents of a storage tank.

Other PhysicalProperties that have been added to the ontology for specific case studies include GISLocation[13], PhaseState, LowerHeatingValue, HigherHeatingValue, CASNumber[14], Voltage, Current, Energy, Area, Shape and DangerClassification.

3.5.3.3 Design Properties

The most widely applicable design properties are:

MaximumCapacity The maximum capacity of a process, storage tank, etc.

MinimumCapacity Same as above but for the lower value.

DesignCapacity The capacity it was designed for (e.g. most efficient).

ConstructionTime The amount of time it takes for a Node or Edge to be created.

CreationTimeStamp The moment at which a Node or Edge was created.

LifeTime The expected lifetime of a Node or Edge.

For case-specific needs, classes such as MaximumSpeed and EnergyEfficiency have been added.

3.5.3.4 Other properties

Alongside PhysicalProperties, EconomicProperties and DesignProperties, a number of other subclasses of Property are defined:

GoodName To indicate a class of goods, for example crude oil or ethylene. Goodname has properties GoodLabel, PhysicalProperties

Coordinates A pair of two coordinates

UnitName To indicate the unit used to define other properties, for example kbbl (kilo barrels) as a unit for Volume.

Labels All sorts of labels that can be used in various places. See below for an overview.

The following commonly used Labels are included in the generic ontology:

FunctionLabel Labels that indicate the function of a data tuple in an OperationalConfiguration such as LimitedEmission, CoProduct, Feedstock, FosilOrigin etc.

GoodLabel Labels that can be attached to goods to indicate whether it is for example an OrganicChemical or an EnergyCarrier. These labels can later be used to reason about goods (e.g. an agent can look for all goods labelled EnergyCarrier).

[13] Geographical Information System.
[14] Chemical Abstracts Service Number.

3.5.4 Instances

As said in Section 3.3.1, instances are *concrete* single identifiable objects belonging to a class defined in the ontology. Many instances of the classes described above have been created and they can be read and used by anybody using the ontology. In many cases it will be required to add new instances for a specific case study, but there are many situations where it is useful to re-use instances that have been defined earlier.

As an example of this consider somebody wanting to build an agent-based model of a pharmaceutical company and its links with suppliers (something which has, until this date, not yet been done within this framework). Rather than starting with an empty model, elemented that have already been added to the shared knowledge base may be applicable. One could think of several petrochemical industrial plants that may have been used in other case studies (See Chapter 4) as well as agent definitions for the world market. Additions to the instances will still have to be made, but those are in time perhaps also useful for other models again.

To give an impression of what the current[15] contents of the knowledge base are, below is an overview of some often used classes in the ontology and the number of instances they have at this stage:

Agent Some 90 Agents have been created, including the world market and the environment as well as owners of specific technical installations. Typical examples include a transport company, several households with their own characteristics, a seller of liquid natural gas and an operations department. Using CaseLabels it is possible to not use all Technologies that are associated with a specific Agent, but limit it to those important for a specific case. When including the world market agent, for example, one might want it to only be able to sell certain products.

Technology Over 300 different Technologies are defined in the knowledge base, each providing a rich and detailed definition of the inputs and outputs of the transformation process. Examples range from an oil refinery to a wind turbine, and from a fuel cell to a storage tank for diesel. For a large number of technologies detailed data on, for example, maintenance costs, maximum capacity and construction time is included. These properties are defined using the concepts presented in this chapter.

GoodName More than 300 different goods have been included in the knowledge base so far, including many products from the chemical process industry (based on work done in cooperation with industry partners). It is important to re-use these as much as possible between models, as it will make it possible for trading clusters to be formed of Agents using the same definition of a GoodName. Furthermore, GoodNames can be labelled with one or more classes, such as BioMass or EnergyCarrier, enabling reasoning about the products. For example, an Agent who is the owner of a bio-mass power plant can search for other Agents who can supply any Biomass, regardless of what type. The energy content and other properties of these goods can also be defined (e.g. in a ComponentTuple for the OperationalInputs of the power plant, see Figure 3.7) so more flexible descriptions of technologies are possible.

[15]The knowledge base is in constant use and new instances are added on a regular basis for new case studies. The data given here is from June 2009.

3.5.5 Putting it all together

In any new model these elements can be re-used to quickly create new models and, if needed, new elements can be added. Definitions of entities in the physical network are easily re-used. For agents this is less straightforward, because actors often have very specific behaviour that needs to be modelled. Still, through the use of the ontology and shared building blocks, also behavioural rules of the agents can be shared between models. One way of doing this is by *extending* already defined agents classes thereby inheriting methods of the super class, or by simply copying the code into a new model.

As an example the trading and production behaviour of agents can be given. This has been implemented in a generic way: an agent looks at its technologies to find out which raw materials are needed to run production. Agents will then contact others who are potentially able to supply them with this product (e.g. because a physical connection between two Technologies exists). These potential suppliers again decide, based on the possible outputs of their production technologies, if they can make an offer and send a trade contract. Because the agents use the same ontology, it is well defined that such communication and the exchange of contracts is interoperable. Functions such as paying for maintenance costs, again as defined in the ontology, and arranging shipping contracts, for example, can easily be re-used.

With the classes defined above and (optionally) re-using instances that have been included in the knowledge base, socio-technical system definition can be created. Looking back at Figure 1.1, both the social and the physical networks can be defined as well as the links between them. The ontology is stored in a Protégé (Gennari et al. 2003) knowledge base (See Section 3.3.5) which can be changed without having to adjust the model source code, which works independently. A knowledge base reader (see Box 5) has been created to read the instances in the knowledge base. It creates instances in the Java model, which can then be used in the model. All classes defined in the ontology become objects so they can be instantiated at run-time. In some cases, such as *transport contract* and *physical flow*, instances are predominantly created during the model run based on actions of the Agents.

Box 5 — Knowledge base reader

The shared ontology is stored in a Protégé knowledge base, while the agent-based models need access to instances of Java objects to operate. A *knowledge base reader* was developed to create Java objects which are instantiated with the information read from the knowledge base.

For example, a modeller wants to use a certain power plant, which is defined in the shared knowledge base, in a model. The knowledge base reader then accesses the Protégé knowledge base and reads all properties of this Technology and creates a new Java instance of the right class. Based on the names of the properties and the allowed values, it finds the appropriate methods to *set* the value in the Java object. The object created is then returned to the model. The user can make a selection of *which* objects to read based on class type or CaseLabels, among others.

The knowledge base reader uses *recursion* to traverse through the knowledge base. When reading an object it will encounter several new objects (e.g. an OperationalConfiguration when reading a Technology) which in turn may also refer to multiple others. The read an object function is called recursively until the leaf of a branch is reached. The reader is designed to work with any conceptualisation and none of the classes or properties are hard-coded. As such, no adjustments need to be made to the reader when the framework is updated.

One of the main advantages of the use of a separate reader is that it makes it possible to create models independently of the ontology tool or the *ontology language* and it allows the definition of an ontology independently of the models. This idea originated from the design of an "ontology translator" described in van Dam (2002).

3.6 Modelling steps

The following model building tasks should be performed to develop an agent-based model using the framework presented in this thesis (Lukszo, van Dam, Weijnen & Dijkema 2008):

M-1. Conceptualise the problem in terms of actors and physical systems, including their relations and properties. Distinguish the set of properties that will act as variable in the model (i.e. the model parameters) and possibly visualise the interaction between model parameters in an influence diagram.

M-2. Define possible disturbances on the system that one wants to study and that may effect the system in such a way that the "normal" operation can no longer cope with them.

M-3. Refine the generic ontology with new abstract classes applicable for this case or the addition of properties to already existing classes.

M-4. Make the model specification by creating concrete instances of the abstract classes from the ontology.

M-5. Implement the behaviour of the agents (under normal conditions and, if applicable, responses to possible disturbances), making use of generic components (e.g. searching for suppliers, determining a price, accepting contracts) and add new components if needed.

M-6. Verify the model. During the verification the model is checked against its conceptual design (i.e. are all relevant entities and relationships from the conceptual model translated into the computational model in a correct way?). In other words, it is studied whether the modeller "built the thing right".

M-7. Validate the model. Validation is a continuous process, not a single test to check if the model output matches its "real world" system. During the validation step modellers and domain experts evaluate if the model is useful and convincing. In other words, it is studied whether the modeller "built the right thing".

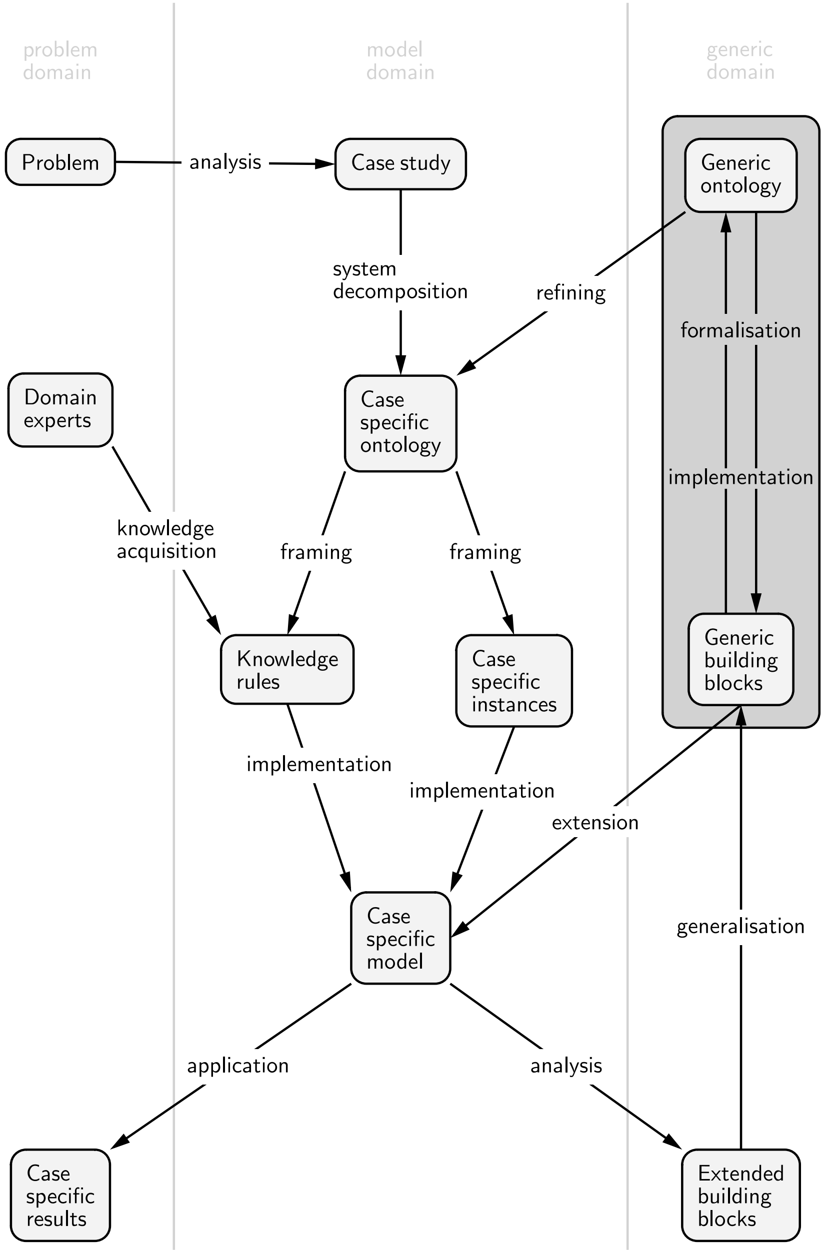


Figure 3.12 – *Application cycle for re-use of ontologies and generic building blocks (based on van Dam & Lukszo 2006)*

3.7 Core elements and conclusions

In this chapter an introduction of the framework for socio-technical infrastructure systems was given. The core of the framework consists of three elements:

Interface An interface definition between components, between models, between developers and between developers and problem owners, expressed and formalised in an ontology.

Library A shared library of source code that can be re-used, including agents (e.g. shipping agent) and specific behaviour (e.g. procurement behaviour). The components of the library can be seen as "building blocks" to create new models using the framework.

Procedures Procedures on how to use the library and interface to define and build models of socio-technical infrastructure systems.

Figure 3.12 shows how the framework can be used to answer case-specific problems (in the *problem domain* on the left) by using generic building blocks (from the *generic domain* on the right) within a case study. By refining the generic ontology and creating subclasses, properties and specific instances, new models can be set up based on the generic framework. The re-use of building blocks allows new models to be set up quickly so one can focus either on modelling case-specific behaviour, or on experimenting with new scenarios. Full strength of the framework is shown when the modifications done for a case study are fed back into the generic elements. The model not only delivers case-specific results, but new elements developed for the model contribute to the reusable framework.

The framework itself is never finished, but, as will be demonstrated in Chapter 5, it can now be concluded that the core of the framework is stable (i.e. no new concepts are added that the majority of the projects require, and those concepts that are most used are not changed any more) and suitable for a large variety of cases and domains. Models have already successfully been designed and implemented using the framework for various infrastructures, including transport, energy and industrial networks, as will be demonstrated in the next chapter.

Chapter 4

Case studies — application and use of the framework

This chapter is based on van Dam & Lukszo (2009).

4.1 Introduction

In this chapter the framework from Chapter 3 is applied to three illustrative case studies: an intermodal freight hub and an oil refinery supply chain, followed by a revisit of the initial model from which the framework was developed (See Section 3.4.1). These cases show how the framework is used to develop models in different infrastructure domains. The aim of these applications is twofold: The applications demonstrate that agent-based models can successfully be built using the framework for the application domains covered by this thesis and they serve to illustrate how the framework works in practice, including re-use of "building blocks" between the models. In this chapter model *development* is addressed, rather than the *use* of models, which will be addressed in Chapter 7.

The first two cases presented in this Chapter have been selected from different infrastructure domains and are based on real-world challenges. The third case as an abstract proof-of-concept model. Together with a brief overview of a number of models developed by others — but using the same framework — they demonstrate the wide applicability of the approach and highlight that at higher levels of abstraction the various domains can, indeed, be considered as similar. Furthermore, cases one and two are illustrative of *Intelligent Infrastructures* and the challenge is how to make best use of the already available infrastructure capacity through smart and innovative policies and approaches. The focus is on the relationship between the distributed actors and the effect their decisions have on the overall system performance.

The rest of this chapter is structured as follows:

Section 4.2. Intermodal freight hub: A model of an intermodal transport system is developed to allow the user to experiment with different locations for a new hub and with different measures to encourage stakeholders to agree with a certain proposal for the hub location. The system consists of different actors with their own interests and control over (part of) the physical transport network, incorporating different modes of transport.

Section 4.3. Oil refinery supply chain: A model of a supply chain is built, revolving around an oil refinery, including production, storage and transportation of raw materials and products. This supply chain, which goes beyond the control of one actor, can be considered as an infrastructure and a typical example of a socio-technical system. The model can be used as a decision support tool for experimenting with, for example, various policies for procurement and storage as well as abnormal situation management.

Section 4.4. Chocolate production network: The proof-of-concept model of an industrial cluster is revisited. In this test case a *chocolate production chain* was imagined as representing clusters in the (petro-)chemical industry, and it is re-developed to use the current state of the modelling framework and the ontology, completing the loop from Figure 3.12.

Additionally, case studies have been executed by others, again using the same framework. This shows the wider applicability of the framework as well as the re-use and sharing of model components between models developed by different modellers. Two of such cases are briefly discussed in Section 4.5, namely:

Section 4.5.1 Evolution of industrial clusters: A model to visualise the development, growth and decline of clusters aiming at increasing understanding of dependencies between nodes in the cluster. Applied to the petro-chemical industry.

Section 4.5.2 CO_2 emission trading: A model of energy producers investing in their portfolio of electricity production facilities to assess the impact of the CO_2 emission trading scheme.

Together these case studies give a good overview of applications in the infrastructure, energy and industry domain. For each model in the sections below, the model building steps from Section 3.6 are executed and conclusions are drawn about the specific case, before summarising this in Section 4.6.

4.2 Case 1: Intermodal freight hub

Intermodal freight transportation is defined as a system that carries freight from origin to destination by using two or more transportation modes. Intermodal freight transportation has become an attractive alternative to road transport, as the latter can no longer assure a reliable and sustainable service delivery as a result of traffic congestion, rising fuel prices and air pollution problems. However, the increasing demand of intermodal freight transportation has posed a new challenge, namely how to provide sufficient infrastructure that will meet that demand and maintain a satisfactory level of services through investments in freight hubs and transport links. A comprehensive review of intermodal rail-truck freight transport literature is given by Bontekoning, Macharis & Trip (2004).

In an intermodal freight transport system hubs are one of the key elements: they function as transfer points of freight from one transport mode to another. The success of an intermodal freight hub depends on four major factors: location, efficiency, financial sustainability, and level of service (e.g. price, punctuality, reliability or transit time) (Meyrick and Associates 2006). The location of hubs is a critical success factor in intermodal freight transportation and needs to be considered very carefully as it has direct and indirect impact on different stakeholders including investors, policy makers, infrastructure providers, hub operators, hub users and the community (Sirikijpanichkul 2006, Sirikijpanichkul, van Dam, Ferreira & Lukszo 2007).

The study presented in this section relates to making models to support the decision making process for choosing a location and realizing a new freight hub. It is a real policy problem in the South East Queensland region in Australia (Sirikijpanichkul 2006), see Figure 4.1.

4.2.1 Conceptualisation of the problem in terms of actors and physical systems

As said, the system consists of different actors with their own interests and control over (part of) the physical transport network, incorporating different modes of transport. As an initial model for this case study a simplified version of the complex realistic transport network is used. The actors modelled in the social system are shown in Table 4.1 alongside the physical nodes. Hub users are companies that use the hub for their transport demand and that buy goods to sell. Aggregated demand is modelled as one actor, called the consumer and the world market is seen as the source for all products. The container terminal and intermodal freight hub operators are responsible for the transfer of goods from one mode of transport to another and all transport on the links is taken care of by one aggregate party, here called the transport company. Other stakeholders, such as the community[1], are left out of this version of the model for now. Such actors can easily be added to the model later.

The ownership relationships between the SocialNodes and PhysicalNodes are also shown in Table 4.1, but the system contains many other relationships including PhysicalFlowContracts (between Agents) and PhysicalFlows (between Technologies). Most of these Edges are created on the fly in the model as the result of the behaviour of the agents, except the PhysicalConnections between the Technologies (which are fixed based on the

[1] The community is a key stakeholder because the location of the intermodal freight hub can have significant effect on, for example, noise pollution but also on traffic for other road users.

Table 4.1 – *Agents and PhysicalNodes and their relationships for the freight hub case*

Agent	Relationship	Physical Node
World market	owns	Delivery installation
Container terminal operator	owns	Container terminal
Intermodal freight hub operator	owns	Intermodal freight hub
Hub user	owns	Toy factory
Consumer	owns	Consumer installation
Shipper	—	not applicable

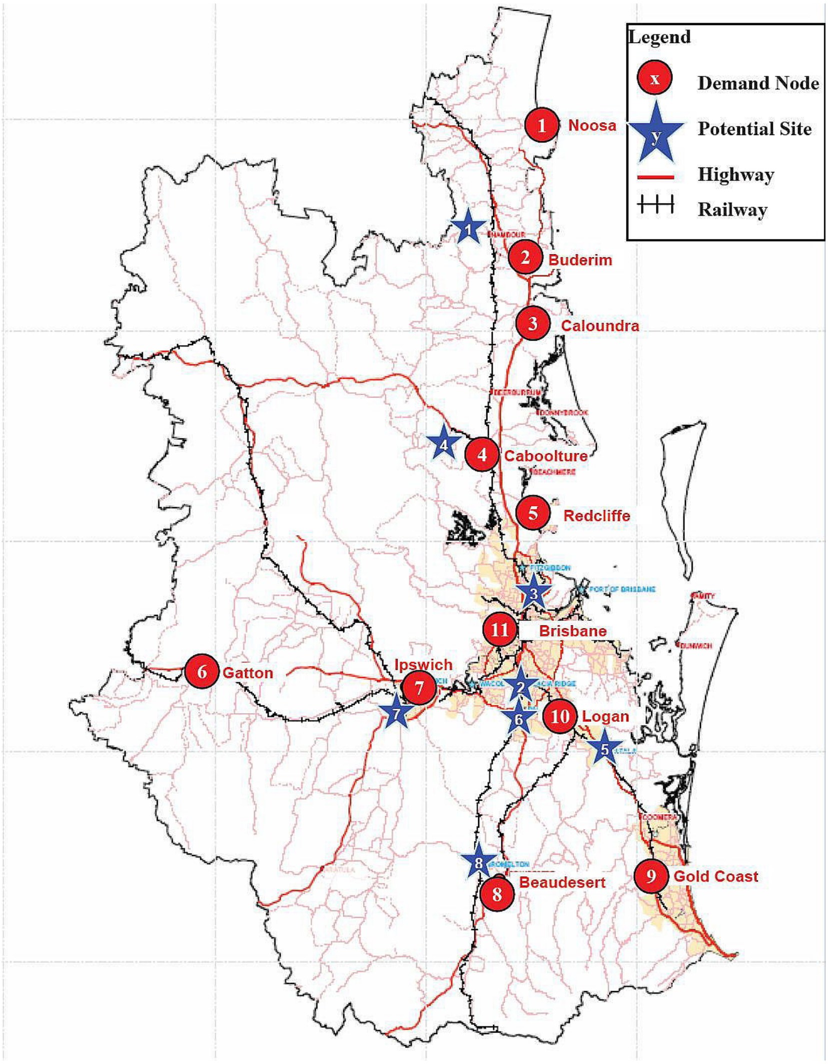


Figure 4.1 – *Candidate sites for a new intermodal freight hub and potential demand nodes in the study area in Queensland, Australia (Sirikijpanichkul et al. 2007)*

existing infrastructure) and the Ownership relations (which are fixed based on existing relationships between the social and physical system). Figure 4.2 shows a screen capture of the model during a simulation run, to illustrate these other relationships. See Lukszo et al. (2008) for a more detailed description of the conceptualisation of the model.

4.2.2 Definition of possible disturbances on the system

For the intermodal transport system incorporating the freight hub various disturbances can be considered. These include, for example, technical problems with cranes in the

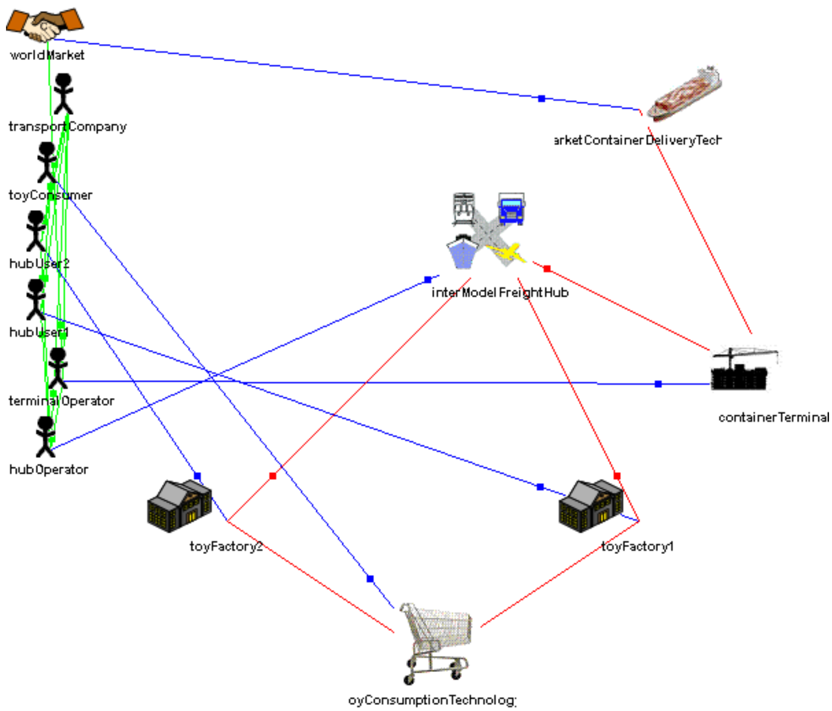


Figure 4.2 – *Geographical representation of the social nodes (i.e. the Agents, on the left) and physical nodes (i.e. the Technologies, on the right) in the freight hub model. Edges between Agents are SocialEdges (e.g. PhysicalFlowContracts) as are those between Agents and Technologies (e.g. Ownership). Edges between the Technologies are PhysicalEdges (e.g. PhysicalFlows)*

hub, problems with transport links, traffic jams, clients going out of business, sudden changes in transport demand, etc. At this step, however, no disturbances are included in the model. With the same model of the physical infrastructure and the key actors — but with different needs from a problem owner — one could envisage expanding the model and running it for a different purpose for which disturbances are important and need to be included.

4.2.3 Refinement of the generic ontology with new abstract classes

The generic ontology for socio-technical systems was refined for the freight hub model. While many concepts could be reused, new abstract concepts such as GIS[2] location (a subclass of PhysicalProperty), TransportContract (a subclass of SocialEdge) or new transport modalities for rail and truck (a subclass of TransportModality) were created to enable the specification of the model in the ontology. These new concepts are shared with other

[2]Geographical Information System, a set of coordinates used to determine a position on Earth.

models so they can be re-used. The concept of a TransportContract that was created for this case, for example, was later re-used for the oil refinery case study as presented in Section 4.3.

4.2.4 Creation of concrete instances

Instances for the actors, physical nodes, and all fixed relations (as described in Section 4.2.1) were added to the shared knowledge base too. Instances of agents that are also used in other models, in this case the world market, did not have to be created but are re-used. The instance of the world market was updated by adding an ownership relationship with the container delivery installation, so in addition to the goods traded on the world market in other models (e.g. petro-chemicals and natural resources) the world market can supply containers with toys. Values for various properties (e.g. the location of PhysicalNodes and price for transport charged by the transport company) had to be based on the results of an initial set of experiments because for this proof of concept model no real data was used.

4.2.5 Implementation of the behaviour of the agents

The behaviour of all actors is modelled as searching for other agents that can offer the desired goods. This means that Agent A looks for a set of other agents that own a Technology which has an output that matches with the input of the Technologies of Agent A. This procedure is generic and not dependent on the goods (i.e. containers or toys for this model) traded. All behaviour is demand driven and agents act in the order of the supply chain (i.e. the consumer starts first, ending with the world market). Agents thus collect a number of unsigned PhysicalFlowContracts from other agents and then choose the best contract to sign, thereby committing to a transaction. The generic trading algorithm based on the inputs and outputs of the Technologies of Agents is also why the world market agent needs a *container supply technology* and the consumer a *consumption technology*: this way the owner of the Technology "knows" it can supply this good or that it should ask for a specific product. After agreeing to buy goods, agents also need to arrange transport for these products by asking the transport agent for a transport contract.

All agents pay maintenance and operational costs for the physical system they own (if applicable), which are defined through EconomicProperties of the Technology[3]. Again, this is generic behaviour and all agents inherit this behaviour. The buying goods — and, on the other side, offering of trade contracts after being asked to supply a certain product — as well as the paying for maintenance and operational costs is done in a generic way and could be re-used from earlier models developed using the framework[4]. Specific behaviour for the transport agent was needed though, as an agent with this functionality had not been used in previous case studies. The transport agent is contacted by other agents who have signed a PhysicalFlowContract and the transport agent makes an offer for the transportation of this flow, charging a price based on the distance, mode of transport and volume traded. The transport agent then monitors the traded flow and connects it to the receiving party when the shipment has arrived.

[3]Note that the transport company, not owning a physical node, does not need to buy any goods on the market and does not have operational costs defined the same way as other agents: it does pay maintenance and operational costs for its fleet of vehicles, but these are added as an EconomicProperty of the company itself.

[4]This was already implemented as part of the generic "building blocks".

4.2.6 Verification and validation of the model

The model presented here is a proof-of-concept model and is designed to illustrate the applicability of the framework for the intermodal transport domain. In series of model runs with different extreme values for the parameters the model was verified. The results of these tests matched the expected outcomes.

The model is not based on a real-life case and real data, therefore, no validation by comparison with real world data took place. Expert judgement by people in the transport modelling domain and experts in intermodal freight systems was used (Sirikijpanichkul et al. 2007) to conclude that the model is useful (i.e. it offers added value to solving a real complex problem) and there is confidence in the model results (i.e. the results match with the expectation of the experts).

4.2.7 Conclusions

In this section a model of an intermodal freight transport system, incorporating a freight hub, was presented. While the model itself is not based on a real system with real data, the proof-of-concept model does serve to illustrate the applicability of the framework from Chapter 3 in this important infrastructure domain. Most of the behaviour of this model could be based on previously developed "building blocks". The transport company behaviour, specifically implemented for this case, was added to the shared library of source code. This was later re-used for the oil refinery case study as presented in Section 4.3

4.3 Case 2: Oil refinery supply chain

In this section a model of an oil refinery supply chain is presented. A hierarchy of decisions has to be made in managing the supply chain: strategic (e.g. capacity investments, adding units, upgrading technology, supply chain reconfiguration), tactical (e.g. production planning, policy evaluation, disruption management) and operational (e.g. procurement, storage, scheduling, throughput level). These motivate the development of simulation models of the supply chain, which could reflect the dynamic behaviour of the entities in the face of the various uncertainties. This model enables decision making for supply chain management by allowing the user to evaluate the impact of a particular decision on the supply chain performance, analyse different supply chain policies, and identify the consequences of a disruption, through simulation. The model is based on the system description from Pitty, Li, Adhitya, Srinivasan & Karimi (2008) and is described in more detail in van Dam, Adhitya, Srinivasan & Lukszo (2008) and van Dam, Adhitya, Srinivasan & Lukszo (2009).

4.3.1 Conceptualisation of the problem in terms of actors and physical systems

An *oil refinery supply chain* begins from the oil reservoirs, both onshore and offshore. Crude oil is tapped from these sites and then transported to various refineries around the world mostly by pipelines or large ships called very large crude carriers (VLCCs). Transportation times of crude are relatively long; it takes four to six weeks for a VLCC carrying crude oil from the Middle East to reach refineries in Asia, for example. The crudes are then processed in crude distillation units (CDUs) and separated into fractions based on their boiling points. These fractions are processed further in different downstream refining units such as reformer, cracker, and blending pool to get the various products. A single crude mix may yield numerous products and their variants through a suitable alteration of processing conditions. Hence, refineries must adapt their operations to the different crude batches to meet the required product specifications from their customers.

The refinery occupies a pivotal position in the supply chain with its functional departments initiating and controlling the interactions with the external entities, which are oil suppliers, third party logistics providers, shippers, jetty operators, and customers. The operation of the refinery supply chain requires various decisions in every cycle — what mix of products to make, which crudes to purchase and in what quantities, which mix to process and in which processing mode, etc.

Different actors are responsible for the different decisions (Julka, Karimi & Srinivasan 2002). These actors and their interactions are shown in Figure 4.3. The entities (shown as blocks in Figure 4.3) communicate with each other through information flows (broken arrows) in order to control the material flows (solid arrows).

The refinery physical units (shaded blocks) may be further sub-divided into storage units such as crude and product tanks and processing units such as the CDU, reformer, cracker and blend tanks. The functioning of these units and other supply chain activities is overseen by the functional departments: the storage department and the operations department. The actors and physical systems are shown in Table 4.2. Note that some actors, for example the logistics department, do not own or control a physical system, but they do have their own specific tasks and communicate with the other actors.

Table 4.2 – *Agents and PhysicalNodes and their relationships for the refinery supply chain case*

Agent	Relationship	Physical Node
Refinery company	owns	Refinery units (incl. CDU)
Refinery company	owns	Raw materials storage tanks
Refinery company	owns	End product storage tanks
Operations dept. (Refinery)	controls	Refinery units (incl. CDU)
Storage dept.(Refinery)	controls	Raw materials storage tanks
Storage dept.(Refinery)	controls	End product storage tanks
Sales dept.(Refinery)	–	not applicable
Procurement dept.(Refinery)	–	not applicable
Logistics dept.(Refinery)	–	not applicable
3rd party logistics provider	–	not applicable
Shipper	–	not applicable
Supplier	owns	Oil wells
Jetty owner	owns	Jetty
Consumer	owns	Consumer installation

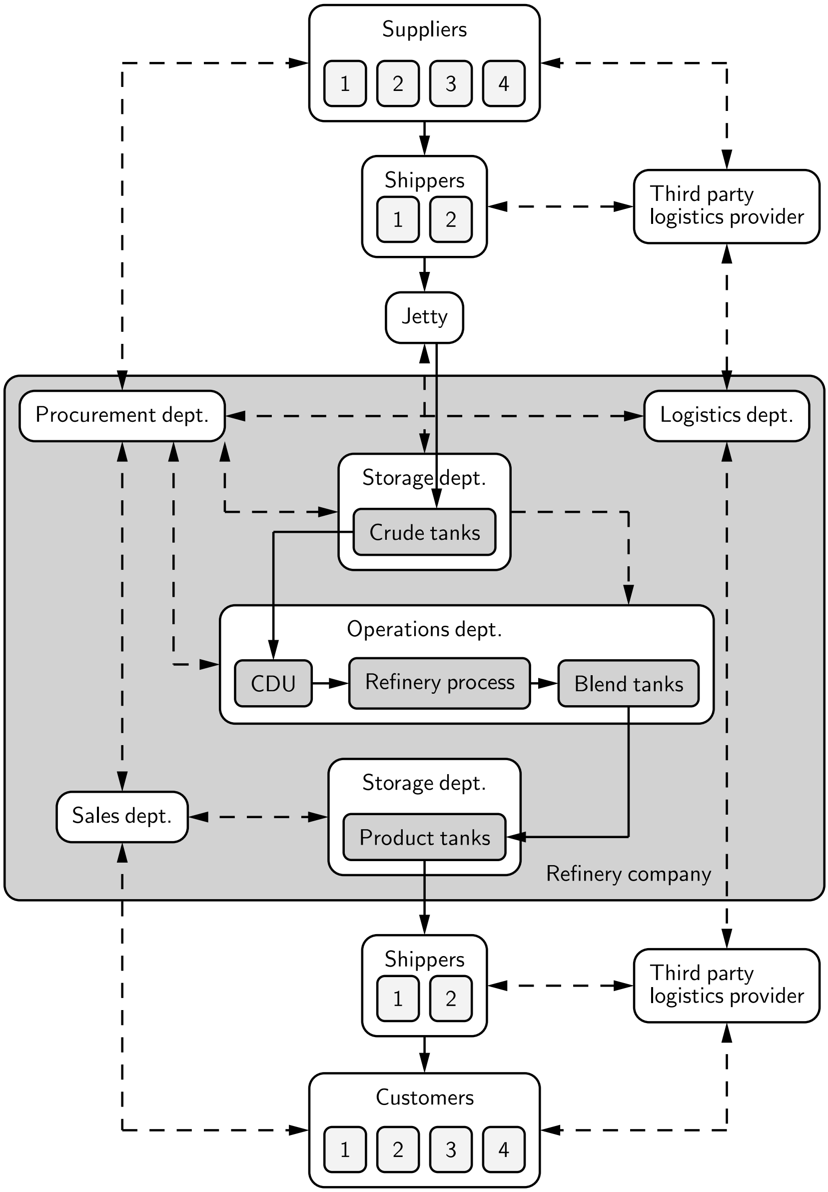


Figure 4.3 – *Schematic of an oil refinery supply chain (from Pitty et al. 2008). Arrows with a solid line represent material flows and arrows with a dashed line represent information flows*

4.3.2 Definition of possible disturbances on the system

An important element of supply chain management involves dealing with disturbances, occurring at a certain point in time. Hence, a number of possible disruptions to the normal flow in the supply chain is considered in the model. For instance, a disruption in the supply can be caused by a delay in the shipment of crudes from the supplier, who is at a large distance from the refinery, or problems in the tank farm. Additional disturbances can be thought of, but these are not yet included in the model. In case of a delay with a shipment or a problem with one of the storage tanks, the operations department runs the risk of not having access to enough crude to perform the scheduled operations. A delay for a ship can be just one or two days due to bad weather, but in case of hijacking (i.e. piracy) or technical failure this could be more than one week or even longer[5]. In the model, the duration for a ship delay can be longer than the time horizon used for the model, effectively "sinking" the ship.

For simplicity, it is assumed that the magnitude of the disturbance is known as soon as the disturbance occurs. In reality, this may involve uncertainty. Furthermore, currently delays are in the order of magnitude of days, but the granularity could be adjusted so that a delay could be expressed in parts of a day (e.g. hours) instead of full days of 24 hours.

4.3.3 Refinement of the generic ontology with new abstract classes

For the development of the oil refinery supply chain model no major changes to the generic ontology were needed. All the key classes needed to define the system were already in place, based on earlier case studies. Only minor additions were needed, such as adding properties to the TransportContract for more detailed registration of transport delays and payment.

4.3.4 Creation of concrete instances

The Agents and Technologies as defined in Table 4.2 were added to the knowledge base. The values of the properties are based on Pitty et al. (2008). New additions to the ontology were needed (see Figure 4.4) in the form of instances for the technical elements (e.g. the refinery units and storage tanks, delivery installations and fixed infrastructure connections between them) with their properties (e.g. production recipes for the refinery for the various mixes of crudes, maximum capacities of the storage tanks, distances for the shipping routes). Furthermore, the initial conditions of the system (e.g. current stock levels in the storage tanks and current financial assets of the agents) were defined.

4.3.5 Implementation of the behaviour of the agents

Each entity acts based on its policies and the combined actions of the entities determine the overall performance and economics of the supply chain. For example, the procurement department decides the type and amount of crude to buy, the logistics department oversees transportation of the crude, and the storage department manages the crude unloading from the ship to the storage tanks. The combined actions from these three departments determine crude arrival at the refinery. The complex maze of flows among the entities could lead to unforeseen domino effects. Furthermore, the refinery has to contend with various uncertainties such as prices, supply availability, production yields, and demand variations.

Following the agent paradigm, the tasks are distributed between the agents. Some tasks therefore have to be split into several subtasks (requiring communication between

[5]Note that an average journey for a VLCC is considered as 14 days in this model.

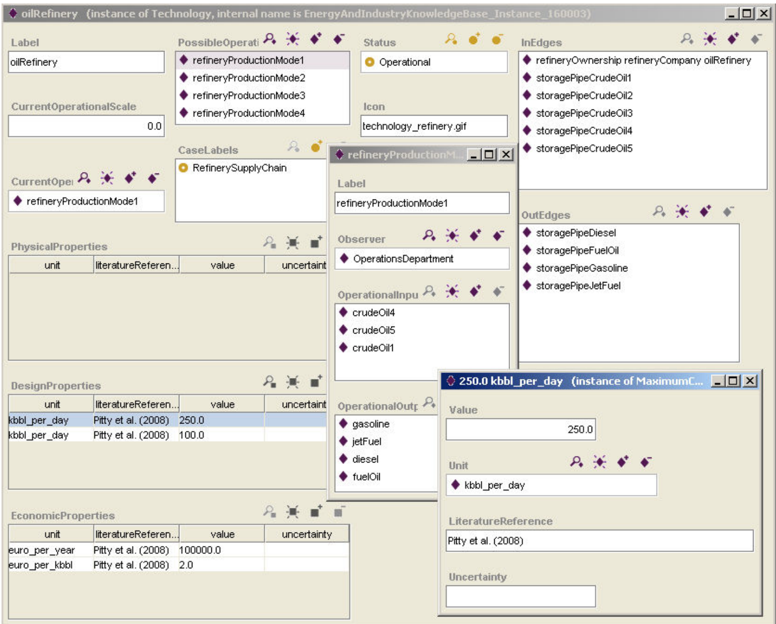


Figure 4.4 – *Screenshot of the graphical user interface of Protégé for the creation of instances: the MaximumCapacity is defined as a DesignProperty*

the agents). A schedule is made so that some processes (e.g. procurement) only occur at certain intervals while others (e.g. production) happen at each time step of the simulation. Events such as the arrival of a VLCC at the jetty are monitored each time step.

As an example of some basic algorithms related to the transfer of crude in the system, consider the following. The storage department can receive incoming crude (pumped from the jetty) and it can request to release a certain amount of crude (from the operations department). The storage department monitors the incoming flows at each time tick of the model and sets the new level of the storage tank:

Algorithm 1 Monitoring inflow of crude (Storage Department)

```
Volume addedVolume = (Volume) storageInFlow.getPhysicalProperties().get(
     Volume.class);

currentVolume.setValue(currentVolume.getValue()+addedVolume.getValue());
```

where 'addedVolume' is the amount added to a storage tank, 'storageInFlow' the flow of crude from the jetty to the storage tank and 'currentVolume' the current volume of crude in the tank. The first line reads the volume that is transferred into the storage and the second line sets the new value of the volume in the storage tank by adding the added amount to the current amount. It is important to note that the volume of the 'storageInFlow' is set by the jetty owner in another algorithm.

The operations department decides on the amount to be released based on the current production mode and current throughput of the refinery, again using an algorithm. When the amount is determined, the storage department is asked to release this from the storage tank. When the storage department receives a request to release a certain crude (from the operations department) it will first check if there is enough in stock and then create a flow to the CDU. The amount of crude in the storage tank is thus adjusted in the following way:

Algorithm 2 Release of crude (Storage Department)

```
Volume substractedVolume = (Volume) outFlow.getPhysicalProperties().get(
    Volume.class);

currentVolume.setValue(currentVolume.getValue() - substractedVolume.
    getValue());
```

where 'substractedVolume' is the volume of crude that will be released and 'outFlow' the flow from the storage tank to the CDU.

All other behaviours in the model are also split up in similar fashion between the different agents. Another example of this is the selection of the production mode by the operations department, which is based on the forecasts made by the sales department and the crudes selected by the procurement department.

Where possible, modelling the behaviour of the agents was based on existing behavioural rules for trading (as also used in the freight hub model from Section 4.2), but additional rules had to be implemented, for example for various procurement policies (e.g. forecasting of demand deciding on procurement), scheduling which Operational-Configuration to use, and for the activities of the jetty which had not been used in earlier models (van Dam, Adhitya, Srinivasan & Lukszo 2009).

4.3.6 Verification and validation of the model

The oil refinery model has been tested extensively. A benchmarking study has been executed in which the agent-based model was compared with an equation-based model of the same supply chain (van Dam, Adhitya, Srinivasan & Lukszo 2009). The aim of the benchmarking study was to compare the modelling paradigms (i.e. the agent-based and equation-based paradigms) and to learn about the advantages and disadvantages of the different approaches. For benchmarking modelling paradigms, it was necessary to demonstrate that the models under study are comparable. A numerical analysis was performed, proving that the two models show the same behaviour (See Chapter 6). This concludes the verification and the validation phase of the agent-based model as successful, the more so because the equation-based model has been validated against the real system and it was applied to offer decision support (Pitty et al. 2008).

4.3.7 Conclusions

An agent-based model of an oil refinery supply chain was presented, following the modelling steps and "building blocks" from the framework. Many elements, including behavioural rules, developed for the freight hub model from Section 4.2.1 were re-used. The ontology did not need any major expansion as the system could be well expressed with already defined concepts. This demonstrates not only that different infrastructure domains can be expressed, but also that it is suitable for models in different temporal and spatial scales.

4.4 Case 3: Chocolate game

In Section 3.4.1 the chocolate game was introduced as the initial model that sparked off the development of the generic framework, after which the development loop from Figure 3.12 could be started. Through use and continued development the initial ontology (which proved to be good, but not yet perfect) was adjusted, refined and expanded so that a wider range of case studies (such as the ones listed above) could be built with it. This initial model is revisited and re-described with the current state of the ontology to make a full circle and to show where changes were required and how "building blocks" have been developed. This model is less advanced and more straightforward than the models developed in Case 1 and Case 2, but it serves as an illustration of the re-use of the elements developed for, among others, those two cases.

4.4.1 Conceptualisation of the problem in terms of actors and physical systems

The chocolate production network considered contains two types of producer agents: intermediates producers, who take raw materials (e.g. cacao beans) from the world market and process them into intermediates (e.g. cocoa powder) which are needed by the end producers to make different types of chocolate bars (e.g. a plain chocolate bar). End producers can, optionally, add (processed) raisins or peanuts to the chocolate bars they make and sell them again on the world market. The actors and physical nodes for this system are shown in Table 4.3. For each type, except for the world market, there could be more than one agent in the simulation. It is assumed the physical network is not a restriction to trading and that a transport connection between all physical nodes exists or is created on the fly when needed (cf. the network evolution model in Section 4.5.1).

The original game had a role for the transport company, but from the initial agent-based implementation this role was left out for simplification: the focus was on the negotiations between the actors and the network that results from the decision making process (van Dam, Nikolic, Lukszo & Dijkema 2006). Now that the model is revisited, this simplification is no longer needed because the behaviour of the transport company was already implemented for the freight hub case as described in Section 4.2. The new ontology is used, which opens the doors to the library of "building blocks", including the implemented behaviour of the transport agent. While it was originally omitted, now it is easy to include this element to the model, making it more alike the game played with human players without much extra effort of the modeller.

4.4.2 Definition of possible disturbances on the system

For a chocolate production chain one could think of similar disturbances to the ones listed in Sections 4.2.2 and 4.3.2 including delays in shipment and suppliers going out of business. In this model, however, no such disruptions have been included as they would not contribute to the aim behind developing this illustrative case.

Table 4.3 – *Agents and PhysicalNodes and their relationships for the chocolate game case*

Agent	Relationship	Physical Node
World market	owns	Delivery installations
World market	owns	Consumer installation
Intermediate producer 1	owns	Cacao bean processor
Intermediate producer 1	owns	Cacao bean warehouse
Intermediate producer 2	owns	Raisin processor
Intermediate producer 2	owns	Raisin warehouse
Intermediate producer 3	owns	Peanut processor
Intermediate producer 3	owns	Peanut warehouse
End product producer	owns	Chocolate bar factory
End product producer	owns	Processed cocoa warehouse
Transport company	—	not applicable

4.4.3 Refinement of the generic ontology with new abstract classes

No major refinement of the generic ontology was needed, as the conceptualisation fits well within the scope of the existing classes. For this case study only new GoodNames had to be defined (for various raw materials, intermediates and end products). The original agent-based implementation did not include 'raisin' or 'peanut' as a commodity (as yet another simplification compared with the serious game that was played) but here these goods are also added because with the more generic trading and production behaviour (as will be discussed below) it is easier to incorporate these new goods (as well as related goods from raw materials to end products, such as a peanut chocolate bar).

4.4.4 Creation of concrete instances

Instances are created for all agents and physical nodes from Table 4.3. For the technologies the inputs and outputs are defined to model the production processed. Other than the ownership relationship, no other edges are needed because, as said above, the physical infrastructure between the nodes is not considered. Properties, such as maintenance costs or location, are not important in this model and do not need to be added. If more instances of one type of agent are needed in the simulation, they can be *cloned* from the instances defined in the knowledge base. This means that a so called "deep" copy of the object read from the knowledge base is made, for which also the instances of its properties (including, for example, any Technologies that have an OwnershipEdge to the agent that is cloned) are copied.

4.4.5 Implementation of the behaviour of the agents

All behaviour of the agents in this model could be based on already existing "building blocks". This is no surprise as the initial implementation of the chocolate model formed the basis of the first set, but for a number of aspects it was implemented differently. Next, for each of the steps in the initial agent-based model a description follows on how they are now implemented:

Choose strategy: The selection of the "strategy"[6] determined the identity of the agent: it contains what the agent is producing and which product(s) are needed for this. The strategy was based on the name of agent (as defined in the instances in the knowledge base) during initialisation of the model. In the re-implementation this is made more generic; based on the technology (or technologies) the agent owns it "knows" which products are required and which ones can be sold. Furthermore, rather than merely during the initialisation, it is determined at each time step so that it becomes possible to include agents investing in new technologies, temporarily having units unavailable for maintenance, for example, *during* a simulation run.

Do market research: Collecting a number of (unsigned) contracts from potential sellers is generic behaviour and the way this was implemented for the initial model is still mostly the same as in the generic framework now (except from the strategy, as discussed above). This behaviour forms one of the key "building blocks" in the framework.

Sign contracts: After a number of contracts have been collected, it is again generic behaviour to choose the best and make a formal arrangement by signing the contract. Again, this behaviour was implemented as a generic "building block" and could directly be re-used here.

Arrange transport: If applicable (i.e. when one or more contracts were signed) agents can arrange transport. This was a separate step in the initial model, but it was not implemented yet. Now this step can, as with the previous steps, be based on generic "building blocks".

Produce: Through definition of the inputs and outputs of the technology, the production step is again completely generic behaviour re-using existing "building blocks". Agents only produce as much as they have arranged to sell.

Pay bills: The final accounting step is, once again, generic behaviour. All transactions, be it for trading products of transporting goods, are formalised through contracts (and this was already the case in the initial model). The implementation of browsing through all valid contracts and making the financial transactions could be re-used.

A simplification made in the initial agent-based implementation was that each time an agent is asked to sell a certain good, it will ask a random price for it (van Dam, Nikolic, Lukszo & Dijkema 2006). The same simplification is made, even though it should be stressed that it is easy to later add either historical price patterns or more intelligent price setting behaviour, for example based on lessons learnt from the past.

4.4.6 Verification and validation of the model

The chocolate model is based on existing model elements that have previously been used in models that were verified and validated extensively. Furthermore, the source code of the model is documented in even greater detail[7] than for other models, serving as an

[6]The term *strategy* is perhaps slightly misleading in this context because it can have a much wider implication, but this was the word used in the initial implementation to qualify the different roles of the agents and the product(s) they were interested in to buy.

[7]With approximately one line of documentation in the source code for each line of Java code.

example for new modellers of agent-based systems and as an introduction to the ontology. As such, the definition of the model is studied by several people, contributing to an increased confidence in the verification. As with the freight hub model from Section 4.2, this model is not based directly on a real system so no matching of model output with an actual system is possible. However, since the aim of the model is demonstrating re-use of "building blocks" as well as introducing the ontology in a proof-of-concept model and since it is being used in practice for this goal, it can be concluded that it fulfils its purpose.

4.4.7 Conclusions

In this section a model was revisited that was originally used for a serious game (see Figure 4.5) and was then implemented as an agent-based model that shaped the initial version of the framework. It was rebuilt using the current version of the framework (see Figure 4.6).

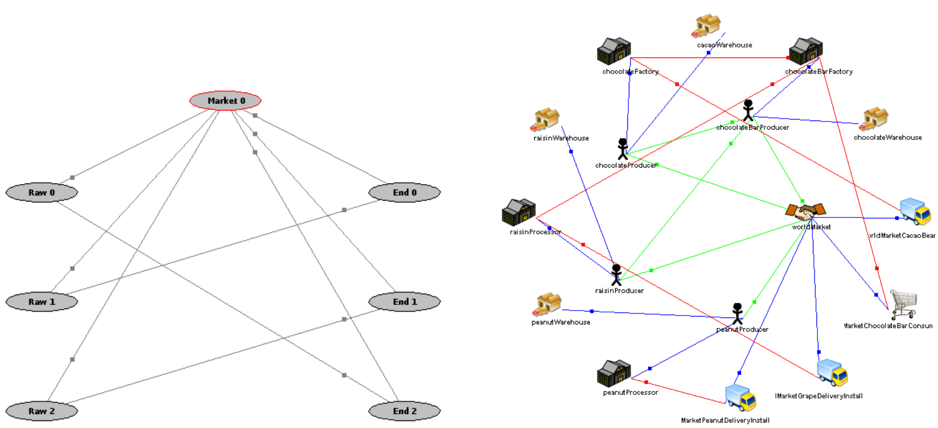
The main difference in implementation between the initial agent-based implementation and the model presented here, is that originally discrete products were used (i.e. individual batches of products, such as parcels of 100kg of cocoa powder) but that, using the new ontology, this was now modelled as flows (i.e. the transportation or production of products over time, such as 1 ton of cacao processed per day). Originally the world market was implemented with storage tank, for example for raw cacao, which was "filled" during the initialisation of the simulation with a certain amount of product that could be sold during the run of the model. The storage tanks were now replaced by *delivery in-*

Figure 4.5 – *Students negotiating about trading goods in the chocolate production game (van Dam, Nikolic, Lukszo & Dijkema 2006)*

stallations (production technology requiring no inputs), making the world market more similar to the producer agents. This makes it possible to use the same generic trading rules and production behaviour. In other words, the world market is a producer agent that does not need to buy any (raw) materials to produce products it can sell.

Conceptually the main difference lies in the strict distinction between social and physical elements that was introduced. Originally agents, modelling the social elements, would also be the recipient of the mass flows themselves whereas now the social edges (communication about trading) and physical edges (actual transactions of products) are separated. This is illustrated with screen shots of simulation runs of the two models in Figure 4.6.

All functionality of the initial implementation could, without much effort, be reproduced with the new model by completely basing it on generic elements. New functionality was added, for example it was easy to now include the Transport agent. In a similar fashion to the explicit inclusion of additional GoodNames, the handling of waste can also be added to the model. This was not done here to stay closer to the initial implementation of the chocolate model (as it would introduce new dynamics), but could be added by creating new GoodNames (for waste coming from all different products) and adding one or more actors that can process waste for a certain price. In the original model this was also possible, but required specific implementation of the waste handling agent while now it can be seen as any other actor which — instead of asking other agents for, for example, peanuts — asks for waste (possible using the labels instead of product names, as described in Section 3.5.3.4, to get *any* waste independent of what product it is from) and instead of paying money for it, it can ask for money to process it. Transportation of waste can be dealt with in the same manner.



(a) *A snapshot of the initial agent-based model of the chocolate game. The ovals represent the agents and the arrows represent a signed contract and subsequent mass flow. On top is the world market, on the left three intermediate producers and on the right three end producers (van Dam, Nikolic, Lukszo & Dijkema 2006).*

(b) *Screenshot of the agent-based model of the chocolate production game, after it was redeveloped using the final version of the framework. The social nodes are on the inner circle and the physical nodes on the outer circle, together forming the socio-technical system.*

Figure 4.6 – *Screenshots of the old and new agent-based implementation of the chocolate game*

With this exercise it was demonstrated that the system described with the initial ontology (after which the iterative development process started with additional case studies) can still be expressed with the new framework and that it is easier now to add functionalities that were more difficult to introduce before. This has brought the loop from Figure 3.1 to full circle.

4.5 Cases by others

The framework has since been used by various other researchers, too. By applying it, these modellers have also significantly contributed to the ontology and the "building blocks" in the framework. This section briefly describes the aims and results of these case studies, but focusses on how the framework was used to help solve a specific problem.

4.5.1 Case 4: Evolution of industrial clusters

In this case the development of industrial clusters is studied. Regional industrial clusters dominated by process industry, such as the Rotterdam-Rijnmond area and Groningen Seaports in the Netherlands, the German Ruhr Area, the Antwerp region in Belgium, Le Havre in France and Teesside in the United Kingdom that largely evolved in the twentieth century must find a way to make timely adaptations to novel and stringent ecologic, economic and supply chain pressures and demands. These include, amongst others, the nascent reduced availability of cheap fossil feedstock, dwindling of suitable metal ore resources, dilution of metal stock, limits to or penalties on CO_2 emissions, and global competition for feedstock, commodity, specialties and pharmaceuticals markets. The aim is to support systematic and rational shaping of sustainable networked industrial systems on a regional scale and to study the effects of the decentralisation of decisions that determine network growth. A managed evolution in both physical and social dimensions is required to shape such systems and ensure their sustainable future and feasibility (Nikolic et al. 2009). To study this evolution of clusters, an agent-based model of this evolutionary process was developed using the framework from Chapter 3.

Conceptualisation of the problem in terms of actors and physical systems For this model all agents are producers in the industrial cluster, with the exception of the world market (as in the cases described above, responsible for delivering goods to the agents and buying products they may sell) and the environment (which collects all waste from the agents, for example CO_2). Every production agent has one or more technologies which it controls and operates.

Clusters between agents[8] are formed based on supply and demand and not pre-defined. A set of possible technologies is defined based on real systems that exist in the industrial clusters mentioned above (and possibly with some future technologies that are developed but not installed yet, to see how they would fit in the cluster) and agents are created that operate one of these technologies. Over time, more agents with more technologies are

[8] As opposed to the other cases described here, no special division between the social and physical system is made, even though they are defined separately. A trade contract between two agents is automatically coupled with a mass flow between two technologies of these two agents and thus it does not matter if one speaks about a cluster of agents or technologies, as they overlap here.

added, either new ones or similar ones to those already in the cluster and their *fitness*, based on profits, is measured.

Refinement of the generic ontology with new abstract classes In this model the core of the ontology is used, including the OperationalConfigurations for technologies and various properties. Compared to the initial ontology, specifically for this case study properties like CASNumber (Chemical Abstracts Service Number, a unique identifier for chemical elements assigned by American Chemical Society) were introduced as well as several labels (e.g. for primary feedstock, by products, limited emission). For the Agent class a new property called *droolsFile* was added to facilitate the use of reasoning rules in separate files (see below). Not all modellers have to use this, but it is an optional extension that is not an obstacle when a modeller decides *not* to use it.

Creation of concrete instances A lot of effort went into the collection of real data from already existing industrial clusters. For example, a complete model of the Groningen Seaports industrial cluster was developed to study the transition to a bio-based industrial cluster (Blokker 2006, Nikolic, Dijkema, van Dam & Lukszo 2006). For all available technologies an instance was created in the knowledge base (with a label to indicate from which cluster the data was collected) so agents could be created *on the fly* with technologies that are actually in the current cluster or that are being considered as suitable additions. This work has lead to a large set of realistic descriptions of the process industry in the shared knowledge base.

Implementation of the behaviour of the agents In the social network, agents can be specified that represent different roles in a production organisation. Agent behaviour for this case can be defined at four levels: identity ("Am I willing to give up some short term profit to develop a more sustainable future?"), strategic ("Which type of technology should I invest in?"), tactical ("Which specific technical configuration will I employ, at which capacity level do I operate the process?") and operational ("Do I have the resources that I need to produce?"). These represent decision making rules and data on different temporal scales. Together with deciding on which activities to perform, the higher hierarchical levels also influence the behaviour of the agent by constraining decisions on the lower level.

There are many different ways a decision can be made. In the operational domain, for example, when selecting a contract, an agent can just choose the cheapest or consider the element of trust or familiarity with the supplier. At the tactical level the decision making determines which of these two modes of operational behaviour are to be employed. Previous experiences of an agent can influence tactical behaviour by choosing different tactical subsets of behaviour at the strategic level. Thus, in the simulation, the overall agent behaviour is a result of the decision making process across all levels and compartments, defined by appropriate rule sets and reasoning algorithms.

The various decision making rules, including more advanced trading rules, are mostly implemented using the Drools business logic language (Proctor, Neale, Frandsen, Griffith, Tirelli, Meyer & Verlaenen 2008) using a Rete algorithm (Forgy 1982), as described in more detail in Nikolic (2009). For each agent a file is created that contains the logic and this file is referenced in the ontology so that agents "know" which rules they have.

Conclusions This model is a major extension as well as a specification of the chocolate game model (Section 4.4), which was designed as proof-of-concept model for industrial clusters. Several extra levels were added, such as the more advanced decision making rules at different levels, but in its essence it is the same model. Using the model described in this section, it was possible to examine, for example, the possible transition paths from a chlorine-based cluster to a bio-based cluster. Detailed results for various cases studies and the specifics of the application of the framework can be found in Nikolic et al. (2009) and Nikolic (2009). Other researchers have already built upon this model by adding data or making additions to the behavioural rules, thus changing the outcomes of the model without having to fully understand all elements of the model and reading all the source-code.

4.5.2 Case 5: CO_2 emission trading

The case study presented next addresses part of the electricity infrastructure, notably the electricity production sector where producers must deal with CO_2 emission-rights trading. The main objective of the case study was to obtain insight in the effect of the CO_2 emission-trading scheme on the types of power plants power producers prefer to build and the power generation portfolio that emerges from their decisions over time (Chappin, Dijkema, van Dam & Lukszo 2007). Because the long term impact of CO_2 emission-rights trading is unknown and serious experience is lacking, Chappin & Dijkema (2009) conjecture that agent-based simulations could help to provide insights.

Governments implement CO_2 emission-trading schemes because it is assumed that it will lead to a less CO_2-intensive generation-portfolio. However, the producers, the agents in the electricity infrastructures, are autonomous. History shows that individual producers do not exhibit the same decision-behaviour. Furthermore, investment and disinvestment decisions are discrete events about capital-intensive pieces of equipment (Chappin, Dijkema & Vries 2009). Agent-based models are suitable for explicitly simulating this. The case study is of an exploratory nature, mainly because of the lack of historic data on emission-trading and its impact (Chappin et al. 2007).

Conceptualisation of the problem in terms of actors and physical systems In the electricity infrastructure electricity producers play a pivotal role. Today, electricity producers must effectively operate in power-exchange markets, but also in markets for fuels, capital and emission rights. The electricity infrastructure is, just like the other infrastructures discussed in this thesis, a socio-technical system: The social network is composed of power production companies, retail companies and consumers that trade on different markets. Governments and regulatory bodies are also part of this network. Power generation facilities, power grids and end-user appliances form the technical network. There are strong interdependencies of the social and technical networks. Electricity producers must operate and invest in power plants respecting current rules and regulations. Consumers invest and operate their end-user equipment. Power grids operated by distribution companies or controlled by government connect the two. Each of these must anticipate and act upon demand, market and regulatory developments expected in interdependent social and technical subsystems.

The model contains the following agents: one industry agent, three markets (electricity, CO_2 permits and fuels), the government, the environment, the consumer and finally six electricity producers. For the physical system there is one industry installation, five fuel delivery installations, one installation for the environment, one consumer installation, and initially thirteen power generation units of different types. During the model run the portfolio of power generation units changes and units are removed as well as added. The number of producer agents is fixed in the model, even though one can envision expanding the model by allowing new players to enter the market etc.

Refinement of the generic ontology with new abstract classes For this case study CO_2 rights were traded, so next to PhysicalFlowContract and TransportContract a new type of contract was added to the ontology, namely an ObjectContract. With this new construct two agents can make an agreement about any object. For this model this was used to make contracts between agents for CO_2 rights. The definition of CO_2 rights is again based on already defined concepts such as GoodName, PhyscalProperties and Units.

Additionally, *multi-criteria analysis* was introduced in the decision making (see below). For this purpose a new subclass of Node, namely DecisionMakingNode, was introduced with MultiCriteriaAnalysis as a subclass. This makes it possible to define — in the knowledge base — what possible alternatives are for a certain decision and what the criteria are to compare them, and furthermore it can be defined which agent uses which method for performing its multi-criteria analysis. The code for this was made generic, so it is not dependent on the domain of this model but can be used for other multi-criteria problems too, thereby contributing to the library of "building blocks".

Until this model was built, properties (e.g. construction costs of a plant) were considered to be static during the simulation run. Because this model needed to take longer time periods into consideration to study the long term effects of different measures, the concept of a "modifier" was introduced to make it possible to vary these values over time. An extra property was added to the ConstructionCost class (an EconomicProperty) and EnergyEfficiency (a DesignProperty), which functions as a modifier. The effect of learning and incremental technological innovation is thus included by gradually increasing the efficiency of power plants and reducing the investment costs of new facilities (Chappin et al. 2009).

Creation of concrete instances Instances for the agents listed under the conceptualisation of the problem in terms of actors and physical systems were created and added to the knowledge base. Furthermore, a large number of possible power plants (from nuclear plants to wind farms and from coal to gas fired plants) with their own properties based on realistic data were created and an initial portfolio for the producers was added (Chappin 2006). Again, case labels were used to determine the set from which the producer agents can choose during the simulation run so that even though the instances from all models presented in this chapter as well as various other models were in the same shared knowledge base, only a sub set is available within the boundaries of the system modelled.

Implementation of the behaviour of the agents For this case the strategic and operational levels are considered. On the strategic level agents make and plan for long term decisions that can affect their long term performance. Such decisions can be investment decisions, but also reconsideration of the company's objectives. On the level of operational management, daily procedures are followed and decisions are taken on a regular basis. This includes negotiation and contracting, buying resources and selling products, for example.

The strategic management (with emphasis on the multi-criteria analysis for investment and disinvestment decisions) and the behavioural steps for the operational management (for bidding on the power exchange, acquiring resources and acquiring CO_2 emission rights) of electricity producing agents are discussed in detail in (Chappin et al. 2007).

Conclusions The last model presented in this chapter shares a lot with the other models, but also added valuable new "building blocks". The model is now being used to test different policy options and it can easily be extended or adjusted, for example to assess the impact of additional policies or to include new types of power plants. A major contribution of this model is the multi-criteria analysis that was added to the ontology. Additions to the ontology made for the trading of CO_2 rights can also be re-used for other types of objects for which a contract between two agents is needed, as it is again set up in a generic and modular way.

Note that in this model different decision making layers are used as compared to the evolution of industrial clusters case from Section 4.5.1. Modellers have full flexibility in *how* to internally model the decision making behaviour of agents, but here they share the same *interface*, allowing the different agents to connect.

Interesting *emergent* behaviour can be observed from this model, showing the power of the bottom-up agent-based approach. Although all agents "live" in the same world and thus are impacted by the same external factors, their actions lead to very different power generation portfolios, because of different management styles. Furthermore, these developments are found to be interdependent: the portfolio development of one electricity producer depends on the other electricity producers.

4.6 Conclusions

In this chapter five different case studies were presented, each with a model developed using the same framework. The focus of the first case studies was not to solve specific problems but to demonstrate how the framework is applied to different types of systems. Still, recommendations can be made already using these models as will be demonstrated in Chapter 7. The cases studies discussed in 4.5 have been executed to find answers to real and hard to solve problems and the use of agent-based modelling has proven to be fruitful and applicable.

With the description of these models, following the modelling steps, it was demonstrated that:

- The framework is applicable to various socio-technical systems in different domains.
- The framework can successfully be used by different modellers.
- The more it is used, the larger the set of shared "building blocks" becomes.
- Elements that were created for one case study can be re-used in other case studies...

- ...even when they are built by somebody else, or are in a different application domain.
- Extensions of the ontology do not *have* to be re-used; classes and properties as well as complete instances can be added to the shared knowledge base without disrupting the work-flow of other users.
- Modellers are flexible in *how* they implement, for example, the decision making aspects in the model.
- People with little experience in modelling can start *using* models quickly and they can make (small) changes to isolated elements (e.g. strategic decision rules) because of the modularity, enabling them to perform new experiments to answer new questions.
- Some new models can be developed without having to make any changes to the ontology and by only re-using existing "building blocks", simply by specifying the instances in the knowledge base and re-using already implemented behaviour.

Next, in Chapter 5, the development of the framework over time is analysed and the relationship between changes in the class structure, the creation of new instances, the development of new models and the application of models to new problems is studied, before showing how some of the models presented here can be deployed for decision support problems in Chapter 7.

Chapter 5

Framework development trajectory — looking back and looking forward

5.1 Introduction

The development of the framework as presented in Chapter 3 was an iterative process: by applying the framework and the ontology to case studies, changes were made that added to the shared conceptualisation and "building blocks" following the application cycle from Figure 3.12. As shown in Chapter 4, during the step "Extension of the generic ontology with new abstract classes" new additions were made specifically for certain cases. As new "building blocks" became available, more and more could be re-used. For some models (e.g. the revisited initial model of the chocolate supply chain in Section 4.4) it was even sufficient to *only* use existing concepts and building blocks, with minor case-specific modifications.

Not only were new elements added to the framework, but also old ones were modified by adding properties or even completely redefining concepts, for example by changing the hierarchy and inheritance. Especially during the initial phase, such redefining of concepts was often needed as only through application in models and by sharing with other modellers the framework could be shaped. The challenge is now to demonstrate that such major revisions are no longer needed to model socio-technical systems and that indeed only smaller additions are needed for new cases. In other words, the challenge is to show that the framework has matured. Furthermore, it should be demonstrated that the framework helps modellers set up new models more quickly because they can base the work on already existing concepts and building blocks.

In this chapter a systematic approach to studying the development trajectory is presented, looking at how the ontology and building blocks have been created and used in various models. The framework itself is tested here, rather than the models built with it. The following indicators are used in this study:

Completeness: The framework is complete when it is suitable for the various domains that fall within the scope of socio-technical systems and new models are built based on already defined concepts. The hypothesis is that most additions to the classes and properties are case-specific and that the foundation of new models consists of existing concepts that are shared with models developed earlier.

Correctness: Concepts that were created have been tested through use and adjusted when necessary. If classes are no longer changed but still used, this is a good indicator that they are correct. The hypothesis is thus that concepts created earlier are not removed but are still being used.

Usability: The usability is established by looking at how others, who may not have been part of the initial development, use the framework and how much effort it takes for new models for new case studies to be built using the framework. The hypothesis to be tested is that later in the life cycle of the framework fewer structural changes need to be made than in earlier stages and more concepts and source code are re-used.

This chapter is structured as follows. First, in Section 5.2, a tool is presented to collect the data needed for this study from various sources as well as previous versions of the framework, so that this data can be used to test the hypotheses subsequently. In Section 5.3 statistics on the development of the ontology (classes, properties and instances) are presented. Next, in Section 5.4, the re-use of concepts from the ontology in various projects and models is dealt with, followed by a studying the re-use of instances defined in the shared knowledge base in Section 5.5. Finally, in Section 5.6 conclusions are drawn on the indicators as presented above and an outlook for future use of the framework is presented.

5.2 Analysis tool

To be able to study the development trajectory, already during the development of the framework data should be collected. Furthermore, information should be gathered about the use of the framework. By bringing this data together and analysing it, the hypothesis as stated in Section 5.1 can be tested. The following information is required:

- Which adjustments were made;
- When adjustments were made;
- By whom adjustments were made;
- Which projects use the framework;
- Which classes are used in which project; and
- Who is working on which project.

One way of collecting this data is using version control software and making sure the work is stored on a central storage server[1]. The server, with version control, keeps track of exactly what has been changed, when and by whom, and offers central access to the

[1]First the Concurrent Versions System (CVS) was used, which was replaced by the more advanced Subversion (SVN) in the beginning of 2007.

source code of the various projects. Modellers and programmers can share a number of *repositories* for storing their work or they allow others to access their repository. This makes it possible to study the history of the development as well as the link between changes in the ontology and changes in specific models, for example.

To collect the data on the development of the framework from Chapter 3, a tool has been developed to support the analysis of the development and to automate comparison between different versions of the ontology and the models that are built using it. Next, the design of the tool (Section 5.2.1), data collection (Section 5.2.2) and how missing values are dealt with (Section 5.2.3) are discussed, followed by a discussion on lessons learnt from the development of the tool (Section 5.2.4). Afterwards, from Section 5.3 onwards, the data collected with this tool is presented and conclusions are drawn based on this.

5.2.1 Design of the tool

First a tool is built to automate the comparison of different stages in the life of the ontology, independently from the development of models that employ the ontology. The steps taken by this tool are:

1. Check out the first[2] version of the ontology from the SVN server.
2. Collect data by performing metric calculations on the downloaded files.
3. Store the results in a convenient way.
4. Iterate and check out the next revision from the SVN server.
5. Stop when the last revision has been reached.
6. Export the results to a graph.
7. Print an overview of which classes were added when and by whom.

5.2.2 Data collection

To study the growth of the ontology, the *number* of instances, classes and slots of the ontology are collected over time (the date and time of committing the new version to the server) or over revision number (the "version number" of the collection of files stored on the central server). Plotting these changes per revision number has the advantage that it illustrates how much work is done before a commit, which generally means work was done by one person. On the other hand, these plots give a false impression of development over time, because in times of concentrated development there will be many revisions within a short time frame while in other phases there are only few. Therefore it is important to collect the time on which each revision was committed to the server. In this chapter only data collected over time is analysed and not the changes over revision number.

[2]The tool is structured in such a way that it is possible to only check a subset of revision numbers, for example only those revisions committed to the repository by one author or those done in a certain time frame. The first revision does therefore not necessarily have to mean the first revision available on the SVN server, but the first revision from the set.

Using the number of instances, classes and slots as metrics has three disadvantages which are discussed below:

1. It does not show changes in values;
2. It does not show when an element was removed and replaced by another one; and
3. All changes are considered equal.

When a user decides on *changing a value* of a property, (e.g. changing the efficiency of a power plant) and a new revision of the knowledge base is committed, this causes no changes in the number of instances, classes and slots. While such modifications can completely change the outcome of simulations, they do not change the framework itself. It is therefore no obstacle that the indicators used here will not make these actions visible.

When only counting how many instances, classes and slots are in the knowledge base, it does not show when classes are removed and *replaced* by others within the same revision: the total number can remain the same. Furthermore, all changes are considered equal this way: one small change might not jump out in a graph if a larger number of changes was made the next day, but this small change can have bigger impact. For example, restructuring a major class (e.g. Technology) by adding a new subclass requires no new slots, but only a single new class. This could be a very important change with major impact on how models are, or should be, developed in the future. Nonetheless, it can be said that a well-thought of design of the ontology and models means that even a relatively large change in structure might have limited influence on existing models: as long as no classes are removed but only new ones added, it does not cause existing models to break.

To study the restructuring, not only *how many* changes were introduced, but also *which* changes have been made, should be analysed. For this purpose, in addition to the number of instances, classes and slots, for each revision also the list of all classes is compared with that from the previous revision. This way an image of which classes were added or removed is created. For each class it is also stored by whom and when the change was made, so that a more detailed analysis can take place.

The migration from a CVS system to SVN for version management and the subsequent taking offline of the old server means that only the changes from February 2007 onwards can be taken into account by this tool. While it could be interesting to study the earliest developments of the framework, the migration to SVN coincided with the phase in which the framework was made widely available also to people who were not involved in the early development, making this the most important phase to study for the purposes of this chapter. The first version on the SVN server can be considered as a first *base version* of the framework.

To summarise, the tool presented in this section collects data on the development trajectory by recording the number of instances, classes and slots over time as well as data on which classes and slots were added when and by whom.

5.2.3 Missing values

It is possible that a committed version of the ontology cannot, after downloading, be opened. There may be various reasons for this. For example, the file could have been corrupted by the editing tool or conflicting versions were merged in an inappropriate way. The file can also (accidentally) have been deleted/moved and thus cannot be found on the repository. Additionally, in rare cases it is possible that even though the revision can be read it should be omitted from the data set, for example because it may give a false image of the development when a large number of additions or deletions were made that were undone right afterwards. In such a case there are *missing values* in the data collection for a certain revision number and date. There are different options of dealing with this:

- They can be set to 'null' (i.e. no value, creating a gap in the graph). This will not be visible in a graph when the number of revisions plotted is large.
- Set the value to -1 (or another value that is not feasible), so as to distinguish from 0 occurrences. The advantage is that it is clearly visible, but a disadvantage is that it is distracting and draws the attention away from what is really important.
- Ignore them completely because these missing or broken revisions say nothing about the actual state of the framework, but only of the state of the framework on the SVN server.

Here the last option is chosen because the main focus is on the growth of the actual ontology, and not the files on the server. A different choice would be made when studying the use of the version control server as part of the practical arrangements for working on an ontology is the main focus. In that case, those revisions where data cannot be read are especially interesting and should be studied. For the current study, however, such data points only distract from what is important.

5.2.4 Lessons learnt on the analysis tool

Version control turned out to be extremely useful — not only during the development process itself but also for analysis afterwards. In addition to offering control of different versions, an overview of all changes (and possible causes for bugs), easier sharing of code between developers and extra flexibility when working from different locations (e.g. university, home, during conference trips, etc.) as well as secure backups, the use of version control also allows analysis of the development efforts ex-post. Most of the other advantages could be obtained in different ways, but for studying the development over time version control is essential. It is therefore recommended future projects continue using it.

While version control was used to keep track of all changes, not all commits were well documented. This is a lesson learnt for future projects, because proper documentation makes analysis of what has happened much easier. Now it could be analysed *how many changes* were made and with some extra effort it was included *what* has changed, but finding out *why* this was done is much more difficult and resulted in the need for additional (manual) checks, as will be shown in Section 5.4. If all changes are well documented it could already be interesting to simply study the log messages or consult them when unexpected patterns are observed to find out why something was removed or added. Developers should be, even more than now, encouraged to document their work also through log messages.

A possible extension of the analysis tool is to trigger it automatically each time a commit is done, for example by running it on the SVN server itself. This should not affect performance of the server. The output can be generated and committed automatically too, so that it is immediately available for a view of the latest state of the framework.

Instead of running it for all older versions, the collected data should be stored and only the new version added. Currently the tool, during each run, analyses all revisions and not only the latest addition, but already collected data could be stored so it is available still for a next check as it cannot change any more.

5.3 Ontology development

This section deals with the development of the abstract parts of the ontology, namely the class and slot definitions.

5.3.1 Results

The growth of the number of classes and slots in the ontology, forming the formalisation of the abstract concepts and their properties, is shown in Figure 5.1. These concepts and properties are used in the framework for the definition of concrete instances and the behavioural rules in the models. At the start of the analysis there were approximately 150 classes and 90 slots. Both numbers have nearly doubled in the more than two years of development that followed the initial phase.

It can be observed that the growth usually takes places during short periods in which many new additions are made, after which the number is stable for some time again. Furthermore, changes in the number of slots and the number of classes often go hand in hand, but not always. There are periods where the number of classes steeply increases while the number of slots stays more or less the same, or the other way around.

A list of which classes were added or removed — and when this happened — is also generated. The results can be found in Appendix D. One can see that only few classes are removed, and if a class is removed it is very soon after it was created.

5.3.2 Conclusions

When analysing the link between the number of slots and the number of classes, several interesting things can be concluded. During some time periods, more classes are added but the number of slots stays the same. This can partly be explained by the fact that many new classes are a subclass of Label which does not have any slots, but is used to "tag" other classes. However, there are also many situations where new classes are added that inherit properties from their superclass and do not need any new slot definitions, or when already existing slots are added to the newly created class.

A more interesting situation occurs when new slots are defined but the number of classes does not increase. While a few such occasions can be found in the data set, it is not very common and the effect is not very big. Still, in that case it can be said that the existing classes became richer, because new properties were attached to them so the instances of these classes can contain more information. This does not, however, mean that a new slot has to be used by all modellers that use instances of the class or classes the slot belongs to: the richer ontology still contains the expressive power that is needed by those models that only rely on the more sparse class definitions. Other modellers using the same shared ontology only benefit from such additions, but they do not lose anything. This makes the design of the ontology flexible enough for new class specific modifications.

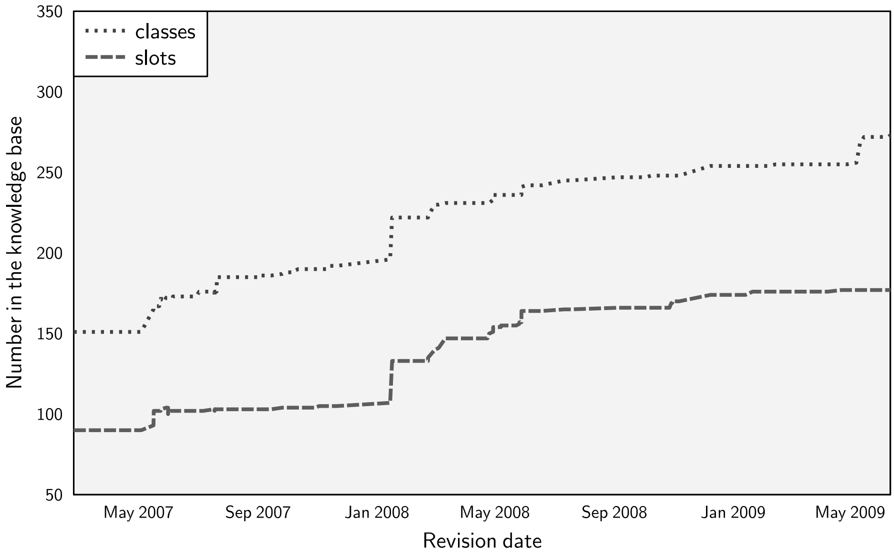


Figure 5.1 – *Growth of the ontology in number of classes and slots over time*

It is important to note that classes added to the ontology at a later stage are, indeed, more case-specific. On top of the list in Appendix D more generic classes such as Ownership and PhysicalNode were introduced and the concept of Flow was removed and replaced by PhysicalFlow and labels such as Fossil or Renewable were added. Going down towards the end of the list, one can see for example the case-specific classes for a computer game to support education. The link between when classes were added, and how often they are used, is studied in more detail in Section 5.4.

Finally, it can be concluded that the number of changes in the structure has stabilised. After an initial period of strong growth (especially the period just before the versions accessible by the analysis tool, when the base of the class structure was built), now a steady growth can be observed. In any case, the development has not stopped: the framework is still being used and is still being expanded.

5.4 Re-use of concepts in practice

In Section 5.3 it was shown that the number of concepts in the ontology has nearly doubled during the development. In this section their use in models is addressed. An additional analysis tool was developed for this purpose. This tool downloads the latest version of the source code of all projects that are on the SVN server and for each project it scans all Java files that are part of the project. In a Java program, before a class can be used, it has to be specifically imported[3]. This fact was exerted by the analysis tool that

[3]In Java a generic import can be done by using a '*' instead of the name of a class to get all classes in a certain package. That means that it cannot be deduced from the import statement which classes it entails. If

parses all files to collect these import statements, thus creating a set of all classes that are used within one model. By adding the sets for all projects, an overview of which project uses which classes, and which class is used by which projects, was generated. For each class found this way the date it was first added to the ontology was looked up.

5.4.1 Results

In total 21 different projects that use the ontology were analysed[4]. An overview of these projects can be found in Figure 5.2b and they are briefly described in Appendix C. Below is the top-15 of most used classes and their use, as part of the output of the analysis tool.

- Agent (20 uses);
- Technology (19 uses);
- UnitName (19 uses);
- GoodName (18 uses);
- Node (17 uses);
- Edge (16 uses);
- PhysicalConnection (16 uses);
- PhysicalFlow (16 uses);
- ComponentTuple (15 uses);
- Contract (15 uses);
- Mass (14 uses);
- Price (13 uses);
- EconomicProperty (13 uses);
- CaseLabel (12 uses) and
- Ownership (12 uses).

All the classes in the top-10 were created *before* time period I in Figure 5.2b (i.e. before the first revision available on the server. A number of classes in the top-15 were created *in* period I. No classes created at a later date have found such widespread use. In general, the lower a class is on the list, the more specific it is and the more recently it has been created.

such an import statement was found by the tool it was manually removed and replaced by an import of only those classes which are used in the Java file.

[4]Note that some of these projects are still not finalised and may be work in progress.

5.4.2 Conclusions

The concepts in this top-15 are all classes that are being used in at least 50% of all projects so far. These results underline the completeness of the base set: no new classes were added that the majority of models needs. The most used classes are also those described already in Chapter 3. Classes added later are, as said in the hypothesis, more case-specific.

From the 21 projects considered, only one does not use the concept of an *Agent*. Detailed analysis of the list of which projects use which class revealed that this one project not using the Agent class is, indeed, not an agent-based model: it is an electricity market simulation game that is completely based on the ontology, but that uses human players instead of agents (de Vries & Chappin 2009, de Vries, Subramahnian & Chappin 2009). However, also adding computerised players (i.e. agents) to the game was left open as a possibility and is fully enabled because the game is using the framework for socio-technical systems and is thus expandable.

A remaining question then is whether concepts created for specific case studies are later re-used in other models. Already in the list it can be seen that several of the more recently created classes are already used in more than one model, but it would require further analysis to make a clear statement about re-use of concepts which were initially case-specific.

Finally, it should be mentioned that not all classes in the ontology are used in models. This can be explained by the fact that often only a subclass is used while the superclass is merely an abstract container. Only the more specific classes are used. In other cases (e.g. a Contract) it can in fact be useful to reason about a collection of subclasses by using the superclass name (e.g. to pay for all Contracts, regardless of what subclass of Contract it is).

5.5 Re-use of instances

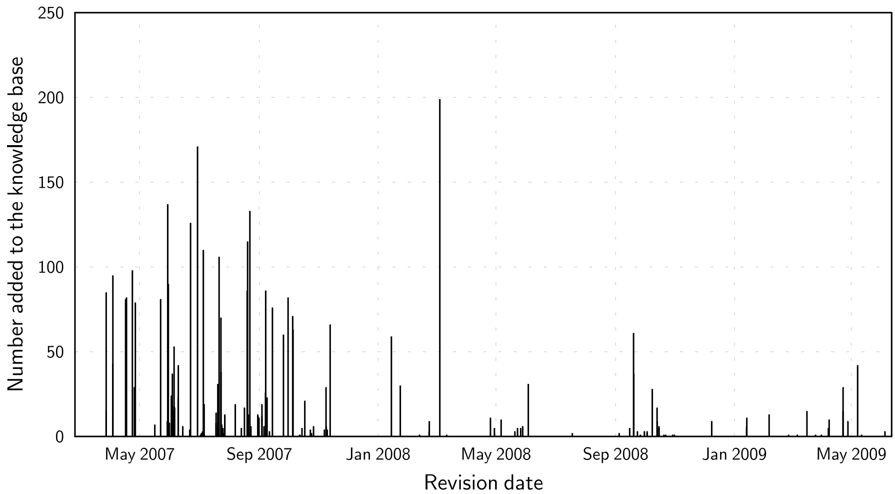
In this section the development of the instances in the ontology is studied and compared with the time planning of projects that use the framework for various purposes. The tool from 5.2.1 produced the data on the ontology development. Furthermore, a list of all projects that use the shared ontology was made and the start and end dates were logged.

For this study all negative growth (i.e. instances that were removed from the knowledge base, for example during clean-up of unused instances and removing those that were created more than once by mistake) were considered as 0. Some of the additions may have been undone later so the results slightly overestimate the growth of the number of instances, but distraction from what is important to study is avoided this way.

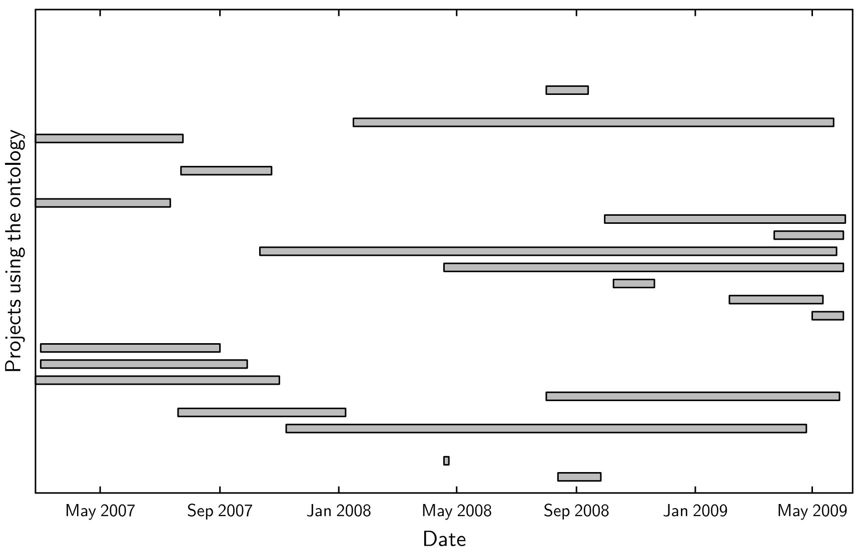
5.5.1 Results

The development of the instances in the shared knowledge base by different modellers and the time schedule of projects in the same time frame is shown in Figure 5.2. Figure 5.2a shows the individual efforts of modellers[5] and it can be seen that significantly more instances were added in first periods compare to those afterwards. Detailed study of

[5]By September 2009, a total of 24 modellers have already contributed to the shared repository and 20 of them have made changes to the shared knowledge base and the ontology.



(a) *Instances added to the shared knowledge base over time*



(b) *Time planning for projects that use the ontology. Every row is a separate model developed for a specific problem, with the bar indicating approximately when the project started and ended*

Figure 5.2 – *The addition of instances to the shared knowledge base and an overview of projects that use this shares knowledge base*

who made which changes reveals that different people use the ontology now than in the beginning, while others have actively contributed over the full time horizon.

The time planning for a selection of projects that use the ontology, namely those projects that were active in the same time frame as analysed by the tool use in this chapter, is shown in Figure 5.2b. In Appendix C these projects are briefly described. Figure C.1 shows the chart for all projects, instead of only those that fall within the boundaries used here. In Table 5.1 the number of projects for each time period is listed, based on the data from Figure 5.2b.

In Figure 5.3a the cumulative growth of the number of instances in the shared knowledge base is displayed. This growth is the result of the individual actions that were shown in Figure 5.2a. Studying the graph with total number of instances in Figure 5.3a and comparing this with the development of the number of classes in Figure 5.3b, it can be seen that changes in the structure are directly followed by a period of increased activity on the instances. Furthermore, changes in the structure occur in steps, while the addition of instances proceeds more gradually.

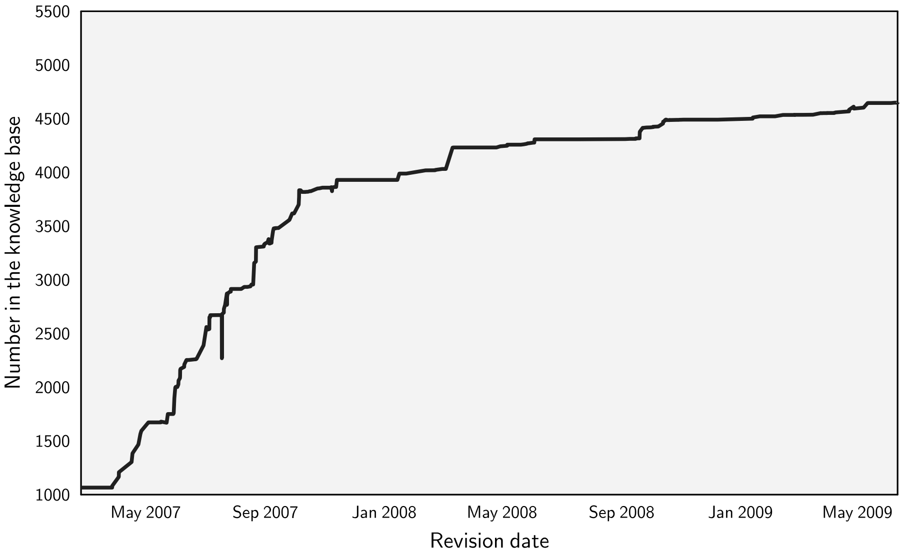
5.5.2 Conclusions

Studying Figure 5.2a in relation to Figure 5.2b and the number of active projects during different time periods from Table 5.1, it can be seen that during the first half of the time frame (periods I, II and III) fewer projects were active but more work was done on the collection of instances. In the second half (periods IV, V, VI and VII) more projects were active but less instances were added to the ontology. From these results it can be concluded that:

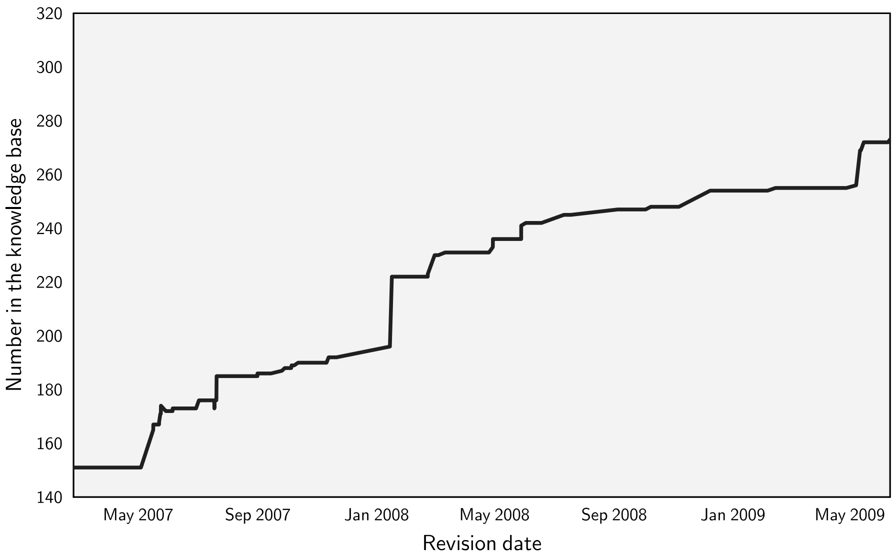
- New models are being developed but the modellers do not have to add as many new instances as in the early stage of the framework.
- Without the bulk of the work done in the earlier stage, the creation of instances would have to be done later.
- Different people use the ontology now than in the beginning, but they re-use the work that was done earlier and build upon that. Among those that have stopped using it are students who used it for their MSc project, for example.
- At the start of a project the abstract definitions are expanded for case-specific projects (see also Figure 3.12), after which instances can be added for this case (if needed). A time lag between change in structure (i.e. classes) and content (i.e. instances) may be observed.
- Less structural changes are needed to incorporate new instances.

Table 5.1 – *Time periods and number of active projects*

Period	I	II	III	IV	V	VI	VII
Number of projects	5	6	5	6	9	7	9



(a) *Growth of the ontology in number of instances over time*



(b) *Growth of the ontology in number of classes over time*

Figure 5.3 – *The growth of (abstract) classes and growth of (concrete) instances of these classes*

It is not possible to know exactly which instances are re-used between different models, as the instances are only read by models at *run time* and it cannot be fully deduced from the source code which instances are loaded. Furthermore, it may require many different runs of a model with different parameters for a certain instance to actually be used even if it is known that the model uses it. However, the fact that there are more models now, while less instances are added, strongly suggests that many instances are in fact re-used.

To give a concrete example, this means that instances of electricity production facilities added for a model of the electricity sector in one project were later not only used in a new agent-based model of the energy sector to study possible behaviour of actors and responses to different policies, but also in the serious game that is used for educational purposes to teach about bidding and investment strategies. Because a shared and formal language was used, the initial investment of collecting the data and adding it to the knowledge base later paid off when it could simply be read and re-used.

5.6 Conclusions

The ontology has grown since early development to a now stable set of classes and slots that can be used to describe a still growing number of cases. The ontology is still being expanded, but no fundamental changes to the structure are made and have been made recently. New concepts are being added, not replaced and the new concepts added are also of a different nature and scope from those in the base version. Without exception, the classes in the top-15 of most used classes that was presented in Section 5.4 have all been created in the early stage of the development when the fastest growth took place (See Figure 5.3b). There is a strong relation between the date of creation of a class and the number of projects that use it. Together with the fact that by far most models were not developed until after the classes from the top-15 were defined, one can conclude that the foundation of the ontology, which is shared by most models, is stable and complete.

Starting with a number of initial models in different domains, sometimes only conceptual or illustrative, has been very fruitful. It allowed the development of a generic set of concepts which, as demonstrated in this chapter, have then been re-used in a large number of models outside the scope of the original case studies. It is therefore recommended that a rapid development approach, in which a number of illustrative models is quickly set up using the ontology and by immediately feeding the changes back into the framework, is followed. When the circle of developers is still small it is relatively easy to make large changes that — at this stage — can still have major impact on existing models. At later stages of the development it is no longer acceptable that a change in the ontology, for example, causes older models to no longer work. If it is possible for models to fail because of "improvements" done by others this would strongly discourage people from participating in a shared development involving a shared knowledge base.

The hypothesis that most recent additions to the ontology are case-specific, was tested positive, which leads to the conclusion that the framework as presented in this thesis is complete. Furthermore, no major concepts created in the early stage were later removed or replaced, while they have been used in large number of projects which means the framework is correct. Finally, it was demonstrated that later in the life cycle of the framework fewer structural changes as well as fewer new instances were needed while the number of projects increased, proving that using the framework for to build models of socio-technical systems significantly reduces the effort required for new developments.

Chapter 6

Benchmarking

This chapter is based on van Dam, Adhitya, Srinivasan & Lukszo (2009).

6.1 Introduction

A critical evaluation of the advantages and disadvantages of the framework presented in Chapter 3 and a detailed comparison with other modelling paradigms is called for. By building different models and looking, for example, at how they are built and how they can be expanded, a well-founded justification of the choice of modelling paradigm can be found and recommendations and guidelines on which paradigm is more suitable for which application or problem can be given. As concluded in Section 2.4.9, equations can also be used to model decision making of individuals (e.g. Ortega-Vazquez & Kirschen 2008) and this has traditionally been the modelling paradigm of choice.

In this chapter the model of an oil refinery supply chain is taken as a case study (See Section 4.3). It is suitable for this purpose as it comprises complex interactions among a number of decision-making actors and physical processing equipment. There are different options available when choosing an appropriate modelling paradigm for supply chains. Traditionally, supply chains have been modelled with equation-based models (Sterman 2000) but more recently the agent-based paradigm has received much attention in this field (Chaib-draa & Müller 2006).

Others, including Parunak, Savit & Riolo (1998), Borshchev & Filippov (2004) and Tang, Parsons & Sklar (2006), have attempted to perform a similarly motivated comparison between equation-based models and agent-based models. However, in all these papers a clear definition of what is being compared is missing. Inconsistency in the definitions of modelling paradigms has led to a situation where conclusions from one author are used as an unfair justification for the choice of a certain modelling paradigm by others.

Parunak et al. (1998) were the first to compare agent-based models, then a very new field of research, with the traditionally used equation-based models. They write that "understanding the relative capabilities of these two approaches is of great ethical and practical interest to system modellers and simulators". The case study they use is that of the (relatively simple) Forrester supply chain. In their study, the equation-based model uses differential equations, while the agent-based model is more detailed, comprising different classes of agents (e.g. company agents representing the different companies in the supply chain, shipping agents used to model transport with uncertainty and delays, etc.).

After performing a comparison between the models, they conclude that agent-based models can be applied to all domains that traditional models have been previously used in, and that there are some advantages such as a more natural fit, ease of construction, support for more direct experiments, and the ease of translation back to practice. Perhaps some of these benefits do accrue, but good measures or indicators to establish them have been hard to come by. Even though it does not compare two models made for the same purpose, the article's conclusions (fuelled by the fact that the term equation-based model used in the title is much broader than the system model examined in its body) are often used to justify the application of the agent-based paradigm.

Chatfield, Hayya & Harrison (2007) discussed the use of different formalisms combined, to take advantage of strong points of each: "Forcing modelers to conform their understanding of a subsystem to an unnatural viewpoint may lead to added model building difficulty. For example, agent-based concepts are easily mapped to some supply chain entities and actions, such as basic supply chain participants (retailers, warehouses, etc.) and their behaviors, but are not suited for other areas of the supply chain, such as process items (materials, orders, etc.)"

Macal & North (2005, 2002) developed an agent-based implementation of the beer game (a frequently used case study for research on supply chains), based on the original System Dynamics model. They claim that their results "exactly duplicate" Sterman's (1989) equation-based model which once again demonstrates that dynamics of an equation-based model can be captured by an agent-based model. While insightful, the results from these comparisons are obtained without resorting to a well-defined approach and without clear definitions of what is compared, making it difficult to generalise the findings.

It should be stressed that comparing modelling paradigms based only on the conceptual model specifications is not enough; rather a well-defined benchmarking process and the execution of experiments are required. In this chapter, a strategy for executing benchmarking studies of modelling paradigms is presented. A benchmarking exercise is executed for two models (each motivated by a different paradigm) of an oil refinery supply chain. The lessons learnt will be applicable for modelling supply chains, but can be generalised to other socio-technical systems, for example in the infrastructure domain.

The rest of this chapter is structured as follows. The corner-stone of this chapter is the detailed benchmarking process described in Section 6.2. Two additional models of the oil refinery supply chain case study (from Section 4.3) are presented in Section 6.3. The benchmarking process is then applied to the refinery supply chain models and conclusions from the exercise are drawn and recommendations for the use of the different modelling paradigms are given in Section 6.4. Finally, in Section 6.5, the chapter is summarised.

6.2 Benchmarking

Benchmarking is about making comparisons and, through these, learning generalisable lessons. It is not possible to compare modelling paradigms based only on the conceptual model specifications; rather a well-defined benchmarking process is required.

In order to assess the performance of the two modelling paradigms, the following scheme, inspired by Monch (2007) and refined in van Dam, Adhitya, Srinivasan & Lukszo (2008) and van Dam, Adhitya, Srinivasan & Lukszo (2008), is adopted:

1. Definition of the objectives for the study
2. Identification of what is to be benchmarked
3. Evaluation if objects of study are comparable
4. Determination and specification of performance measures
5. Description of scenarios (well-structured experiments) and their simulation
6. Conclusions

Next, each of these steps will be explained.

6.2.1 Definition of the objectives for the study

In the first step of the benchmarking process, the objective for the study has to be defined. Examples of suitable objectives are:

- Choosing the fastest model for online decision making support
- Learning the advantages of one modelling paradigm over another
- Justification of choice of modelling paradigm for a new project
- Testing a new modelling platform with a traditionally used or well validated approach

A clear definition enables selection of appropriate performance measures in step 4.

6.2.2 Identification of what is to be benchmarked

Next, the objects of the study should be identified. Or in other words: what is going to be benchmarked. This has to be clear and detailed so others are able to reproduce the experiments, if desired. This adds to the transparency of the benchmarking study. The objects of study should be specific models that have been implemented and that can be used to perform experiments. This step can for example refer to a detailed model description or even include the source code. It is assumed that the objects of study are comparable to make a successful benchmarking study, something that is evaluated in the next step.

6.2.3 Evaluation if objects of study are comparable

Valid and useful conclusions can only be drawn from a benchmarking study when it has been demonstrated that the objects, defined in the previous step, are actually comparable. It is not easy to say when models are comparable and even more complicated to say that they are *equivalent*, if this is at all possible. Perhaps it is better to talk about models being similar. But what is similar enough to justify saying that they are comparable?

Inspired by Sterman (2000), distinguishing between model verification and result validation, models are examined at different levels to determine if they can be called comparable:

1. The definition of the models that produce the results (*model verification*);

2. The numerical results of the model for (selected) output variables (*result validation*); and

3. The *decision* that follows from studying the results of the models.

The first step, *model verification*, deals with the way the models are formalised using equations and algorithms. It can be checked whether the model has been coded correctly and consistently. For all models the model definitions can be compared and it can be demonstrated that, even when a different *language* is used for the system description, models encode the same *behaviour*. This stage of testing has to some extent overlap with the specification of the model.

Next, in the *result validation phase*, the model behaviour is studied by comparing the numerical results of the model (e.g. for extreme conditions or other pre-defined conditions). If the definition of models is exactly the same, one can assume that if there are no model-dependent errors, then the same numerical results can be achieved as well. However, it can happen that the model definitions are precisely the same, but due to numerical-method-dependent differences (e.g. different solution methods for solving differential equations) or different random numbers in the case of stochastic models some dissimilarity in the numerical results can still be observed. The results can be exactly the same, or differ within accepted boundaries. If the numbers are the same, or close enough (e.g. only very small constant shift, small phase or amplitude difference; again, this has to be well defined), one can consider the output of the model run to be the same for that situation. Of course, that does not mean the model will behave the same in other situations, too. Furthermore, models can be deterministic (meaning that every run of the model will produce the same outcomes) or stochastic (in which "chance" or random elements are introduced so that the model output is not always equal to the output of the previous run). For stochastic models even when multiple runs of the models are averaged out, it still cannot be concluded that the output will be the same as well. But after looking at the definition of the models and how the results are produced, it is possible to get a better understanding of how models will behave under different inputs and different scenarios.

The last step compares the *decision* made using the models. Building models and running simulations is not an end by itself, but rather the means to reach a decision (e.g. policy recommendation). The values for the output variables can lead to a decision or recommendation (dependent on the goal of the model) and these can be compared. For example, do the models recommend the same policy to be deployed (qualitative similarity), even if they predict different profits based on them (quantitative similarity)? If this is the case, the models can be said to be comparable. The same problem can be observed as with the numerical results: if the outcomes are the same for one or more scenarios, it does not guarantee they will be the same for other scenarios too. If the simulation, used as a decision support tool, leads to the same decision made based on both models for one case, there is no immediate guarantee that this will also happen in a different scenario. Repeating this exercise for many scenarios and with varied data sets gives more confidence in the similarities, but still does not *prove* that the models produce the same outcomes.

Proving that the same numerical results and the same recommendations will be produced in every scenario can only be done by comparing the definition of the models and concluding that they are the same. From this it can be deduced that the output will be equal, too. However, it should be stressed that this is not obligatory for getting useful results from a benchmarking study: when the models are comparable for certain scenarios, the benchmarking conclusions will still be useful and applicable for these scenarios. These results can still be generalised as a hypothesis that has to be tested for other scenarios.

6.2.4 Determination and specification of performance measures

To determine the performance measure for the different ways of modelling that are being compared it is essential to, besides a comparison of the outcomes, also reflect on the modelling exercise as a whole. For instance, it is now widely accepted that the ease of developing the model and maintaining it over the lifespan of the application is an important (sometimes critical) determinant in successful industrial acceptance (Cameron & Ingram 2008). Therefore, in addition to comparing the numerical simulation results from the two models, other qualitative key performance indicators are also considered.

Cavalieri, Macchi & Valckenaers (2003) describe a benchmarking service for different users of control systems (e.g. researchers, vendors as well as practitioners from the industry) and performance is evaluated in terms of efficiency, robustness and flexibility. The same indicators are used here. Considering efficiency, the ease of expressing the problem in each modelling paradigm is considered. For robustness the possibility of extending the models can be compared and for flexibility their re-usability. Inspired by the work of Cavalieri et al. (2003), an additional performance indicator is formulated: the ease of explaining the model and its applicability.

In the following sections these performance measures are discussed in more detail.

6.2.4.1 Ease of expressing the problem

The first step while creating a model is making a conceptual specification of the system. Subsequently, it can be operationalised by implementing it using one or more software tools, resulting in a computational model that can be used for simulations and experiments. The ease of conceptualisation of a system into a model through a paradigm is problem-dependent. This could be measured in amount of time spent in expressing the problem from conceptual design to implementation, but one should also consider if all aspects can faithfully be included. Some aspects can be quickly implemented in one approach, but others may have to be left out or simplistic assumptions made. It is much more difficult to assign an exact measure for this; the ease of expressing the problem is therefore mostly subjective.

6.2.4.2 Ease of extending the models

The ease of extending the model can also be considered as a subjective measure, but here an attempt is made to put a value to a set of indicators. For possible or desired extensions to the model (such as adding certain functionality to the model or including new elements of the system that is modelled) a prediction can be made of the tasks that are required to bring them into being. For each of these tasks a judgement can be made on how difficult, time consuming, and risky (to the functioning of other parts of the model) these extensions might be. The following are typical indicators that could be used to assess the ease of extending models:

- Number of changes required
- Expected amount of time needed to complete the tasks
- Number of (self-containing) model elements in which changes have to be made
- The chance that something goes wrong
- Difficulty level of the tasks

Note that some of these indicators may be more difficult to objectively compare than others. Factors such as the knowledge and experience of the modeller have to be taken into account, for example. Furthermore, many tasks require teamwork or cooperation with other modellers so problems with communication between them may arise, also.

6.2.4.3 Ease of re-use

An important aspect of model development, and software development in general, is the re-use of parts that have been developed previously. This could save time and resources, because earlier work can be integrated in a new project. For models, however, there is an even more important aspect to re-use: it allows models to be built upon previously verified and validated components. If parts of models are used in different case studies and they are verified in each of them, it increases the modeller's confidence in their reliability. Furthermore, when parts of a model can be re-used it becomes easier to create larger models to study larger and more complex systems as well as interactions between different systems.

6.2.4.4 Ease of explaining

When used as a decision support tool one of the most important aspects, in addition to the validity of the model, is how easy it is to explain the model and interpret the results. As stated by Zee (2006) "a fundamental challenge in simulation modelling of manufacturing systems is to produce models that can be understood by the problem owner".

Especially when the model-builder is not the same person who makes a decision based on its results, it is critical that the model outcomes can be explained so the right interpretation can be made. It can be dangerous to simply trust the model output and focus on the numbers and predictions. Instead models should be used to gain insight about *possible* future states as well as possible effects of actions. Again, it is not possible to put a single quantitative indicator to the ease of explaining – it has to be analysed by testing with users and their innate preferences for different descriptions (e.g. equations, algorithms or diagrams)

6.2.5 Description of scenarios and simulation

A benchmarking study is based on a number of scenarios that can be executed with all objects of the study. How this is done is completely case-specific, but a reproducible description should always be given. Such a description might include what will be measured and which settings are used. If the objects that will be benchmarked are comparable, then the same scenarios should be used in all objects of study. Once the performance measures have been specified, the last step is to perform the simulation and evaluate the results, with the aim to learn from this benchmarking exercise and to produce a number of recommendations about the applicability and use of the approach, based on the key performance indicators (Section 6.2.4).

6.3 Models of the oil refinery supply chain

In this section two additional models of the supply chain from Section 4.3 are described, each occupying a different place in the modelling space shown in Figure 2.1.

The first model is a high-level low-resolution model that can deal with throughput related equations (Section 6.3.1). It describes mass flows and economics of the supply chain, but lacks the elements to describe the decision making involved in the supply chain operation, for example, crude procurement. To be able to add more detail to the model one needs to consider an "individual" level description. This is done in the second model (Section 6.3.2). This can be considered as moving left on the horizontal axis in Figure 2.1 from system observables to individuals. The decisions that have to be modelled then *require* the use of algorithms, for example for the selection of which crude to buy and at which production mode to operate the process. This consequentially means moving up along the vertical system description axis, from equation to algorithm. The logic involved in transferring crude between the VLCC and the storage tanks, through the jetty, can be best described through algorithms, concentrating on the actor that *causes* each transaction (as was described in Section 4.3). Afterwards, the distances between the models can be visualised in the modelling space.

6.3.1 Model E: An equation-based model in Excel

The first model considers only system observables and not individuals, using *only equations*. This set of equations can be implemented in a simple spreadsheet such as Microsoft Excel (hence Model E). This model is at the bottom right of the modelling spectrum, i.e. Quadrant IV (See Figure 2.1). Modelling of decisions requires algorithms, but algorithms are not used in this model. For example, an "if-then" algorithm is employed in a reorder point procurement decision: if crude inventory is less than x, then order y amount of crude z. Decision making entities are thus not captured in this equations-only model.

Furthermore, as information flows mainly serve to control material flows in the form of decisions, they are not modelled. As a result, the refinery supply chain considered in this model has to be simplified, as shown in Figure 6.1 (cf. Figure 4.3). The refinery receives crudes from the suppliers and stores them in the crude tanks. The crudes are processed in the processing units and converted to valuable products stored in the product tanks. Finally, the products are delivered to the customers.

In this model, the supply chain is modelled through equations involving system variables. Let IC_{lc} be the inventory of crude c at the beginning of cycle l, CA_{lc} the amount of crude c arriving at the refinery from the suppliers in cycle l, and TP_{lc} the amount of crude c processed in cycle l. Equation 6.1 describes the mass balance around the crude tanks:

$$IC_{(l+1)c} = IC_{lc} + CA_{lc} - TP_{lc} \tag{6.1}$$

As crude procurement, transportation, and unloading are not explicitly modelled, their effects are captured in the parameter CA_{lc}, which is an input to the model. Similarly, the

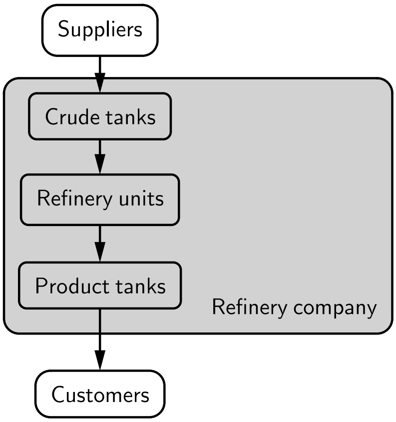


Figure 6.1 – *Simplified schematic of refinery supply chain as used in Model E*

decision on production throughput requires algorithms and is not modelled, thus TP_{lc} is another model input.

Conversion of crudes to products in the refinery units is modelled through simple yield calculation:

$$R_{lp} = \sum_{c} Y_{cp} TP_{lc} \tag{6.2}$$

where R_{lp} is the amount of product p produced in cycle l and Y_{cp} is the yield of product p from crude c. The processing units can be broken down further into the different units (CDU, reformer, cracker, blending) similar to (Pitty et al. 2008) since they involve only equations and no algorithms. For simplicity, here they are lumped into an overall crude-to-product yield Y_{cp}.

The inventory of product p at the beginning of cycle l, IP_{lp} is obtained by mass balance:

$$IP_{(l+1)p} = IP_{lp} + R_{lp} - D_{lp} \tag{6.3}$$

where D_{lp} is the amount of product p delivered to customers in cycle l. This parameter D_{lp} is also an input to the model.

Besides the crude inventories, other model outputs are profit and customer satisfaction index. Profit is obtained by deducting crude procurement cost, crude inventory cost, operation cost, and product inventory cost from revenue:

$$\begin{aligned} Profit = \sum_{l} (\sum_{p} P_p D_{lp} - \sum_{c} P_c CA_{lc} - \sum_{c} InvCost_c IC_{lc} \\ - CostOp \sum_{c} TP_{lc} - \sum_{p} InvCost_p IP_{lp}) \end{aligned} \tag{6.4}$$

where P_p is the price of product p, P_c is the price of crude c, $InvCost_c$ is the inventory cost of crude c, $CostOp$ is the refinery operation cost, and $InvCost_p$ is the inventory cost of product p.

Customer satisfaction for product p in cycle l is measured by the ratio of product delivered to customer demand:

$$CustomerSatisfaction_{lp} = \frac{D_{lp}}{AD_{lp}} \tag{6.5}$$

where AD_{lp} is the actual customer demand in cycle l for product p.

Since decision making is not modelled, Model E provides limited decision support capability. Its application is limited to estimating crude inventories, profit and customer satisfaction given a set of input parameters, including crude arriving CA_{lc}, crude processed TP_{lc} (throughput rate at which crude is sent to the CDU for processing), and product delivered D_{lp}. The next model, on the other hand, employs algorithms to explicitly capture decision making holistically within the supply chain model.

6.3.2 Model M: A numerical model in MATLAB

The second model is a numerical model implemented in MATLAB/Simulink (MathWorks 1996) (hence Model M). Moving left from Model E on the model element axis (Figure 2.1), this model has a focus on individuals – decision making entities and executing entities, which in turn necessitates the use of algorithms for describing behaviour. Four types of entities are incorporated in the model: external supply chain entities (e.g. suppliers), refinery functional departments (e.g. procurement), refinery units (e.g. crude distillation), and refinery economics. Some of these entities, such as the refinery units, operate continuously while others embody discrete events, such as arrival of a VLCC. Both are considered using a unified discrete-time representation. As such it covers the full complexity of the supply chain as shown in Figure 4.3.

System variables – material, information, finance – are all modelled as flows interconnected by various (mathematical, logical, algorithmical) operation blocks. For better organisation and presentation, the flows and operation blocks related to a particular entity are grouped together under a masked block (MathWorks 2008). Hence, there is a "Supplier" masked block, a "Procurement" masked block, a "CDU" masked block, etc. In addition to these, decision making policies (procurement, unloading, production) are coded in MATLAB m-files (Pitty et al. 2008). Thus, Model M can be placed somewhere near the border between Quadrants II and III (Figure 2.1).

Since Model M allows the modelling of crude procurement, transportation and unloading, the crude inventory balance equivalent to Equation 6.1 here is:

$$IC_c(t+1) = IC_c(t) + PR_c(t) - TP_{lc} \tag{6.6}$$

where $PR_c(t)$ is the rate at which crude c is pumped into storage from the pipeline at time t.

Owing to the greater modelling detail, a finer simulation time step can be used: time t instead of cycle l. One time tick could be one-hundredth of a day while a cycle could have a duration of seven days. In contrast to one overall crude arrival term CA_{lc} in Equation 6.1, Model M has different variables for crude ordered, crude transported, and crude unloaded and pumped into storage tanks. These are not user input, but calculated by the model as an outcome of the various policies. In effect therefore, it is the policies that are input to the model.

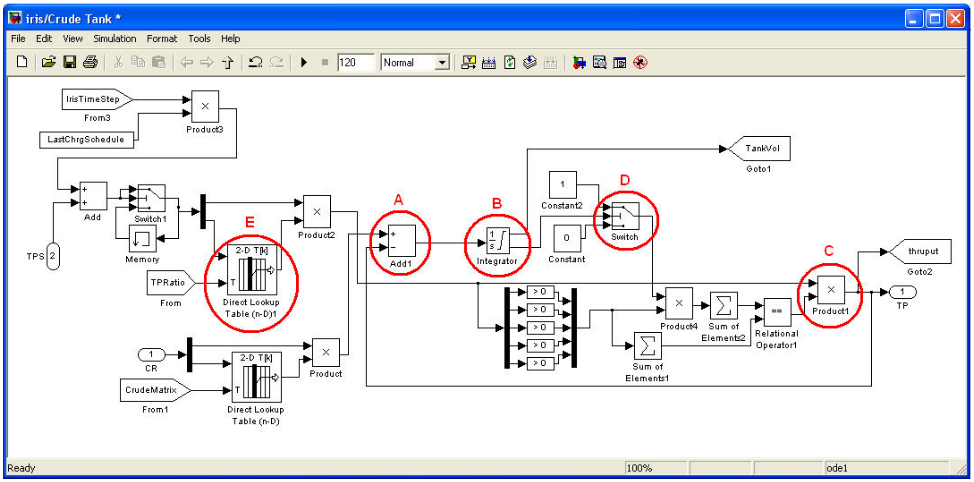


Figure 6.2 – *Crude tank masked block in Model M*

Equation 6.6 is *implemented* under the "Crude Tank" masked block (See Figure 6.2). The main operation blocks used are an addition block (A in Figure 6.2), an integrator block (B), a product (i.e. multiplication) block (C), a switch block (D), and a lookup table block (E). The first three blocks embody equations. The addition block is adding the crude input flow $PR_c(t)$ and subtracting the crude output flow TP_{cl} to get the net crude movement. The integrator block calculates $IC_c(t+1)$ from $IC_c(t)$ and the net movement. The product block multiplies the crude ratio (based on the recipe) to the crude inventory to get the correct crude mix in the throughput to CDU.

The other two blocks represent algorithms. The switch block is used to enforce a logical constraint and set a particular crude's output to zero when its inventory reaches zero. The lookup table block is used to get the crude recipe for a production mode. Connection tags are used to convey the information of the flows to other masked blocks. The assigned production mode comes from the production policy. The crude input flow $PR_c(t)$ comes from the "Pipeline" masked block. The crude output flow TP_{lc} is sent to the "CDU" masked block.

Model M is more detailed than Model E and can provide more extensive decision support. It can be used to evaluate strategic, tactical, and operational decisions, analyse different policies, and study disruption management as demonstrated in (Koo, Adhitya, Srinivasan & Karimi 2009).

6.3.3 Model R: An agent-based model in Repast

The third model considered here, using the agent-based paradigm, has been presented in Section 4.3. It was created with the same level of detail as in Model M. This agent-based model is implemented in Java using the Repast agent simulation toolkit (North et al. 2006), hence for this study it is given the label Model R.

6.3.4 Mapping the models on the model-space

Three different models of the same supply chain have been presented here. The first model, Model E, only uses equations and the second model, Model M, adds algorithms for the various decisions that have to be made and that cannot be captured by equations only. The third model, Model R, implements the same behaviour as Model M, but using a different paradigm in which equations are not explicit and in which decision algorithms as well as mass balance equations are distributed among the actors.

While they have different designs, there are also many similarities between the models (see Figure 6.3). Model E and Model M are both implemented in mathematical software tools and equations are the predominant system description elements. As such both could be labelled *equation-based models*. However, these two models are very different with respect to their model elements: Model E includes system observables only while Model M takes individuals as the constituents of the model. This brings Model M closer to Model R than to Model E. The fact that Model M and E share the same category of software tools can hide the differences between them, but Figure 6.3 reveals that they do not have much else in common.

Model E is included in this chapter to demonstrate that the same system can broadly be described using only equations, but that some of the behaviour cannot be captured without algorithms. Profit and customer satisfaction are output variables of the modelled actions of individuals in Models M and R, while in Model E they are observed in the real system and have to be provided as input by the user. Because the scope of Model E is different, only models M and R are included in the rest of the benchmarking study. Still, it should be stressed that Model E is an important category to include here to illustrate a class of models that is often used when comparing equation-based and agent-based models and to highlight that conclusions drawn from such a comparison are not always valid.

6.4 Benchmarking case study: Oil refinery supply chain

In this section the two models of the oil refinery supply chain are benchmarked following the steps outlined in Section 6.2.

6.4.1 Definition of the objectives for the study

The objective is discovering the added value of different modelling paradigms (not models) for supply chain management.

6.4.2 Identification of what is to be benchmarked

The objects of the study are two modelling paradigms that produced two different models of an oil refinery supply chain, presented in Section 6.3: Model M (Section 6.3.2) and Model R (Section 6.3.3). As discussed in Section 6.3.4, they differ along different axes. Model E (Section 6.3.1) is not included in this benchmarking study because conclusions drawn from it would not be fair in comparison with the other two models given the differences in the granularity.

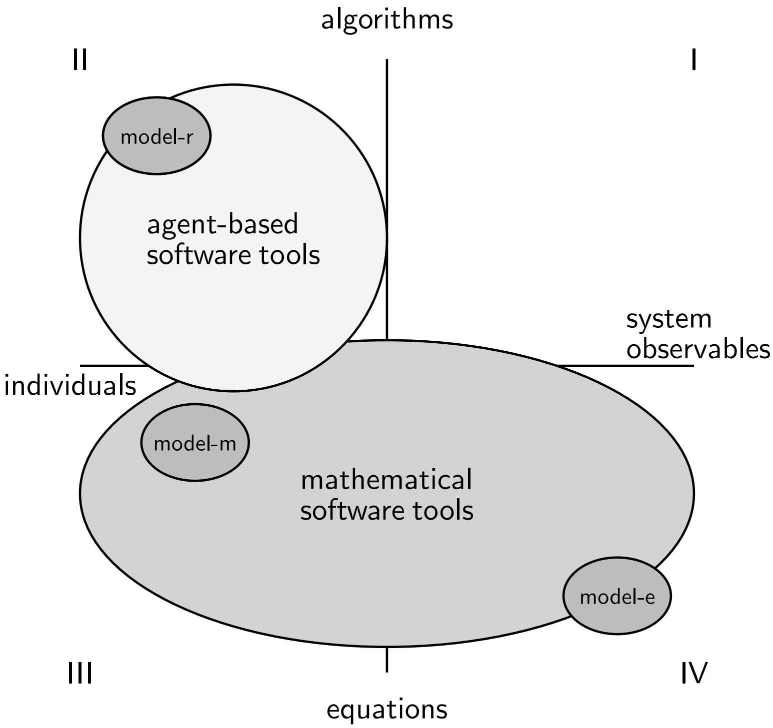


Figure 6.3 – *The three models presented in this chapter plotted on the modelling space from Figure 2.1*

6.4.3 Evaluation if objects of study are comparable

As said in Section 6.2.3, there are three different levels at which it can be demonstrated that the objects of study are comparable. The following sections address the model verification, result validation and the comparison of policy recommendations following from the simulation models.

6.4.3.1 Model verification

Both models are computational models and – even though the behaviour is formulated with different system description elements – the actual calculations are exactly the same. As an example of this, consider the calculations for the amount of crude in stock at a given moment in time as formulated by Equation 6.6. In Model M this is implemented using masked blocks, and in Model R these calculations are done by storage department agent using addition functions in Java (See Algorithms 1 and 2 in Section 4.3.5) of which the value is determined by algorithms in other agents. The same comparison can be made for other equations, and each time the same calculations are made but the formalisation of the calculation is done differently.

There are no numerical-method-dependent errors to be be expected: In MATLAB the equations are solved in a numerical way with matrix multiplications, which should always produce the same results as multiplications in the Repast model. Any differences in the numerical results should therefore be explained by inaccuracies in the model definition (perhaps imposed by a natural way of working in a certain paradigm).

6.4.3.2 Result validation

Because calculations are the same, the outcomes should be the same as well when the same input is used (this is addressed in the next step). Random numbers, however, can prevent the outcomes from being exactly the same. Stochastics are used for transport delay, demand, forecast error, etc. To prevent stochastics from influencing the outcomes, all variances can be set to 0. In this section, the following three cases are tested to evaluate result validation:

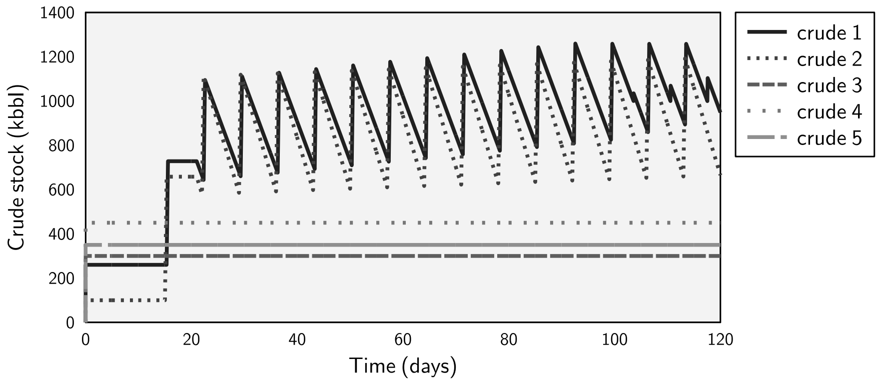
1. No stochastics, but fixed demands for each time step;
2. No stochastics as above, but halfway during the simulation the mean demands are doubled to test extreme values; and
3. With stochastics for demand and forecast error.

For each of these three cases, the crude inventory profiles produced by both models are compared. If the two models indeed produce the same results for the base case, more experiments can be done.

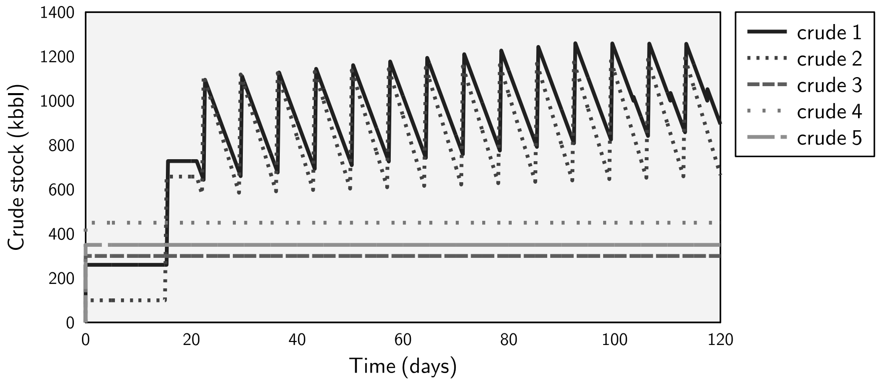
Figure 6.4 shows the results of the first experiment, without any stochastics. With demand variance set to 0, during each demand cycle the same order is placed by the consumer. This requires the same crudes to be procured and the operations department plans the same production mode and throughput. Both plots show the same behaviour and the same sawtooth profiles. Even detailed dynamics are the same. For instance on day 103, the crude parcel cannot be unloaded because of lack of ullage (i.e. there is not enough unfilled space in the storage tanks to start the unloading from the vessel). Furthermore, towards the end of the simulation it can be seen in both cases that there is excess crude. This is caused by a cumulative error resulting from the way the jetty behaviour is implemented in both models: the jetty has a fixed pumping rate and the smallest time unit is one tick which causes a + 1% difference between the amount procured and the amount transferred to storage in each cycle (see (Pitty et al. 2008) for more details). Both models show the same behaviour where pumping is paused until enough space in the crude tank becomes available. When comparing the numerical output of the stock levels at the end of the 120 days, both models produce the same number. In this scenario the models can be considered comparable.

Next, in the second experiment, a step change in the demand is introduced halfway during the simulation: the mean demand for all products is doubled. As could already be observed in the previous experiment, the crude stock levels are near the maximum capacity of the storage tanks, so further increasing the demand means testing extreme values for the simulation. Experimenting with such extreme values can be helpful to elucidate specific behaviour.

The results of this experiment are shown in Figure 6.5. The same sawtooth patterns can be observed during the start of the simulation, being exactly the same as in the previous experiment. In the second part, with extreme values, the behaviour of the models when dealing with under-capacity can be seen. Again, both models behave the same way:



(a) *Model M*



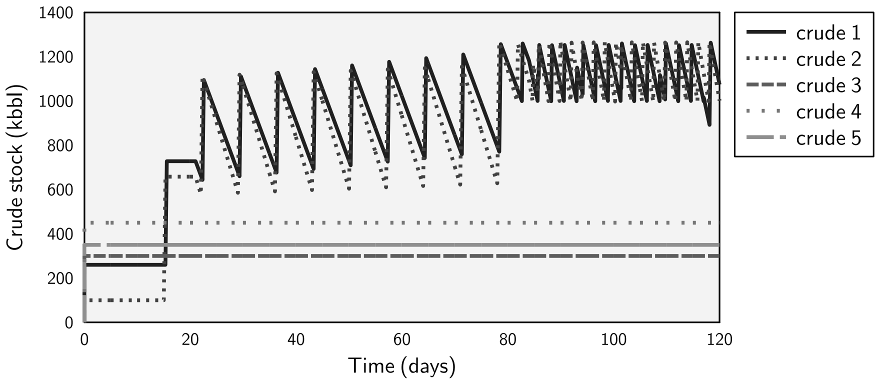
(b) *Model R*

Figure 6.4 – *Results from Case 1: no stochastics. Time (in simulation ticks) is on the horizontal axis and the inventory level (in kbbl) is on the vertical axis*

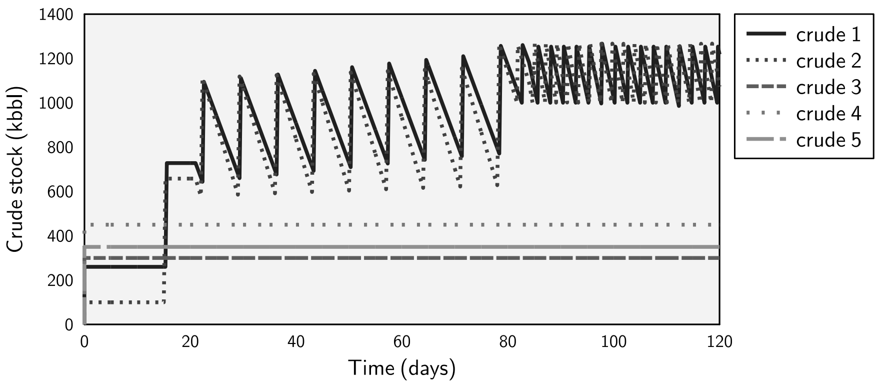
The two crudes are pumped alternately which creates a sawtooth pattern without any "flat" periods. The moment at which the increased demand becomes actual the effects of a higher throughput are visible. Looking at the graphs in closer detail, one can see that the crude stocks show a slightly steeper decline on day 83. This again confirms that the model results can be validated against each other.

However, in this experiment small differences can be observed around day 118, so it has to be concluded that the model behaviour is not exactly the same. Looking into the model definition again, a minor misalignment in the implementation of the jetty behaviour can be seen with regards to the arrival of a new ship while the previous VLCC has not been fully unloaded. This study therefore also helps the internal verification of the model by checking if the model is doing precisely what the modeller set out to build or that some (unknown) elements of the platform play a role, for example.

Finally, a new experiment is done in which demand variance is set to 25% and the forecast error to 5%. This means that during each demand cycle, different amounts of the products are ordered and that the refinery will run in various modes of operation,



(a) *Model M*

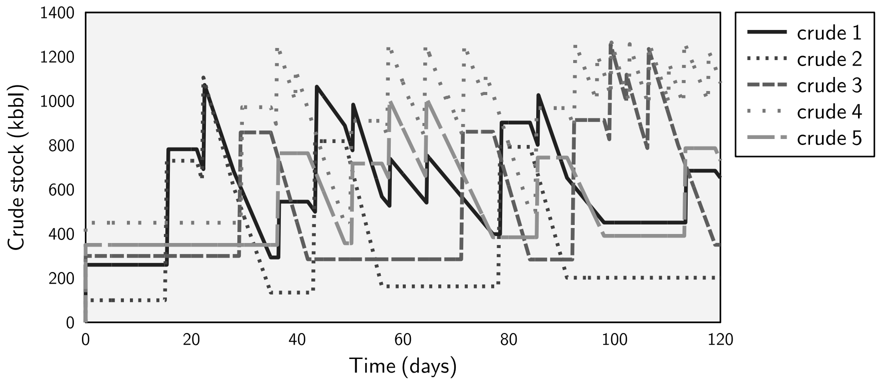


(b) *Model R*

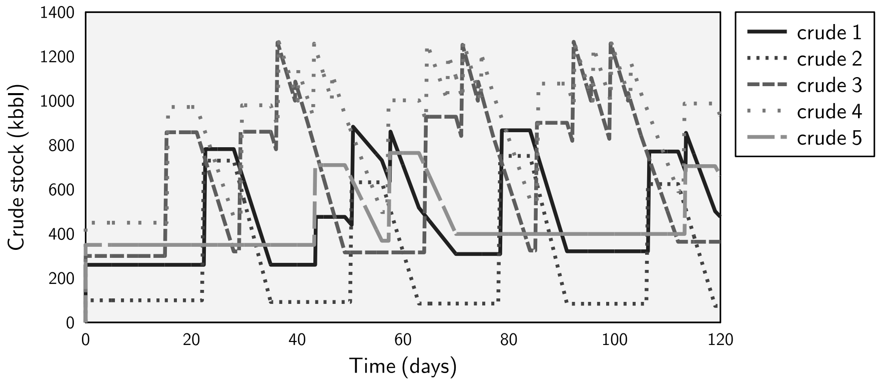
Figure 6.5 – *Results from Case 2: no stochastics with a step in demand. Time (in simulation ticks) is on the horizontal axis and the inventory level (in kbbl) is on the vertical axis*

requiring different crudes as input for the recipe selected. Also, the amount forecasted by the sales department (and used by the procurement department to determine the amount and which crudes to buy) can differ from the actual demands, possibly causing an imbalance between the crude bought and crude used. This scenario is more realistic, because fluctuations in demand and errors in forecasting demand are part of every supply chain. A demand variance of 25% is not a realistic value though as it is higher than one would see in practice, but has been used here for modelling purposes and to amplify the effect, allowing the system to be tested in the extremes.

Figure 6.6 shows the results of this third experiment. At first glance, the results from the two models appear to be broadly comparable; similar saw tooth patterns can be observed between arrival of crudes when pumping happens at high speed, and the much slower release of the crudes into the distillation unit. Also the output graphs seem to fall within the same boundaries and in both cases none of the crude tanks falls empty during the simulation period. However, the random numbers drawn in the two models are different, hence a more precise match cannot be expected since different crudes are se-



(a) *Model M*



(b) *Model R*

Figure 6.6 – *Results from Case 3: with stochastics. Time (in simulation ticks) is on the horizontal axis and the inventory level (in kbbl) is on the vertical axis*

lected and different amounts are bought, resulting from different demands and demands forecasts.

6.4.3.3 Decision recommendations

Even though after introducing stochastics[1] into the simulation the numerical results are not the same any more, these models and these settings can be used to find out if the two models give the same recommendation when used for decision support.

The model output from both models (Section 6.4.3.2) shows that the procurement policy is quite successful in that, under normal conditions without disturbances in delivery and production, crude never runs out (See experiment 3). However, the maximum capacity of the tank is reached a number of times, possibly resulting in increased demurrage costs and adding to higher storage costs. From both models it can be concluded

[1]Note that both models use different random number generators and draw numbers in a different order, so using the same *random seed* would not result in the same outcomes. It would be possible to store random numbers from one generator in lists for each variable and to let both models draw numbers from these.

that a more *efficient* procurement policy could be found in which the volume bought is reduced and the average stock levels are lower, while still ensuring adequate crude is in stock for production.

A new procurement policy has been created in which the current stock levels are taken into account. In the standard procurement policy (described as Procurement Policy 1 in Pitty et al. (2008)), the decision of how much crude to buy is solely based on the amount that is needed to meet the forecasted demand. However, if any excess crude is still left in the storage tank after production (for example because the forecast by the sales department overestimated the actual demand), then this crude is not taken into account. The new procurement policy (referred as Procurement Policy 2 in Pitty et al. (2008)) looks at the current stock levels in the tanks and possibly lowers the amount ordered from the supplier.

This policy has been implemented for both models. Even though, with full stochastics, the two models do not provide the exact same output, both models do recommend to use the new procurement policy, namely that the new procurement policy offers a reduction in the storage costs of raw materials. The two models each show that the average crude stocks are lower and yet the refinery never runs out of raw materials.

While the predicted impact on the profit of the refinery is not the same, both models do show that using this policy the profit of the company increases. This means that both models, when used for decision support, give the same recommendation for this scenario. That makes the models comparable.

In conclusion, despite the fact that the numerical results are not the same when stochastics have been introduced, it is evident that the models are similar enough to continue the benchmarking of the modelling paradigms and to learn generalisable lessons.

6.4.4 Determination and specification of performance measures

The four performance measures described in Section 6.2.4 are used here:

1. Ease of expressing the problem;
2. Ease of extending the models;
3. Ease of re-use; and
4. Ease of explaining.

In Section 6.4.6, the conclusions of the benchmarking study for each of these performance measures is given.

6.4.5 Description of scenarios and simulation

Performance is evaluated on a set of scenarios (well-structured experiments) that can be simulated to further compare the two models. The base case used above considers the operation of the integrated refinery supply chain over a period of 120 days. The following scenarios can be thought of as expansions to the base case:

1. New procurement policy (as already executed in section 6.4.3.3);
2. New production scheduling policy;

3. Utilisation of crude storage tanks under unexpected conditions;
4. Extra capacity CDU to deal with predicted growth in demand;
5. Disruption in ship arrival and emergency procurement; and
6. Coping with unexpected orders.

The specification of the data and parameters for each of these scenarios is reported in Pitty et al. (2008). These six scenarios are illustrative for the value of supply chain models in decision support. The first scenario was used in Section 6.4.3.3 to find a more efficient policy for procurement of crudes, contributing to savings in demurage and storage costs.

In the second scenario the decision support tool is used to try a new production scheduling policy. In the base case production throughput is based on the product with the highest crude demand, making sure that enough of this product is produced. However, this automatically means that for all other products there is an overproduction. A new production planning policy takes the amount of products already in storage into account.

Next, in the third scenario, unexpected maintenance is needed for one of the crude storage tanks, which reduces the storage capacity. This could possibly result in higher demurage costs because there will be less space for one of the crudes. New experiments can be done in which a crude storage tank of a crude that is not used as much is converted from one type to another, so that the economically most viable crude mix can still be bought.

The fourth scenario tests the impact of a predicted growth in demand on the supply chain. To be able to better cope with this, the value of investing in extra CDU capacity is calculated so that a well-informed investment decision can be made.

The fifth scenario deals with the disruption in the arrival of a VLCC. This can have severe impact on the planned production and the delivery of end product to the consumer. An emergency procurement strategy is implemented that, in case the ship does not arrive in time, can order additional crude with a short lead time if this crude is not available in the safety stock. Experiments can demonstrate the contribution of this new policy to refinery profits.

Finally, in the sixth scenario, an unexpectedly large order by the consumer is placed (e.g. following rejection of a batch from another refinery). This is a disruption for the refinery because the product demand will be very different from the predicted amounts that the procurement and operations planning were based on. The emergency procurement strategy from the fifth scenario can be tested again in this situation to see if the refinery can cope with the high demand in the short term.

6.4.6 Benchmarking conclusions

Our results from the benchmarking study are reported next and insights in the relative advantages of the two paradigms are presented. The four performance indicators from Section 6.2.4 are revisited in the following sections.

6.4.6.1 Ease of expressing the problem

Any supply chain contains two distinct types of elements:

- Production processes with complex physical and chemical phenomena (technological system)
- Decision making or business processes involving inter-entity coordination and collaboration (social system)

The behaviour of the former is best described through equations and the latter through an algorithm. The equation-based model caters well to technological aspects. The agent-based model has lesser expressive breadth for these, but offers instead a rich vocabulary for describing business processes behaviour. For example, including hold-ups of crudes in the pipes and the calculation of how much crude is transferred by the jetty were easily addressed in Model M model but were more complicated in Model R. On the other hand, the role of the third party logistics providers and the negotiations between various shippers are easily represented in the agent-based model whereas this provides a significant challenge in Model M. Both models can, however, fully express the same problem.

6.4.6.2 Ease of extending the models

The ease of extending the model is closely linked with the ease of expressing the problem. In general, parts of the model that are easily expressed in one model, are easier to change or extend too. However, this is not always the case because it depends on *how* it was implemented. In Model M it was relatively easy to implement the behaviour of the jetty (e.g. determining the amount pumped into storage each tick) but changing it from a fixed pumping rate to a variable pumping rate is more difficult because it requires changes to the structure of the model.

Table 6.1 – *Steps required for the extension of models for the six scenarios described in Section 6.4.5*

Scenario	Changes for Model M	Changes for Model R
1	• Modify m-file of procurement policy: equation to calculate excess crude and equation to calculate amount of crude required considering excess crude and safety stock • Connect additional input (crude inventory) required to the "procurement_policy" function block under "Control Panel"	• New algorithm to determine amount of crude needed for current cycle and compare that with current inventory • Add extra term to set of rules where amount of crude to be bought is decided
2	• Modify m-file of scheduling policy: equation to calculate amount of requisite demand considering product inventory and safety stock • Connect additional input (product inventory) required to the "scheduling_policy" function block under "Control Panel"	• New algorithm to calculate the amount of excess products using a safety stock • The procurement algorithm should be modified to subtract this amount from the forecast demands before deciding on procurement

Table continued on next page...

Table 6.1 continued from previous page...

Scenario	Changes for Model M	Changes for Model R
3	• Change "Crude storage capacity limit" for the two storage tanks in the "Control Panel" dialog box	• The capacity of two storage tanks has to be adjusted in the knowledge base • No change to any algorithms (or any source code) is required
4	• Add "demand_switch" constant block, "mean_demand" constant block, step block, and switch block under "Sales" • Connect the step increase and the random number generator for the forecast demand to a multiplication (product) block under "Sales". • Change "Maximum throughput" and "Crude storage capacity limit" in the "Control Panel" dialog box	• Modify algorithm of sales department to add step up in demand • A new calculation (Spreadsheet) has to be made to determine the yields of the whole refinery given the extra CDU capacity • These new values have to be entered in the instance definition in the knowledge base
5	• Add a combination of blocks to add to the transportation time simulated by the variable integer delay block under "Suppliers", to simulate the supply disruption • Add supply disruption parameters for user input in the "Suppliers" dialog box • Write a new m-file for supply disruption emergency procurement policy • Connect the required inputs to the created "emergency_procurement_policy" function block under "Control Panel" • Add the output from the function block to the order quantity under "Procurement" • Add emergency supplier and its lead time in the "Suppliers" dialog box	• An extra supplier needs to be created in the knowledge base, can use same source code as other suppliers • The shipper agent routine that checks when a ship has arrived needs extra delay factor • Shipper agent needs update of expected travel times for the new supplier • Shipper agent needs to inform refinery about the delay to trigger emergency procurement algorithm • New algorithm to determine the amount of emergency crude; Set of suppliers restricted to emergency supplier

Table continued on next page...

Table 6.1 continued from previous page...

Scenario	Changes for Model M	Changes for Model R
6	• Add a combination of blocks for customer rejection to subtract from actual delivery and send low quality products back to crude tank under "Customer" • Add a combination of blocks to process low quality products into on-spec product under "CDU" • Add on-spec product to product tank balance under "Product Inventory" • Add quality disruption parameters for user input in the "Customer" dialog box • Write a new m-file for quality disruption emergency procurement policy • Connect the required inputs to the created "emergency_procurement_policy_2" function block under "Control Panel" • Add the output from the function block to the order quantity under "Procurement" • Add emergency supplier and its lead time in the "Suppliers" dialog box	• Include rejection of delivered product in customer agent at given time • Update algorithm to determine actual demand (Sales department) to add product that was rejected • Add call to emergency procurement (See Case 5) when a large order is rejected

For each of the six scenarios from Section 6.4.5, Table 6.1 lists what changes are needed to be able to perform the experiment in Model M and what needs to be done in Model R to do this, too. These changes are specific for Model M and Model R in that the steps are based on the implementation in MATLAB/Simulink and Repast with the framework for socio-technical systems. Table 6.2 gives an estimate of the efforts (measured in time of one experienced modeller) required to implement the changes from Table 6.1 in both Model M and Model R (while taking care that the experience in number of years of both modellers is comparable).

Considering the estimated effort required to make these changes in either Model M or Model R it can be concluded that most of these changes are easier in Model M or at least not more difficult. However, these cases were chosen with Model M in mind (Pitty et al. 2008) and they do not change the structure of the model significantly. Changing a quantity is easy, but changing the representation is much more difficult.

A clear example of a difference in effort can be found in Scenario 4, where it is much more work to implement a change in the CDU capacity in Model R than it is in Model M. Equations offer an easier representation of the refinery process itself. Even though the process is modelled in both paradigms, it is easier to do this in Model M. On the other hand, for Scenario 5 it requires more effort to implement a disruption of VLCC arrival in Model M, something that is easy to do in Model R. The shipping agent already has an algorithm that checks when a ship is due to arrive as this is formalised in transport contracts so it takes little effort to make a ship arrive later than agreed upon (and, likewise, requires little effort to include penalties for late delivery, should this be required).

Model R, being a bottom-up agent-based model, has a flexible structure. The con-

Table 6.2 – *Estimated efforts to implement six scenarios (See Table 6.1) in Model M and Model R*

Scenario	Model M	Model R
1	$<$ 4 hours	$<$ 4 hours
2	$<$ 4 hours	$<$ 4 hours
3	$<$ 1 hour	$<$ 1 hour
4	demand switch: $<$ 1 hours change throughput $<$ 1 hours	demand switch: $<$ 1 hours change throughput $<$ 4 hours
5	supply disruption: $<$ 8 hours emergency procurement: $<$ 4 h	supply disruption: $<$ 1 hours emergency procurement: $<$ 4 h
6	customer rejection: $<$ 8 hours emergency procurement: $<$ 4 h	customer rejection: $<$ 8 hours emergency procurement: $<$ 4 h

nections between the constituents (actors) is not hard-coded, unlike in Model M, hence new connections between agents can be created on the fly. One could say that more information is available in this model, because exchange or interaction between agents can be easily defined. The physical elements of the refinery such as the CDU operation, however, are modelled explicitly in Model M and not in Model R. In summary, the two models used different *representation mechanisms* and if something is only indirectly captured in the model, it requires more effort to be changed.

6.4.6.3 Ease of re-use

The agent-based paradigm provides a hierarchical framework to describe the model constituents. In the framework used for Model R, a key part of the model – the ontology – was derived from earlier modelling efforts in other domains. The generic ontology makes re-use easier and also allows connections to other models, for example one of an industrial cluster incorporating other chemical industries where other agents could become consumers of the refinery. The numerical model does not *enforce* any such structure, hence reusability is in general difficult, especially between different modellers. Still, both models presented here have been used to support the development of new models in a similar, but still different, domain.

Two models of a lube additive supply chain have been created: one in MATLAB/Simulink and one in Java/Repast. The main differences with the oil refinery supply chain presented in this chapter are that there are multiple sites with a central sales department which assigns orders to the sites and that the process is order-based and not continuous. Furthermore, this supply chain deals with speciality products so no product inventory is kept. In the development of the MATLAB model of the lube additive supply chain (Wong, Adhitya & Srinivasan 2008) many of the thoughts that have gone into building Model M have been re-used, but none of the actual equations or blocks could be directly transferred. The second model of this same lube additive supply chain, built in Repast (Behdani, Lukszo, Adhitya & Srinivasan 2009), is more closely based on Model R. Not only does it re-use the conceptualisation, but it also shares the same ontology and through this it was possible to directly re-use source code for some of the algorithms (e.g. for procurement and the shipping agent) while adding specific scheduling algorithms that are unique to this case study.

6.4.6.4 Ease of explaining

Model M and Model R can both give the same recommendations (See Section 6.4.3.3), but explaining how the model came to this conclusion is important too. In general one could say that equations are generally best understood by people with a background in mathematics or traditional (process systems) modelling. The agent paradigm provides a more "natural" representation that could appeal to decision makers without a mathematical modelling background. That this natural representation is only possible with agent-based models is, as has been demonstrated above, a misconception but will still be prevalent to many. Still, the explicit hierarchical structure in an agent-based model enables a natural representation for behaviours, both in terms of organisation and visualisation; this is harder with a set of equations.

The mass balance of Equation 6.6 could be used to explain how both Model M and Model R work. Neither of the two models explicitly includes this equation, but both indirectly make this calculation (recall Algorithms 1 and 2 on page 85). Furthermore, since Model M has a focus on *individuals* as model elements, the agent paradigm can also be used to *explain* the interactions and relationships between the actors, even if the model has not been implemented in an agent-based toolkit.

Models M and R are therefore similar in terms of ease of explaining, but Model R still has an edge for natural representation of the decision making processes and interactions between the entities in the supply chain while Model M has an edge for explaining the technical process.

6.5 Conclusions

In this chapter it was demonstrated that different modelling paradigms and tools can be used to successfully create a model of an oil refinery supply chain. In order to come with fair conclusions based on the comparison between them, it is important to stress that in the space of models, *equations* and *agents* are concepts of a different order. The former refers to the system description elements in the model while the latter emphasises the model elements. Thus conceptually, the "equation-based" and "agent-based" paradigms are not mutually exclusive. They are merely labels that are often convenient, but sometimes distracting. The modelling space presented in Section 2.3.3 can be used to visualise to what extent models are similar and how they are different. The three models presented in Section 6.3 all have different characteristics so that they are displayed in a different quadrant of the modelling space in Figure 6.3.

Two of these models, one created using MATLAB/Simulink and one using the Repast agent platform and a framework for socio-technical systems, have been used in a benchmarking study, following the general benchmarking steps from Section 6.2. By performing detailed experiments with the two models, it is demonstrated that the models are equivalent when compared using model definition, numerical results and recommended decisions. However, the modelling process itself is different for the two cases and results in different model structures and different representation mechanisms.

By analysing the efforts required to expand the models, allowing new scenarios to be tested, the strengths of the two paradigms were identified in the context of supply chain modelling. Production processes and the technological aspects are well catered by equations, while the decision making aspects can only be captured in algorithms. The complete system can, however, fully be expressed in both modelling paradigms. When it comes to extending or adjusting the models, one can say that if something is only indirectly captured in the model, it requires more effort to be changed. The physical elements of the refinery such as the CDU operation are modelled explicitly in Model M and not in Model R. On the other hand, Model R explicitly has a flexible structure, allowing new agents and connections between agents to be added in extensions to the model. In general, the efforts required to make changes in the model for a number of scenarios, ranging from operational to tactical and strategic levels, are similar.

Two models of a lube additive supply chain have been created, building upon the two models used in the benchmarking study. Many of the thoughts that have gone into building Model M have been re-used, but none of the actual equations or blocks have been directly transferred. For the agent-based model not only the conceptualisation could be re-used, but the models also shares the same ontology and through this it was possible to directly re-use source code for some of the algorithms. Finally, when it comes to explaining the model and the model results, Model R offers a natural representation of the decision making processes and interactions between the entities in the supply chain while Model M has an edge for explaining the technical process.

Chapter 7

Decision support with agent-based models

This chapter is partly based on van Dam, Lukszo & Srinivasan (2009).

7.1 Introduction

In Chapter 4 *simulation models* of various systems were presented. The next step, namely the development of *decision problem models* for a problem owner, is taken in this chapter. Even though most systems are multi-actor systems, a problem is analysed from the perspective of one problem owner. It should be stressed that the problem owner is part of a multi-actor, multi-criteria and multi-level system and its decisions will influence other elements of this system, but decision support is considered here following an assignment from one stakeholder in the system only. This problem owner can, for example, be a governmental body or a private firm. In some cases different stakeholders could benefit from the same model of the system (e.g. a company's competitor would probably be very interested to see the world from a different perspective and learn how to play the market better, and the government might gain insight in what companies base their decisions on so it can tailor its policies to the behaviour of private players), but it is assumed here that only one actor has access to the model.

After performing the modelling tasks (Section 3.6) the model is ready for simulation to help the decision maker (See Section 1.4). The following tasks, which can be considered as a standard approach to optimisation problems (see for example Edgar, Himmelblau & Lasdon (2006)), have to be performed before a model can be deployed as a decision support tool:

S-1. Formulate the decision problem by specifying the criteria, system model, constraints and the degrees of freedom.

S-2. Choose the search strategy and formulate experiments.

S-3. Decide on the parameters for the search procedure and perform the experiments.

S-4. Analyse the results of the experiments and formulate recommendations for solving the problem that was formulated in simulation step S-1.

Next, these steps are discussed in more detail. The first step — decision problem formulating — is discussed in Section 7.1.1. In Section 7.1.2 different search strategies are addressed and Section 7.1.3 deals with carrying out experiments. Finally, in Section 7.1.4, the analysis of results is discussed. After this introduction, Sections 7.2 and 7.3 demonstrate how two models that were presented in Chapter 4 are deployed for decision support.

7.1.1 Formulate the decision problem

Without a clear formulation of the decision problem it cannot be solved systematically and the outcome cannot be evaluated. The following choices have to be made, independent of how the problem is solved later:

Criteria One or more criteria are selected by the problem owner. The criteria are used to select the best ("optimal") decision from all options, given the model, constraints and degrees of freedom. Due to the complexity of the problem and the system, the term *optimal* solution is not always adequate — often it is hard to find a solution at all and it cannot be said that no better solution is possible. The optimal solution is considered as the best one found (according to the criteria, without any claims that no better solution may exist) or any improvement over the current situation.

Degrees of freedom The degrees of freedom define which parameters may be varied in the system, bounded by the constraints. A solution for a decision problem comes in the form of a choice for the value for these parameters, resulting in a certain value of the criteria. The degrees of freedom at a certain point in time t are defined by a vector $\bar{x}(t)$.

Constraints Constraints limit the search space for feasible solutions by giving boundaries to degrees of freedom and criteria. Some constraints may be hard constraints in the system, for example a minimum throughput of an oil refinery to keep the process running, which is a design property of the production units in the real system. Additional constraints can be defined by the problem owner, for example that a shut-down of the refinery has to be avoided at all cost.

Model In computational model-based decision support the model of the system determines the value for the criteria through simulation. Steps M-1 to M-7 (See Section 3.6) were formulated to define the model. The model can capture several possible disturbances defined in vector $\bar{d}(t)$, with $\bar{d} = 0$ during normal situations. The value of the criteria is then determined by $model(\bar{x}, \bar{d})$. The system model is an input to the decision problem.

With criteria, degrees of freedom, constraints and the model defined, an attempt can be made to solve the problem.

7.1.2 Select the decision problem solving method

A well-formulated problem is merely the first step, next it has to be solved. In this thesis a simulation-based approach is taken, but it is acknowledged that alternative decision problem solving approaches exist and can be useful. One can think of a team of experts evaluating different solutions and coming up with recommendations without having to create and run a simulation model, or the traditional mathematical optimisation (e.g. non-linear programming, mixed-integer non-linear programming). In this thesis an approach based on computational simulation models is taken (Section 1.1). It is also demonstrated how mathematical optimisation can help solve problems using a simulation model.

A search method is needed to choose the experiments to be performed and to find a solution for the decision problem. Below two such methods, genetic algorithms and the Nelder-Mead zero-order optimisation methods, are briefly introduced to give an impression of the wide range of possible search methods. While the first one is merely mentioned as an example, the second sesarch method is used to solve a specific problem later in this chapter.

A *genetic algorithm* is a search method based on the *genetic operators* of selection, cross-over and mutation (Holland 1975) inspired by Darwin's (1859) theory of evolution. A "population" of possible solutions (i.e. values for the degrees of freedom, considered as a "genome") is generated and their "fitness" (i.e. value of the criteria for this set of parameters, for example through simulation) is calculated. The best solutions are kept for the next iteration of the search process, while the worst are replaced by new solutions formed by mutation (i.e. making small changes to a solution) or crossover (i.e. combining elements from old solutions to form new solutions) based on random numbers, after which the fitness can be calculated again. This process continues until a suitable answer is found, time runs out or no improvement can be made. The conditions for using genetic algorithms are that solutions can be encoded as a string in the form of a genome and that a function (or simulation model) exists to calculate their fitness.

An alternative approach to determining which values for the degrees of freedom should be tested in simulation, is offered by the Nelder-Mead zero-order optimisation method (Nelder & Mead 1965). The name zero-order refers to the fact that the search for the optimum is carried out without calculating any derivatives of the performance criterion but directly by measuring (without the help of a process model) or simulating the state of the system. The zero-order search methods are recommended when the process has one or more of the following properties (Wright 1995):

- Process model is difficult or expensive to obtain;
- Process exhibits discontinuities; or
- Measurement data are contaminated by significant noise.

The search with a Nelder-Mead simplex algorithm is applicable to solving decision problems as formulated in this thesis. The method is based on identification of the best, the worst, and the second worst outcomes in each iteration for the pre-defined simplex (a set of experiments). An initial simplex S is defined as a convex hull with $n+1$ vertices $\{\bar{x}_j\}_{j=1}^{n+1}$ in an n-dimensional space $\mathbb{R}^n$ (with n equal to the number of degrees of freedom in $\bar{x}$). These vertices satisfy the non-degeneracy condition, meaning that the volume of the simplex hull is non-zero. For every next iteration $j+1$, the values for $\{\bar{x}_j\}_{j+1}^{n+1}$ are determined by comparing the objective-function values followed by replacement of the worst vertex by another point. The simplex adapts itself to the local landscape and finally contracts to the (local) optimum.

Despite the algorithm's age (1965), the method is still applied in practice today (e.g. Lu, Murray-Smith & Thomson 2008, Ye & Xiong 2008, Ouria & Toufigh 2009). Main advantages are that the method is easy to implement, easy to communicate to decision makers and it is likely to give a global optimum in dealing with non-linear constraints and multi-criteria problems (Verwater-Lukszo 1996). It is acknowledged, however, that there are also significant disadvantages to the Nelder-Mead approach: it converges to an optimum slowly, is inefficient and does not guarantee finding the global optimum. Having said that, the ease of implementation and ease of explaining make it a suitable method to apply as an example of a popular optimisation strategy used in this thesis in combination with an agent-based model.

It should be stressed that the two approaches briefly addressed above are only examples of how a decision problem can be solved and that the proposed solving method is independent of the formulation of the decision problem. Depending on the problem solving method, one might have to first decide which experiments to conduct, using principles from design of experiments (Fisher 1935, Montgomery 2008). Furthermore, less structured approaches are also possible in which domain experts determine which possibilities should be tested by simulating the system under different conditions and how to proceed based on these findings. This is particularly interesting when the fitness or objective function cannot easily be defined and when a different course of action is needed based on initial findings.

7.1.3 Perform experiments

With a well-defined problem and after selecting a search strategy, experiments can be performed. First, however, one needs to decide on the stop condition for the search as well as the initial values for the degrees of freedom. *When* to terminate the search depends on various factors, such as the cost of experiments, the desired accuracy of the outcomes, time limitations, etc. This decision will be specific for a certain problem and search strategy.

After a search method has been selected, the initial experiments have to be defined by choosing *what* the values for the degrees of freedom are. The search method then deals with *how* these values are changing in the process of solving the problem. Depending on the search strategy, one or more values for the degrees of freedom in $\bar{x}$, within the boundaries defined by the constraints, need to be chosen.

Following the search strategy the experiments can be performed (either manually or fully automated), collecting the result for further analysis.

7.1.4 Analyse the results

The results of the experiments are studied and analysed next, leading to a recommendation on the problem that was formulated in step S-1. This outcome consists of recommended values for the degrees of freedom in $\bar{x}$ for one or more time steps t that give the best solution for the chosen criteria and under a certain disturbance $\bar{d}$, if any.

Additionally, more experiments can be done to further analyse the outcome of the problem solving process. The "optimal" solution is chosen based on certain assumptions in the model (needed to simplify the real system) and assumptions for the parameter values. A *sensitivity analysis* can be carried out next, to discover how well this solution holds if the assumptions turn out to be not as expected. Perhaps another solution is more robust and, even though it does not show up as the optimal choice in the search, it can turn out to give better results for the problem under consideration.

The vector of disturbances $\bar{d}$ may also contain uncertainty, for example about the duration of a transport delay or the severity of the malfunctioning of a piece of equipment. As time progresses, perhaps more information becomes available and uncertainty diminishes. It is important to experiment with different solutions and to see how well they behave under uncertainty. Simulation models are, in general, particularly well suited for exploring the *solution space* rather than just giving one single recommendation.

Possible additional steps include the evaluation of this recommendation after it has been implemented in the real system, which can also lead to further verification and validation of the simulation model used as well as the appropriateness of the selected criteria, degrees of freedom, constraints and the search method. This gives a critical view of the whole decision support trajectory and the choices that were made, providing valuable lessons for follow-up experiments.

7.2 Decision support for the location of an intermodal freight hub

The first decision support application deals with the development of a new intermodal freight hub. As said in Section 4.2, a key variable in the development of an intermodal freight transport system is the location of the freight hub. In this case study, three conceptually difficult issues have been identified (Sirikijpanichkul 2006):

- Investments in the infrastructure cause dynamic effects, for example on transport demand. This means that additional investments in the infrastructure may be needed to cope with, for example, congestion. This has to be taken into account already in the design of the new freight transport system.
- Actors can have conflicting objectives that prevent reaching a compromise. Different stakeholders may prefer a different location for the freight hub and somebody will have to give in, perhaps with some sort of compensation.
- The problem owner cannot force other actors to make certain decisions. This includes the actual construction of the hub, which is done by a third party (e.g. a consortium) so the problem owner has to create the right conditions for an optimal solution for all stakeholders.

Models can help the decision maker deal with these challenges.

The problem owner, in this case the governmental transport agency, has indicated that it needs more insight into the relationships between the stakeholders to make a decision. Three steps can be identified on the path towards the development of a new freight hub in which decision support tools can assist the problem owner:

1. Generating possible solutions for the location of the freight hub;
2. Evaluating the solutions for each stakeholder and the overall system level; and

3. Experimenting with different policies and measures to influence the decisions of the individual stakeholders.

Steps two and three form an iterative process. A location can only be chosen after experimenting with different instruments to influence the actors. For each potential location the situation should be analysed using the model and different scenarios be played out, before a judgement on the suitability of the hub location can be given.

7.2.1 Formulate the decision problem

Next, the criteria, degrees of freedom, constraints and model are defined for the intermodal freight hub case.

7.2.1.1 Criteria

The performance criterion is the profit of all n stakeholders for a chosen hub location at time $t = 3$ years. Stakeholders make a profit by selling (intermediate) products and buying raw materials or intermediates. For the transport agent the income comes from transporting goods for another party and it receives money for this service rather than for selling a product. Its expenditures consist of maintenance costs for the vehicle fleet and operational costs of fuel and drivers. The subsidy is the total of money received from tax incentives. The world market is not taken into account as a stakeholder, only the parties directly using the freight hub for their business.

The objective function is thus defined as follows:

$$\begin{aligned}\max_{\bar{x}} \quad P(\bar{x}, \bar{d}) = \sum_{i=1}^{n} (&Income^i_{\text{sales}}(\bar{x}, \bar{d}) - Cost^i_{\text{raw materials}}(\bar{x}, \bar{d}) \\ &- Cost^i_{\text{transport}}(\bar{x}, \bar{d}) - Cost^i_{\text{maintenance}}(\bar{x}, \bar{d}) \\ &- Cost^i_{\text{operation}}(\bar{x}, \bar{d}) + Subsidy^i(\bar{x}))\end{aligned} \tag{7.1}$$

7.2.1.2 Degrees of freedom

The main model parameter to vary is the location of the intermodal freight hub as it directly relates to the question of the problem owner in this decision problem. The choice for the location has a major impact on all other stakeholders. Additionally subsidies for different stakeholders can be introduced alongside a choice for the hub location. The degrees of freedom are thus defined as follows:

$$\bar{x} = (HubLocation, Subsidies) \tag{7.2}$$

$$Subsidies = (Subsidy_1, Subsidy_2, \ldots, Subsidy_n) \tag{7.3}$$

in euro per km per product, for n actors

7.2.1.3 Constraints

There are constraints on both degrees of freedom. In step one on the trajectory towards the development of a new freight hub possible solutions for the location are generated. In step two these are evaluated. It is assumed the first step has been completed and that, through an expert study and a first rough analysis, three possible locations for the freight hub have been identified (See Figure 4.2 for one such location scenario). The subsidy cannot be negative, and here it is assumed that subsidies higher than 1 euro per km per product are neither feasible nor desirable.

$$HubLocation \in \{L1, L2, L3\} \tag{7.4}$$
discrete choice between pre-defined locations
$$0 \leq Subsidy_n \leq 1 \tag{7.5}$$
in euro per km per product, for each of the $n = 5$ actors

An additional constraint is formulated by the problem owner who wants all actors to agree with a proposed location. To ensure that actors will accept a proposal, their profit cannot be negative (which would mean that they are losing money because of a chosen location for the hub). If one or more actors lose money compared to the initial situation, experiments are done to identify measures that can reduce the losses and generate adequate incentives for all actors to agree with a certain proposed location until all actors make a profit. A constraint is formulated that all profits have to be positive:

$$Profit_i \geq 0 \qquad \text{in euro, for each of the } i = 1, \ldots, n \text{ actors} \tag{7.6}$$

7.2.1.4 Model

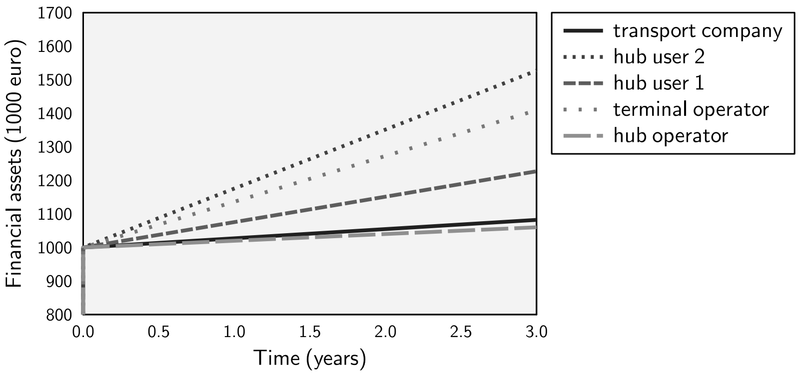
The model used here is the one described in Section 4.2. It gives the profit of the actors in the system for a given location of the hub and possible subsidies. For this case study any possible disturbances to the system are not considered, therefore it is assumed that $\bar{d} = 0$.

7.2.2 Select the decision problem solving method

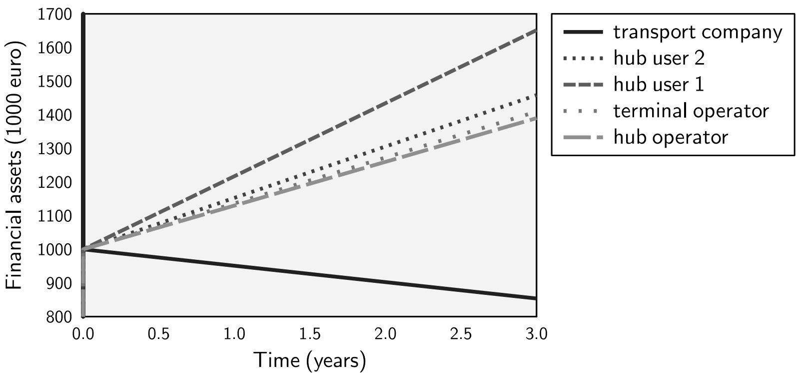
The agent-based model must be run for each identified location of the new freight hub (including expansion of the existing hub). This is the main parameter of the model in the first set of experiments, after which one location is selected by the problem owner. Next, experiments can be done with parameters that influence the decisions of the actors in the system. By experimenting with a second set of parameters such as tax rates or subsidies, and repeating the first experiment, one can check the influence of these parameters on the profit of the actors as seen in the model output graphs. The goal is to change the value of the objective function of agents so that they more or less agree with each other. This gives the problem owner a new insight into how to control the environment and how to influence actors. This insight can be used to create the right conditions for building the freight hub in the best location.

7.2.3 Perform experiments

First, for each $HubLocation \in \{L1, L2, L3\}$ the model is run once. No stop condition is needed because of the limited number of experiments required to test all values of $\bar{x}$ when the $Subsidy$ is set to 0 for each of the n actors. The initial values for the degrees of freedom are therefore $\bar{x} = (L1, 0, 0, 0, 0, 0)$, $\bar{x} = (L2, 0, 0, 0, 0, 0)$ and $\bar{x} = (L3, 0, 0, 0, 0, 0)$. By changing the location of the hub, it can be shown that some actors benefit from this decision while others loose (See Figure 7.1).



(a) *Location L1: hub located at lat:250, lon:250*



(b) *Location L2: hub located at lat:300, lon:300*

Figure 7.1 – *Illustrative results from the intermodal freight hub case study*

Furthermore, it is illustrated that simple measures to support actors that lose money can encourage them to still support a hub location. Experiments to discover an appropriate tax deduction are conducted for one of the locations. Giving a subsidy for the variable costs of the transport agent can help prevent it from losing money. This influences actors to still accept a hub location that was initially not ideal for them. The search can terminate when the constraint $Profit_i \geq 0$ is no longer violated. The initial $\bar{x} = (L2, 0, 0, 0, 0, 0)$ as used before, but the subsidy for the actor that loses money is raised in small steps.

Figure 7.2 shows different values for the operational costs (in euro/unit volume/km) in relation to the financial assets of the transport agent at $t = 3$.

7.2.4 Analyse the results

The results show that not all possible hub locations are suitable for all actors, but that small subsidies can be effective to persuade them to still accept a choice which initially was not ideal. Keeping in mind that the initial financial assets of the agent were 1,000,000 euro, it can be seen from Figure 7.2 that a loss is made at a price of 0.9 euro/unit volume/km,

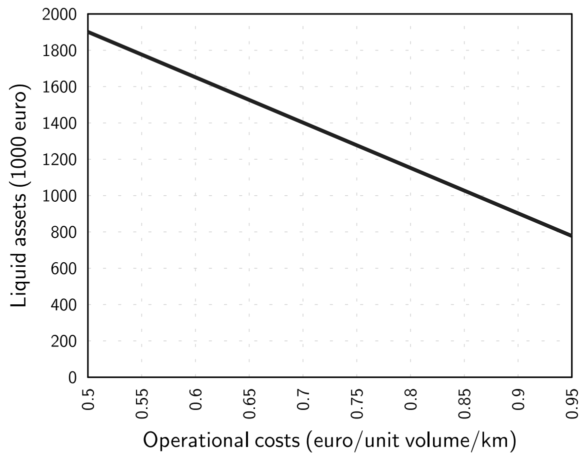


Figure 7.2 – *Financial assets of the transport company agent (after 3 time ticks) as a function of operational costs, to find out minimum required subsidy to prevent losses caused by an unpreferred hub location L2*

while a small profit is made at 0.85 euro/unit volume/km. From this it can be deduced that a tax discount on the operational costs of 0.05 euro/unit volume/km is enough to compensate for the losses that arise from one of the potential hub locations, with $\bar{x} = (L2, 0.05, 0, 0, 0, 0)$ as the final result.

7.2.5 Conclusions

To make realistic policy recommendations, a more realistic network representation and demand (based on actual data collected from the existing network) are needed. In this case the dynamic effects of a certain hub location on transport demands were not taken into account. For this purpose the agent-based model can be connected to transport models as described in Sirikijpanichkul et al. (2007). This means that only further extension of the ontology and the model specification in the knowledge base (and subsequent implementation of these additional components) is needed but that no conceptual changes have to be made to the core of the agent-based model.

In each run, the model updates itself each time step to include the dynamic effects of this choice for a hub location and the results of the traffic model are fed back into the agent-based model. This is done with a time horizon of 20 to 30 years. Over time, the values of the objective functions of all actors in the system can be calculated. This results in a set of graphs that show how much the various actors appreciate the situation and how their degree of satisfaction changes over time as a result of the changes that are caused in their environment. This deals with the difficult issue of dynamic effects and the possibly conflicting objectives of actors and can help the problem owner to better explain and justify its decisions.

Despite the simplification of the system and the limitations of the decision support tool without the underlying dynamic traffic model, the case study already illustrates how the agent-based model can be used to experiment with different locations and incentives and to gain insight into the complicated relationships between the variables in the system. Furthermore, because of its bottom-up nature, the model can easily be extended for more realistic cases.

7.3 Decision support for abnormal situation management in a refinery supply chain

Abnormal situations in supply chains encompass a range of events outside the "normal" operating modes[1], including human error, fires, delays in ship arrival, (unplanned) maintenance and equipment failure. As a consequence, planned production targets may not be accomplished, unless swift response is taken to minimise the negative effects. Which response is the most efficient one, however, is not easy to determine and requires decision support tools. This section describes how a decision support tool using an agent-based simulation model can help an oil refinery in dealing with disturbances and ensuring smooth operation at minimal costs.

To determine nominal process conditions the model performs optimisation for normal operation by choosing which crudes to buy, how much crude is needed and from which supplier to order. The mode of operation is scheduled based on predicted demands and the throughput for operation of the refinery is set based on actual demand from the consumer. When an abnormal situation is manifesting itself this normal approach is not adequate any more. A model-based decision-support tool is therefore called for.

The multi-actor, distributed, complex and dynamic nature of a supply chain can be best evaluated using simulation models. There is a strong need for models that can help decision makers in the process industry to analyse the risks of abnormal situations in the supply chain and to assess possible solutions. While each one of the actors (i.e. stakeholders) can be considered as a problem owner, here the perspective of the oil refinery is chosen. The decision support tool described here is designed for one such decision maker in an industrial setting.

7.3.1 Formulate the decision problem

Next, the criteria, degrees of freedom, constraints and model are defined for the oil refinery case.

7.3.1.1 Criteria

There are different options for the criteria with which to choose the best alternative. As examples, one can look at the overall profit of the refinery (for a certain time frame), profit during the production cycle effected by the disturbance, other financial measures, but also non-economical criteria such as customer satisfaction. Profit P of refinery was chosen, 14 days after a disruption took place. This means that the effect of a disturbance will be simulated over the next two cycles of operation, during which new raw materials are ordered and products are dispatched. It is assumed that the impact of the disturbance would have worn off by then.

[1]Note that disturbances may be part of the normal operation. When a disturbance (or a series of disturbances) leads to a situation where the normal operating mode can no longer deal with it, this is called an abnormal situation.

The objective function for a period of 50 days is defined as follows:

$$\begin{aligned}\max_{\bar{x}} \quad P(\bar{x}, \bar{d}) = \sum_{t=1}^{50} (&Income^t_{\text{sales}}(\bar{x}, \bar{d}) - Cost^t_{\text{procurement}}(\bar{x}, \bar{d}) \\ &- Cost^t_{\text{transport}}(\bar{x}, \bar{d}) - Cost^t_{\text{maintenance}}(\bar{x}, \bar{d})) \\ &+ Value^{t=50}_{\text{product stock}}(\bar{x}, \bar{d}) + Value^{t=50}_{\text{raw materials}}(\bar{x}, \bar{d})\end{aligned} \tag{7.7}$$

In Equation 7.7 the monetary value of the product inventories and raw material stocks at the end of the simulation run are included. The consequences of the disruption on future cycles are included in the cost function (e.g. if the response is to switch to another mode of operation without any emergency procurement, it is possible that during a later cycle the planned operation cannot be met) but no new decisions following to any such new disturbances are assumed; a single response is formulated.

The function for the transportation costs, and therefore profit, is discontinuous because they are a function of the amount of crude procured, the capacity of the ships and the travel time. The transport cost is calculated per vessel, which could either be a very large crude carrier for long distance shipping or a general purpose tanker with much smaller capacity for short haul in the case of emergency procurement, and the unit landed-cost of crude (i.e. procurement plus transport costs) therefore follows a saw-tooth pattern. This discontinuity makes it more difficult to determine the right amount to buy, especially in combination with other measures such as switching to another recipe in the refinery.

7.3.1.2 Degrees of freedom

Faced with a disturbance (see Section 4.3.2), the problem owner has to make a number of choices. Firstly, he has to determine if the disturbance has a significant effect on the operation of the supply chain. If the effect is deemed minor, no action may be necessary, but if not able to execute the previously planned schedules due to insufficient crude, corrective action may be required. A disturbance in the supply of crudes can be addressed by changing the operating mode, the throughput or by emergency crude procurement. Often a combination of these actions may be needed.

For the Emergency Procurement $EmPr$, the procurement department can contact a local supplier to buy crude at a much higher price but with a shorter lead time. The procurement department has to ask the logistics department for the expected delay to be able to make this decision. Note that if there is an error in a storage installation ($StorageInstallationError = 1$), emergency procurement will not solve a disturbance. Crude cannot be transferred directly from the vessel to the CDU but it has to pass through storage tanks, as it needs to settle for a certain time period before it can be sent to the refinery to be processed.

Furthermore, the operations department can choose to Change the Operational Configuration (COC), meaning that a different recipe is selected using crudes that are still in stock, but resulting in yields that are not ideal compared to the scheduled operation. Finally, the operations department can Change the Operational Scale (COS), to run the refinery at a lower throughput producing less end products but avoiding having to shutdown the plant when crude runs out (or postponing plant shut-down, for example to allow emergency procurement crudes to arrive).

The degrees of freedom are defined as follows:

$$\bar{x} = (EmPr_1, EmPr_2, EmPr_3, EmPr_4, EmPr_5, COC, COS) \quad (7.8)$$

7.3.1.3 Constraints

The degree of freedom for each of the five crudes is between 0 kbbl to the amount that could reasonably be available on short notice, which is assumed 600 kbbl (kbbl stands for 1000 standard oil barrels). Furthermore, the number of different recipes in the refinery is assumed to be four, one of which is always selected as the current operational configuration. The CDU in the refinery has a minimum capacity as one of its design parameters; below 40% of the maximum throughput the process no longer works and the refinery has to be shut down. These constraints are defined as follows:

$$0 \leq EmPr_i \leq 600 \quad \text{in kbbl, for each of the } i = 5 \text{ crudes} \quad (7.9)$$

$$COC \in \{R1, R2, R3, R4\} \quad \text{discrete choice between operating modes} \quad (7.10)$$

$$40 \leq COS \leq 100 \quad \text{percentage of CDU throughput capacity} \quad (7.11)$$

An additional constraint on the system was defined by the problem owner, namely that the refinery should not be allowed to shut down after a disturbance but that the refinery has to stay operational at all times.

7.3.1.4 Model

The model used here is the one described in Section 4.3 and it was developed using the framework. The scope is limited to disturbances dealing with the supply of crude oil to the crude distillation units and they are defined by

$$\bar{d} = (ShipDelay, StorageProblem) \quad (7.12)$$

with

$$ShipDelay \in \{0, 1, 2, \ldots, n\} \quad \text{in days, for 1 ship for 1 cycle} \quad (7.13)$$

$$StorageProblem \in \{0, 1\}^m \quad \text{for each of the } m = 5 \text{ crude storage tanks} \quad (7.14)$$

For this case study it is assumed a disturbance to the system occurs on day $t = 22$. A ship at sea is delayed for 30 days, but there are no problems with any of the storage units. This means the disturbance is defined as $\bar{d}(22) = (30, 0, 0, 0, 0, 0)$.

7.3.2 Select the decision problem solving method

The Nelder-Mead optimisation method (See Section 7.1.2) is used here as the search strategy. It is not only chosen because of its ability to deal with discontinuous objective functions, but in also to illustrate how an optimisation method that is commonly used in process systems engineering using mathematical models or samples from experiments in a real system can also be a powerful approach when combined with an agent-based model.

7.3.3 Perform experiments

An initial simplex S for a 6-dimensional space (the number of degrees of freedom in $\bar{x}$) for the Nelder-Mead optimisation method needs to contain 7 vertices. Table 7.1 (rows 1 to 7) shows the initial values that were chosen, based on a first analysis of the problem by an expert. These values for the degrees of freedom provide enough variation for the search algorithm to proceed.

The stop condition for the search is when all $\bar{x}$ in the population have prevented a shut-down of the refinery and no better values for the criteria are found through a new iteration.

Figure 7.3a illustrates the crude stocks of the refinery under normal operation and Figure 7.3b shows the operational scale (both planned and actual throughput) over time after disturbance $\bar{d} = (30, 0, 0, 0, 0, 0)$ which occurs on day $t = 22$. This disturbance results in a loss of $24 million because of loss of production hours, as illustrated by the gap between planned and actual throughput. Next, the agent-based simulation model supports the choice which response is the most appropriate given the many degrees of freedom. Preliminary results for the decision on emergency procurement and the change of operational scale are shown in Table 7.1.

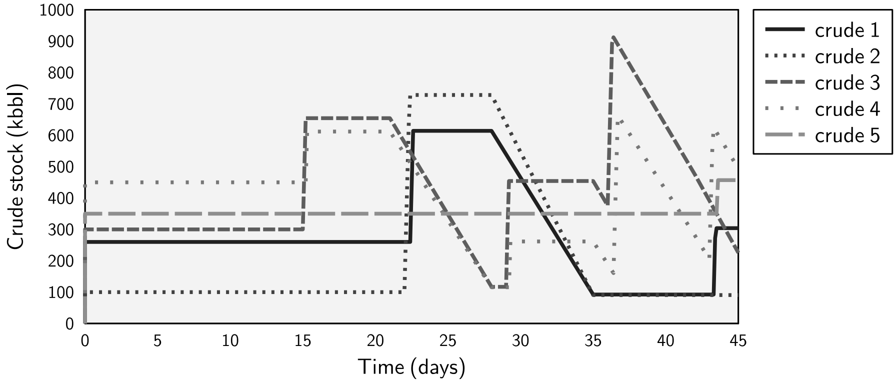
The change in operational configuration was not included in this experiment, rather the planned production mode (in this case $R4$) was used. For each possible $COC \in \{R1, R2, R3, R4\}$ an optimisation for $EmPr_i$ and COS has to be performed because choosing a different recipe will influence the criterion surface.

7.3.4 Analyse the results

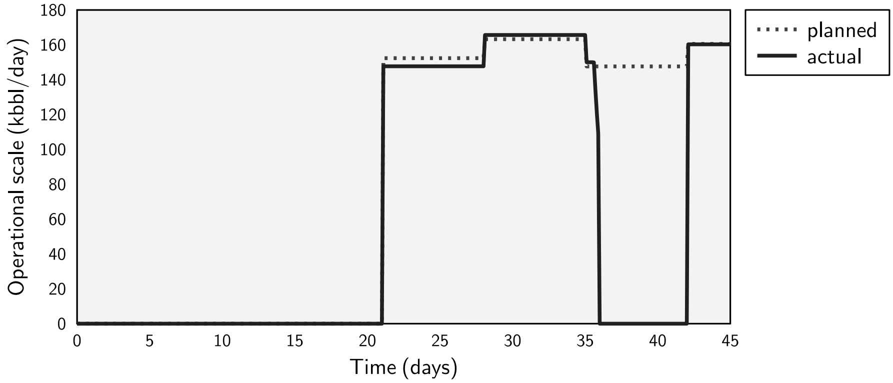
After 20 iterations (see Table 7.1) no further improvement is made so the search terminates. The proposed solution prevents a shut-down of the refinery by buying emergency crudes to make up for the delayed ship and by slightly reducing the throughput. The loss caused by the disruption is reduced by $14.7 million (excluding penalties to be paid by the shipper for delays) which is more than 60% of the $24 million damage caused by the disturbance.

Table 7.1 – *Outcome of the Nelder-Mead simplex optimization algorithm with the agent-based model*

	$EmPr_1$	$EmPr_2$	$EmPr_3$	$EmPr_4$	$EmPr_5$	COS	$P(\bar{x}, \bar{d})$
1	0	0	0	0	0	60	-5.69 E8
2	0	0	100	100	0	60	-3.65 E8
3	0	0	200	300	0	60	-1.61 E8
4	0	0	300	250	0	55	-1.64 E8
5	0	0	300	250	0	50	-6.19 E7
6	100	100	100	100	100	50	-3.70 E8
7	10	10	500	600	10	50	1.39 E8
final	4	4	349	382	4	54	1.45 E8



(a) *Crude stocks under normal operation*



(b) *Production scale under disturbance*

Figure 7.3 – *Results from the oil refinery supply chain case study*

7.3.5 Conclusions

An illustrative case study using an agent-based simulation model of a supply chain with the Nelder-Mead optimisation method was presented and it was demonstrated how agent-based models can be applied in a decision support tool. The approach presented can provide valuable support in choosing the right response to abnormal situations in a highly complex system such as a refinery supply chain, using a simulation model. Nelder-Mead optimisation is a commonly used method in process systems engineering using mathematical models or samples from experiments in a real system. It was demonstrated that it can also be a powerful approach when combined with an agent-based model.

Determining the right combination of options (e.g. a switch of operational configuration and an emergency procurement for those crudes instead of the ones in the delayed tanker) is difficult, and becomes even harder when responses at different times are allowed (e.g. a small emergency procurement now, switching recipe later, switching back when the delayed ship arrives, etc.) and when responses to new disturbances are also included (e.g. the long term effects of not always following planned operation). After formulating a decision problem, the simulation model can be used to find the right response again.

7.4 Conclusions

With two cases studies presented in this chapter, it is shown that models developed using the framework from Chapter 3 can be used to support decision makers in the area of socio-technical infrastructure systems.

Agent-based models are particularly suitable to experiment with different scenarios and to answer "what if" questions, which is critical for decision support under disruptions or in the design phase. The models used in this chapter were developed in a bottom-up fashion, making it relatively simple to change the configuration: it is easy to include new actors in the system (e.g. different users of the freight hub or more suppliers with different prices and lead times in the supply chain) or to adjust the physical configuration (e.g. additional transport links in the intermodal freight network or extra storage tanks for the refinery). Like in the examples of challenges listed in Section 1.3, the cases presented in this chapter required changes in physical and social networks (or both) to provide adequate decision support.

It should be stressed that the decision model is different from a simulation model in that it is designed for a specific purpose and question using a simulation model, where a simulation model can be used for a variety of purposes. For the oil refinery supply chain model used in this chapter, for example, various actors who make their own decisions are modelled, but for decision support the refinery company was chosen as the problem owner who has to make its decisions within an environment that is influenced by the behaviour of other actors.

The systems that were modelled with the framework represent real problems from the infrastructure domain. The freight hub case is a real policy case in the Queensland region in Australia and decision support was called for by the problem owner. An equation-based model that was applied in the detailed benchmarking study to compare with the agent-based oil refinery supply chain model in Chapter 6, has been used in an actual oil refinery in Singapore to recommend various procurement and storage policies. As such, the modelling approach combined with the decision support approach presented here meets actual demand from problem owners and can provide valuable support in making the right decision in such highly complex systems.

Chapter 8

Conclusions, discussion and future research

8.1 Conclusions

This thesis started with considering the major challenges encountered by strategic decision makers in large scale interconnected network systems. Each decision making entity is situated in a dynamic, multi-actor, multi-objective and multi-level jungle: it is part of a bigger system which is constantly changing, it has to cope with the actions of other actors who may have conflicting interests and values, and who operate on different levels of hierarchy. Decision makers often rely on models and simulations for support in the decision process to make better-informed decisions. Successful models should be able to "capture" both the physical and social reality of the system, their interactions with one another and the external dynamic environment, and they must allow users to experiment with *changes* in both the physical and the social network configuration. Building such models poses a significant challenge, which led to the main research question:

> ***What is a suitable modelling approach for socio-technical systems that allows the user to make changes in both social and physical networks and which can support strategic decision makers to experiment with "what-if" scenarios in a dynamic, multi-actor, multi-objective and multi-level world?***

Furthermore, additional research questions were formulated in Chapters 1 and 2:

- What does a suitable modelling approach for socio-technical systems look like?
- What are different categories of modelling paradigms?
- How can different modelling paradigms be compared in a well-defined way?
- How can an ontology be created that describes the relevant elements of socio-technical infrastructure systems, that can be applied to different domains and refined for specific cases?
- Which concepts should an ontology for socio-technical systems contain?

- What are the advantages of agent-based modelling compared to other computational modelling paradigms?
- How can agent-based models support decision makers?

In the following sections these questions are addressed by presenting the key findings from this thesis.

8.1.1 Modelling framework for socio-technical systems

To deal with the challenges that arise from socio-technical complexity a generic agent-based modelling framework has been developed (Chapter 3). This framework aims at supporting the modeller in quickly setting up new applications by re-using building blocks and allowing the connection of existing models to one another. The framework consists of the following three elements:

- **Interface** definition between components, between models, between developers and between developers and problem-owners;
- **Library** of source code that can be re-used; and
- **Procedures** on how to use the library and interface to build models.

The modelling framework can help to set up new models of socio-technical systems by following a number of modelling steps and, where possible, re-using already existing "building blocks" (e.g. facts, procedures, agents or technologies) from models developed during previous applications. When new elements are created for a specific case they can be fed back into the shared framework with the result that they are available for re-use. A basic set of class definitions for socio-technical systems was developed from a number of initial case studies and refined through subsequent applications.

The cornerstone of the approach is an *ontology* for the domain of socio-technical systems. It contains concepts that are generic to systems that fall within the scope of this thesis. The ontology can be expanded for specific cases. The ontology forms the interface needed to bring different aspects of the system (both social and physical) together and to interconnect different models. Besides inter-connectivity, the ontology offers *interoperability*.

The approach has been demonstrated in several case studies (Chapter 4). Even though the problems addressed in the case studies in this thesis are not of the same *scale* as the problems mentioned in Section 1.3 as the motivation for this research, the real challenge is not in the scale of the system but lies in the socio-technical complexity. Viewed from a socio-technical perspective, an oil refinery supply chain with its distributed, intelligent, autonomous entities (each with their own dynamics, goals, desires and plans) interacting with complex production technologies, is not much different from a liberalised electricity sector, for example.

The advantage of using smaller scale problems as illustrations in this thesis is that it made it possible to compare the approach presented with traditional modelling paradigms in which not the decision makers are modelled, but the observed results of their actions. From a common ground – systems where different approaches can all give valuable decision support – it became possible to extract the added value of the solution presented in this thesis, as the limitations of other approaches were revealed.

8.1.2 Categories of modelling paradigms

One conclusion that can be drawn from a study on the commonality among the various perspectives on agent-based modelling, is that there is not a clear line between agent-based and non-agent-based models (Chapter 2). The concept is not black-and-white, rather there is a continuous scale where a model can be more agent-based or less so. A way to visualise this was proposed. There are two main axes in which models can differ: The *model elements* axis and *system description elements* axis. The former deals with *what* is modelled and the constituents of the model, the latter with *how* their structure and behaviour is formally described. Where agent-based models are generally identified by the *model elements* axis, equation-based models are mostly classified by the *system description elements* axis. This also means that the use of equations is not the opposite of the agent-paradigm, nor is it an alternative per se, as is often stated. Rather, *agent* and *equation* are concepts of a different order.

Having emphasised that there is no “black” or “white” when it comes to the label *agent-based model* or *equation-based model*, models can be mapped on the space formed by the two axes to indicate their essential characteristics. This illustrates not only how the various models are different but also to what extent they are similar. This formulation, by acknowledging the absence of a clear dichotomy, makes stark contrasts more difficult, but, for a fair benchmarking, the similarities between models should *also* be fully captured.

8.1.3 Benchmarking modelling paradigms

Benchmarking is about making comparisons and, through these, learning generalisable lessons. It is not possible to compare modelling paradigms based only on the conceptual model specifications; rather a well-defined benchmarking process is required. To assess the performance of different modelling paradigms, a benchmarking scheme was proposed (Chapter 6). It can be concluded that special attention should be paid to the “identification of what is to be benchmarked” and “evaluation if objects of study are comparable” steps. In other comparisons between modelling paradigms found in literature this step was omitted, resulting in an unfair comparison whose results cannot be generalised.

8.1.4 Advantages of agent-based modelling

Agent-based models are particularly interesting to use for decision support when a change in structure (either social, technical, or both) is required, when social elements (i.e. actors performing a specific task) have to be combined with technical elements, or when a natural representation of interaction between system elements (at different levels) is important (Chapter 6). However, when changes in structure are not an intrinsic part of the problem or the solution or when the behaviour of actors is not a key element of the system description, other types simulation models might be more applicable.

Another distinct advantage lies in the re-use of elements of the model. Not only the conceptualisation could be re-used, but because of the bottom-up nature of the agent-based approach and the possibility to use of an ontology as an interface between “building blocks”, it is also possible to directly re-use source code from previous models. When it comes to explaining the model and the model results, agent-based modelling offers a natural representation of the decision making processes and interactions between the entities in the system.

Finally, it should be stressed that an agent-based model is not "mysterious" and "unpredictable", but rather a clearly defined computational model that can be deterministic and able to produce results that can be replicated.

8.1.5 Ontology development

The main challenge of ontology development is the formulation of a shared vocabulary that covers the different perspectives of the world needed to capture the socio-technical complexity (Chapters 3). Especially for an ontology that is intended to be used in various domains, it is important to get people from different backgrounds on one level and find commonalities. It was found that a successful way to develop such an ontology is through several iterations involving new applications. By working with several modellers on a relatively small set of domains – that is diverse enough but still closely related – and keeping the genericity of the concepts that are defined in mind, an initial version was developed which was improved by use. The ontology developed this way is the result of a joint effort with people from different disciplines.

There is a constant struggle between keeping the working process efficient for those who are contributing and extracting what they have in common. "Quick fixes" are tempting, but can be hard to clean-up later. It is recommended that one or more modellers specifically facilitate this process to guard the generic goal. A key part of this task is making sure developers understand the added value of working on a shared ontology above making their own class definitions.

8.1.6 Decision support

After a simulation model has been developed it can be used for decision support by formulating a decision problem and defining experiments (Chapter 7). It should be stressed that the decision model is different from a simulation model in that it is designed for a specific purpose and question using a simulation model, where a simulation model can be used for a variety of purposes. For the oil refinery supply chain model used in this thesis, for example, various actors who make their own decisions are modelled, but for decision support the refinery company was chosen as the problem owner who has to make its decisions within an environment that is influenced by the behaviour of other actors. The approach presented can provide valuable support in choosing the right response to abnormal situations in such a highly complex system.

8.2 Discussion

Next, several topics are discussed to address the limitations and wider application of the findings presented in Section 8.1.

8.2.1 The best approach?

The framework presented in Chapter 3 was demonstrated to be suitable for modelling socio-technical systems. But is it also the *best* approach? This question cannot be answered based on the work done so far. It should be said that this research was not an "optimalisation": the aim was to find *an* approach, not to necessarily find the best one.

It was a goal-oriented study targeted at solving a problem in an adequate way. The application of the framework to build various models (Chapter 4) and their use for decision support (Chapter 7) highlight that it can be useful for solving real problems and as such it is "good enough". When other approaches that can also fully capture the socio-technical complexity are developed, it would be more than interesting to perform a benchmarking study to learn about the advantages of the different approaches.

8.2.2 Scope

Even though the aim was to develop an integrated approach that can be used independent of the infrastructure domain, it cannot be claimed that the framework is universal and that everything can be modelled with it. In this thesis a sub-class of socio-technical systems was considered, namely those systems in which mass, energy or information is literally *transported* through a physical network where the nodes *transform* the mass, energy or information. The infrastructure is an engineered system and the organisational structure is in place to support this transfer or directly use it. The foundation of the ontology described in Chapter 3 was developed with this scope in mind, following a *process* system perspective. A consequence is that for a wide range of infrastructures and related problems, expressing the system in this language may not be useful. Even though the ontology is extensible, it can pose a challenge when the base of the ontology (namely the definition of social and physical nodes with different types of connections) does not naturally fit the system.

However, it should be stressed that the *approach* presented here does apply and the way-of-working is transferable. A new ontology for a different class of systems can be defined in the same way – possible re-using concepts of the ontology presented in this thesis. Again, an ontology can provide the interface between building blocks and between agents, and it can be expanded for future cases. The advantages of the approach and the suitability of agent-based paradigm for modelling socio-technical systems are independent of the definition of the ontology.

8.2.3 Benchmarking

The benchmarking study presented in Chapter 6 was done for supply chains. The benchmarking steps were followed for two models of the same oil refinery supply chain and conclusions were drawn in this context. Through the use of a well-defined process and the nuance of the labels of different modelling paradigms, it is expected the results can be generalised. However, it should be acknowledged that in a different domain than the one used in the benchmarking study (e.g. social sciences) a sharper demarcation between agent-based and equation-based models may exist.

8.2.4 Ontology languages

Different formal languages can be used for the *definition* of the ontology, but the approach is robust when changing the language used. The knowledge base reader is positioned in between the knowledge base and the models themselves, which makes it is possible to adjust the reader to allow all existing models to work with the ontology in its new format. The knowledge base reader acts as an "interpreter" and a new language can be added. The updated knowledge base reader enables reading the ontology in the new language and setting up the instances for use in the models. This also means a different ontology tool can be used instead of Protégé for the definition of classes and instances.

An example of a situation when a change of ontology language was needed, was the move from an ontology defined in *Frames*[1] to one described in the *OWL*[2] language. When the initial ontology (Section 3.4.1) was developed, Frames was the standard language supported by Protégé[3], but recently new projects required the higher expressive power of OWL which by now had also become the default language in the latest version of Protégé[4]. The ontology was automatically translated and a new knowledge base reader was implemented so that existing models did not need any changes in their source code. For the existing models everything remained the same, but still new possibilities opened up for reasoning with the ontology that were not available with a Frames ontology.

8.2.5 Distributed controllers

The agent-paradigm is also suitable for the design of distributed control systems, where the controllers have sensors to measure the environment and actuators to directly influence their surroundings plus the capability to communicate with one another (i.e. a physical box that runs a control algorithm, for example at a traffic light). In the framework presented in this thesis, an agent is seen as a model of a human decision maker and a part of the *social* network. For an agent-based distributed control system, however, the agent would be part of the *physical* system.

In both cases the agent-paradigm is a useful way of looking at the world. However, it should be acknowledged that building controller agents is a different activity. For the controller a new element that did not exist before is *created*. For the agent representing a human decision maker, on the other hand, something that is observed in the system is *modelled*. To be able to use the ontology from Section 3.3 to describe distributed controllers, an extension of the concepts is needed to allow nodes in the physical system to make decisions and communicate directly with other physical nodes.

8.2.6 Application in the industry

The main problem owner for this thesis was the modeller of socio-technical systems: one could think of a modeller working for a consultancy firm, a software engineering company, or perhaps employed in-house with one of the many stakeholders in different infrastructure sectors. The framework presented here, however, has so far only been used within one research group and it should be said that the framework and the ontology and building blocks that were developed cannot be used as an off-the-shelf solution.

For external parties the general approach and the generic elements of the ontology as presented in Chapter 3 are available and parts of the source code are available through a public license. The contents of the knowledge base, with detailed information of an array of technical systems, is not open because it contains propriety data. As modellers know, data-collection is one of the biggest challenges when building a model. On the one hand, it is to be expected that modellers in the private or public sector get access to data

[1] See http://protege.stanford.edu/overview/protege-frames.html.

[2] OWL stands for Web Ontology Language and is now the de facto standard by W3C. See http://www.w3.org/TR/owl-guide/.

[3] Protégé version 3.2.

[4] Protégé version 4.0, released 16th of June 2009.

from one or more stakeholders, but, on the other hand, detailed information from the competition might be hard to come by.

Before one can start building models with the framework it is important to get familiar with the way the ontology is structured and to experience how it has been applied in various domains. The ease of re-use for existing elements is highly dependent on the proficiency level of the programmer. Several new users without any programming experience have been able to successfully enter new data, expand the ontology and adjust several building blocks for their own case-specific problem, but it should be said that this comes at a significant time investment and requires support from an experienced modeller. Starting to model without such a foundation, however, will have a much higher cost. The framework can be of great support also to those without an extensive background in agent-based modelling.

After studying Chapter 3 for an introduction to the framework and Chapter 4 to see how the modelling steps are executed, the reader should be able to start mapping the system's elements onto the ontology presented here, so that a design for an agent-based model can be made which can be implemented. If the approach is followed to develop models, subsequently a set of re-usable "building blocks" is grown.

8.3 Recommendations for future research

In this section several recommendations for follow-up studies, based on the research performed for this thesis, are given:

- The knowledge base currently contains a description of *what* exists in the world that is modelled. The ontology provides an abstract definition and the instances in the knowledge base include the elements that play a role in the system, from the wallet of a key stakeholder to the required temperature for the inputs of a chemical process. For a technology it is defined *what* the inputs and outputs are for a conversion, but how the values change for different settings is not well addressed. *How* is often case-specific, but not unique. Behavioural "building blocks" should be integrated in the ontology so an agent can be defined with, for example, certain trading strategies or a specific investment strategy. Defining a shared language for "behaviour" is an important next step, which can build upon the work presented in this thesis.

- To model the technical system other modelling paradigms might be more fitting, as was concluded in the benchmarking study. The integration of different types of models inside the agent-based model should be explored and the inclusion of more detailed simulation of technical systems (to replace the input-output definition as is used in the OperationalConfigurations) should be studied. The ontology presented in this thesis can serve as an interface between the agent-based model (describing the behaviour of the actors) and the outcomes of specific simulations of the physical system.

- The ontology for socio-technical systems developed in this thesis does not build upon already existing ontologies, even though related ontologies for specific domains do exist. Sharing ontologies with others and re-using conceptualisations from others only strengthens the work. The transition to an OWL ontology makes connecting to other ontologies easier and facilitates the use of *Open Data* (i.e. data that is made available for public use). Additionally, the framework could take advantage of widely available libraries for, for example, the definition of unit names.

- Decision-making does not end with choosing the most preferred option; the analysis should also examine how the decision makers will respond to the existence of *uncertainty*. Therefore, it is necessary to identify the types of uncertainty which exist in and around the activity domain of each agent. The impact of uncertainty on decision support with agent-based models can be studied by expanding the models developed with the framework.

- A benchmarking study in several new application domain should be conducted to strengthen the conclusions on the advantages of the approach and add to verification and validation of the building blocks in the framework. New models are being developed based on the agent-based model used in the benchmarking study as well as based on the Matlab model. A follow-up benchmarking study based on those models could be a valuable addition to the results presented in this thesis too. Furthermore, a replication in a different modelling paradigm (e.g. system dynamics) could broaden the benchmarking conclusions.

8.4 Final remark

The framework development is an ongoing process through ongoing use; new modellers are using the approach for new cases and as such contribute to the shared framework. This is one of the key strengths of the approach: the more it is used, the more that can be re-used. Hereby the reader is invited to start thinking about challenges in the infrastructure domain from a socio-technical and agent-based perspective and to map the system's elements onto the ontology presented here, so that the modelling infrastructure can be used to effectively build better models and make the modelling process more efficient.

Appendices

Appendix A

Literature study on socio-technical modelling

A.1 Literature study approach

A search was executed with *socio-tech** and *model** as keywords, but a quick inventory of the results showed that too many results did not deal with computational models and simulations, and would therefore not be useful for this study. Instead of *model** the keyword *simulation* was used, because this implies that a model was created from which measures can be obtained[1]. The asterisk is used to include variations of the term *socio-technical*, such as *socio-technological*. In Section 2.2.1 *(TITLE-ABS-KEY(socio-tech*) AND TITLE-ABS-KEY(simulation))* was used as query in Scopus.

A.2 Application domains and background of authors

Table A.1 – *Application domains and background of authors for keywords 'socio-tech*' and 'simulation'*

Paper	Application domain	Background authors
Atkinson, Eldabi, Paul & Pouloudi (2001)	Health Informatics and Computing	Brunel University, medicine
Baker (1979)	Biology and impact environment	Penn State Faculty, anthropology
Barrett, Bisset, Eubank, Fox, Ma, Marathe & Zhang (2007)	Spread of infectious diseases	Virginia Tech, simulation
Basnyat et al. (2007)	Safety barriers	York, computer Science
Bergman et al. (2008)	Transport (transition to sustainability)	Oxford (environmental change) and Rotterdam (DRIFT, institute for transitions)

Table continued on next page...

[1] As Epstein (2008) says: "[...] when you close your eyes and imagine an epidemic spreading, or any other social dynamic, you are running some model or other. It is just an implicit model that you haven't written down".

Table A.1 continued from previous page...

Paper	Application domain	Author background
Böhmann & Loser (2005)	IT business	Ruhr-University of Bochum, management of information and technology
Brandt, Hartmann, Sander & Strina (1999)	Logistic chain / intermodal traffic	University of Technology Aachen, Computer Science
Carley (2002)	None specific	Carnegie Mellon University, social science
Clancey (1993)	Software development	Palo Alto, research on learning
Clegg & Frese (1996)	Computer-based systems	University of Sheffield
Cole (2006)	Financial portfolio management	University at Buffalo, Urban and Regional Planning
Cooke & Rohleder (2006)	Organisational response system	Haskayne School of Business
Davenport & Hall (2001)	Online work environments	Napier University Edinburgh, school of computing
Diehl (1974)	Worker safety and health	National Transportation Safety Board, the Federal Aviation Administration, the U.S. Navy and Air Force
Donzelli et al. (2004)	Infrastructures / e-government	University of Maryland, computer science
Donzelli & Bresciani (2003)	Requirements Engineering	University of Maryland, computer science
El-Hassan & Fiadeiro (2007)	Software components / organisational roles	University of Leicester, computer science
El-Seoud & El-Khouly (2004)	Software development	Helwan University, Faculty of Science
Eliasson & Persson (1996)	Electricity grid operators	Sydkraft AB
Emond & West (2003)	Human-computer interaction	Carleton University, Psychology & cognitive science
Faro & Giordan (2003)	Information systems design	University of Catania
Gaines & Norrie (1995)	Intelligent Manufacturing Systems	University of Calgary
Godbersen (1983)	Banking	Technische Fachhoschule Berlin
Goossenaerts (1993)	Manufacturing	University of Tokyo
Govindaraj (2008)	Interactions between human and automation	Georgia Institute of Technology, industrial and systems engineering
Gregoriades & Sutcliffe (2008)	Organisational processes simulation	University of Surrey, School of management
Gregoriades & Sutcliffe (2006)	Workload assessment / command and control rooms	University of Surrey, school of management
Grohn, Jalkanen, Haho, Nieminen & Smeds (1999)	Business process simulation	Helsinki University of Technology
Houwing et al. (2007)	Decentralised energy technologies	TU Delft, Technology, Policy and Management
Iivari & Hirschheim (1996)	Analysis and design of information systems	University of Oulu, information processing science
Jarman & Kouzmin (1990)	Space Shuttle disaster	University of Canberra, Faculty of management
Jenkins, Stanton, Salmon, Walker & Young (2008)	Military / allocation of function between actors	Brunel University, School of engineering and design
Jo & Dockery (1998)	Information warfare	Defence Information Systems Agency, Arlington

Table continued on next page...

Table A.1 continued from previous page...

Paper	Application domain	Author background
Johnson (2008)	Non specific	Open University, Milton Keynes, Department of design and innovation
Kember & Murray (1988)	Manufacturing system	Cranfield Institute of Technology Bedford
Kling, McKim & King (2003)	Communication forums on internet	Indiana University, Centre for social informatics
Kolan & Dantu (2007)	Spam phone calls (on VOIP)	University of North Texas
Leonard (1992)	Self-managing team motivation	Viable Systems Int., Toronto
Little, Birkland, Wallace & Herabat (2007)	Tsunami emergency warning system	University of Southern California
Little (2005)	Several catastrophic system failures	University of Southern California
Liu, Yoshikawa & Zhou (2005)	Nuclear energy	Kyoto University, Engineering
Lu & Cai (2000)	Collaborative design	University of Southern California, IMPACT research laboratory
Maciol & Stawowy (1993)	Information System for public administration	University of Mining and Metallurgy, Cracow
Maiden, Ncube, Kamali, Seyff & Grünbacher (2007)	Aircraft operations	City University London, Centre for HCI Design
Masys (2007)	Climate change	Synthetic Environment Coordination Office, Canadian Forces Experimentation Centre
McCown (2002)	Agriculture	Sustainable Ecosystems, Agricultural production systems research unit
McIntosh et al. (2005)	Freshwater resources management	Cranfield University, School of water sciences
McNally & Heavey (2004)	Manufacturing	University of Limerick, manufacturing engineering
McNeese et al. (2000)	Ergonomics	Wright-Patterson AFB, USAF research laboratory
Moscoso et al. (1999)	Manufacturing systems	Swiss Federal Institute of Technology Zurich, Institute for operations research
Nikitaev (1991)	Non specific	Scientific-Research Engineering Institute, Moscow
Nuutinen, Savioja & Sonninen (2007)	Vessel Traffic Services	VTT Technical Research Centre of Finland
Ottens & Marchau (2005)	Intelligent transport system	TU Delft, Technology, Policy and Management
Qudrat-Ullah (2008)	Electricity supply	York University
Ramanna et al. (2007)	Software engineering	University of Winnipeg, Applied Computer Science
Ramaswamy et al. (2007)	Telephone network	Los Alamos National Lab, Computer, computational and statistical sciences,
Rosenkranz & Holten (2007)	Design of information systems	Johann Wolfgang Goethe University, Information systems engineering
Saeed (1987)	Developing countries	Asian Institute of Technology, Bangkok
Savoyant & Ladure (1972)	Safety at work	EPHE, Paris, Lab. Psychol. Travail

Table continued on next page...

Table A.1 continued from previous page…

Paper	Application domain	Author background
Shah & Pritchett (2005)	Air traffic control	Georgia Institute of Technology, industrial and systems engineering
Shin (2004)	Broadband internet	Penn State University, Information science and technology
Shin et al. (2005)	Requirements engineering	University of Manchester, Human computer interface design, school of informatics
Simone (1989)	Office phenomena	Univerity of Milano
Sims & Henke (2007)	Nuclear weapons	Los Alamos National Lab, Statistical sciences group
Smajgl et al. (2008)	Non specific	Queensland University of Technology, Sustainable ecosystems
Sutcliffe et al. (2007)	Naval command and control system	University of Manchester, Human computer interface design, school of informatics
Thissen & Herder (2003)	Infrastructures	TU Delft, Technology, Policy and Management
van Oosterhout, Talmon, De Clercq, Schouten, Tange & Hasman (2005)	Electronic Patient Record	Maastricht University, Medical informatics
Yahja & Carley (2005)	City-scale social-networks	Carnegie Mellon University, Computation, organisations, and society program
Yilmaz (2007)	Software development	Auburn University, Modelling and simulation group
Zarboutis & Marmaras (2007)	Evacuation for metro	National Technical University of Athens, School of mechanical engineering

A.2.1 Completeness of study

Looking at the broader search of socio-technical system *models* (instead of simulation, which was the keyword used above), the top three authors (based on number of publications) are Gregoriades (e.g. 2006 and 2008) and Sutcliffe (e.g. 2007), with Lukszo (copromotor of this thesis), as the third. From this it can be concluded that no large research groups were missed by adding *simulation* to the keywords for this search instead of *modelling*, as these groups can all be found in Table A.1.

A.3 Modelling approaches for socio-technical systems

A.3.1 Literature review

Table A.2 shows a more detailed version of Table 2.1. In this appendix the definition of socio-technical system is detailed for those cases where it does not match. Also, a more elaborate discussion of the application domain is given for papers where it is not clear or where multiple domains are used.

Table A.2 – *Detailed review of modelling approaches for socio-technical systems (Table 2.1). '–' means no match, '+' means a match and '?' means that it is not clear. No answer indicates that the full paper was not online so it could not be determined. Papers were selected from Table A.1*

Paper	S-T Domain	S-T Definition	Simulation	Reproducable	Generalisable	Extendable
Basnyat et al. (2007)	–	–[2]	+	+	+[3]	?
Bergman et al. (2008)	+	+	+	?	+	?
Carley (2002)	+[4]	+	–	–	+	–
Donzelli et al. (2004)	–					
Eliasson & Persson (1996)	+					
Govindaraj (2008)	+	+	+	–	+	?
Gregoriades & Sutcliffe (2006)	–	–[5]	+	–	+	?
Gregoriades & Sutcliffe (2008)	–	–[6]	+	–	+	+
Iivari & Hirschheim (1996)	–	–[7]	–	–	+	–
Jarman & Kouzmin (1990)	–	–[8]	–	–	+	–
Johnson (2008)	–	+	–	–	+	–
Little (2005)	+[9]	+	–	–	+	?
Liu, Yoshikawa & Zhou (2005)	+					
Maciol & Stawowy (1993)	–					
Masys (2007)	+	?	–	–	+	–
McIntosh et al. (2005)	+	+	–	–	+	–
McNeese et al. (2000)	–	+	–			
Moscoso et al. (1999)	–	+	–	–	+	–
Nikitaev (1991)	–	–[10]	–			
Qudrat-Ullah (2008)	+	+	+	–	+	?
Ramanna et al. (2007)	–	–[11]	–	–	?	–
Ramaswamy et al. (2007)	+	+	+	+	+	?
Saeed (1987)	–	+	–	+	–	–
Shah & Pritchett (2005)	–[12]	+[13]	+[14]	–	+	+[15]
Shin et al. (2005)	?	–[16]				
Simone (1989)	–	–[17]	–	–	+	–
Smajgl et al. (2008)	?	?	–			
Sutcliffe et al. (2007)	–	+		–	?	–
Thissen & Herder (2003)	+	+	–	–	+	–
Yahja & Carley (2005)	–	–	+	?	+	?

Table continued on next page...

[2] Hardware and software
[3] Examples given in the paper include a cockpit and cash machine
[4] No specific domain is used, but potentially possible fit
[5] Command and control room
[6] Interaction between human and information system
[7] Information system in an organisation
[8] No explicit definition used
[9] Main case is a space shuttle, but the paper also discusses a power outage as an example
[10] Natural and artificial elements
[11] Users goals and computer possibilities
[12] Air traffic control
[13] Yes, but focus on humans in work environment
[14] Paper does not present simulation results, but model is capable of it
[15] Explicitly mentions this
[16] Operational problems causes by environment
[17] No explicit definition used

Table A.2 continued from previous page…

Paper	S-T Domain	S-T Definition	Simulation	Reproducable	Generalisable	Extendable
Yilmaz (2007)	–	–[18]	+	–	–	?
Zarboutis & Marmaras (2007)	–	+	+	+	–[19]	–

A.3.2 Selection of papers

Five papers were selected from Table 2.1 (cf. Table A.2) for detailed study in Section 2.2.3. See the '$\surd$' signs in the conclusions column in Table 2.1.

The first paper selected is Ramaswamy et al. (2007), which scores a '+' in all categories (except one question mark for extendibility, which can hopefully be resolved after studying the full paper and other work of the same authors in more detail) so it could potentially be an answer to the challenges posed in Section 1.4. The same can be said for Bergman et al. (2008) and Govindaraj (2008), for which it is again not clear how the models can be extended and also not to what extent they can be replicated. Still, they are promising papers. Basnyat et al. (2007) is also added to this list because, while the domain and socio-technical definition do not match, it is the only paper that was found which provides enough detail to reproduce the model and that can also be generalised, so this work could provide some valuable lessons on re-use of model components. This also means that a paper is included which is representable for the computer science and software engineering domain because that field was identified as one of the most prominent in Section 2.2.1. Finally, Shah & Pritchett (2005) is included in the selection for the detailed study because of its explicit separate social and technical aspects. Even though the main case study and the interpretation of the socio-technical system are not matching, it is interesting to consider how the explicit separation between was implemented and what can be learnt from this.

[18] Technical activities and procedures, strategic management

[19] Not completely generic, but can be used for other evacuation models

A.4 Agent-based modelling in the energy domain

A.4.1 Literature review

The Scopus query *TITLE-ABS-KEY(agent-based model*) AND (LIMIT-TO (SUBJAREA, "ENER"))* was used for Table A.3.

Table A.3 – *Agent-based models in the energy field*

Paper	Application domain	Method and aim
Awadallah & Morcos (2006)	Fault identification in DC motor	Adaptive neuro-fuzzy inference systems
Beck, Kempener, Cohen & Petrie (2008)	Planning of energy networks	Multi-objective optimization, then viability is explored through ABM[20].
Borrie & Özveren (2003)	System operator model	Experimenting with different trading arrangements
Botterud et al. (2007)	Generation expansion	The model simulates generation investment decisions of decentralized generating companies interacting in a complex, multidimensional environment
Bower & Bunn (2000)	Electricity markets	Test alternative trading arrangements, pool or bilateral
Bunn & Martoccia (2008)	Electricity market	Describes a platform
Chen, Yang, Zhang, Wang, Jing & Chen (2008)	Bidding strategy in day-ahead electricity	AMES open-source platform. Agents use auctions, Q-learning algorithm
Delgadillo, Gallego, Duarte, Jimenez & Camargo (2008)	Price behaviour of generation companies	Learning agents
Edwards, Srivastava, Cartes, Simmons & Wilde (2007)	Data communication security	Software agents as another possible channel of cyber attack
Ehlen, Scholand & Stamber (2007)	Electricity market	Uniform-price and real-time price simulation
Fekete, Nikolovski, Puzak, Slipac & Keko (2008)	Electricity market	Agents represent each market participant as an independent software agent. Uses EMCAS
Frezzi et al. (2007)	Electricity market	Liberalization, different levels of market concentration
Guerci, Rastegar, Cincotti, Delfino, Procopio & Ruga (2008)	Physical constrained. Electricity markets	Multi-agent learning algorithms, Q-Learning
Haxeltine et al. (2008)	Transitions framework	Same authors as Bergman et al. (2008)
Huang, Tong, Zhu & Ma (2004)	Supervisory information system	Information sharing within a power plant
Huang, Huang, Yang & Chen (2005)	Video monitoring	Paper is not about energy
Jackson (2007)	Standby rates of Combined Heat and Power	Cellular automata model
Jiang, Kang & Xia (2005b)	Electricity market bidding	Learning agents
Jiang, Kang & Xia (2005a)	Electricity market bidding	Same as (Jiang et al. 2005b)
Krause, Beck, Cherkaoui, Germond, Andersson & Ernst (2006)	Electricity market dynamics	Comparison of Nash equilibria analysis and ABM. Benchmarking

Table continued on next page...

[20] Agent-based modelling

Table A.3 continued from previous page…

Paper	Application domain	Method and aim
Lekov, Lutz, Whitehead & McMahon (2000)	Energy efficiency of residential water heaters	concept agent not used as in ABM, but as a substance
Li, Sun & Tesfatsion (2008)	Wholesale power market model	Strategic behaviour of actors. Uses own open-source testbed
Lincoln et al. (2006)	Short-term energy markets	Review of ABM in energy markets and need for common platform
Liu, Chen, Shen & Fan (2005)	Power system restoration after black-out	Actors and physical components modelled
Liu, Yang & Gan (2005)	Electricity market	Experimenting with different market rules
Müller, Sensfuß & Wietschel (2007)	Customers' engagement in the market	consumer model based on field data, profit maximizing suppliers
Ma, Feng, Yang, Wu & Fitch (2007)	Broker agent designed to bridge the gap between a user and a device	a broker agent method performs better than a client-server method
Morais et al. (2008)	Virtual power producers, cooperation	Presents MASCEM framework
Oh & Thomas (2007)	Deregulated electricity market	Based on a model developed elsewhere
Ortega-Vazquez & Kirschen (2008)	Investment in generation	Study effect of incentives. Presents 'toolbox'
Peerenboom (2001)	Failures and dependencies between infrastructures	methodologies and tools for characterizing and analysing such interdependencies
Saguan, Keseric, Glachant, Bastard & Haas (2005)	Effects of transmission capacity on market power	Combined with load flow model. Computational economics.
Scheidt & Pekala (2007)	Information theory, unclear application	distributed, agent-based control
Sheng, Jiang, Fan & Zeng (2005)	Voltage control	Distributed agent-based control
Sun & Tesfatsion (2007)	Wholesale power market	Experiment with market protocols
Tàbara, Roca, Madrid, Valkering, Wallman & Weaver (2008)	Water management	Describes vision of different tools for stakeholders to learn about complexity
Tellidou & Bakirtzis (2007)	Energy market	Experimenting with monopoly power, Q-learning
Thimmapuram et al. (2008)	Hydro power plants in the market, impact	Uses EMCAS framework, combined with model of hydro operation
Tong, Wang & Ding (2008)	Agent-based wide-area backup protection	Information exchange model
Trigo & Marques (2008)	Power market	Design of the environmental physical properties and entities, simulation of resources, decision-making
Tsoukalas (2001)	Power grid deregulation	On challenges for modelling
Vartanian, Law, Hines, Yinger, Hamilton & Feliachi (2008)	Electricity distribution circuit control	Agent-based control with simulations
Wang, Botterud, Conzelmann & Korintarov (2008)	German wholesale electricity market	Experiments with various bidding strategies, strategic bidding
Weaver & Jordan (2008)	EU Policy assessment, towards sustainability	Part of the MATISSE project.

Table continued on next page…

Table A.3 continued from previous page...

Paper	Application domain	Method and aim
Weidlich & Veit (2008a)	Interrelated electricity markets	Optimize trading strategies over the two electricity markets through reinforcement learning. Emission trading
Weidlich & Veit (2008b)	Electricity market dynamics	Literature survey on Agent-Based computational economics
Whitmarsh & Nykvist (2008)	Land-based mobility	Transition to sustainability, in MATISSE project
Xu, Wu, Hu & Zhang (2008)	Network planning	Multi-agent multi-objective planning, plans made with genetic algorithm
Yu & Liu (2008)	Various	Overview paper of ABM applications in energy. Uses JADE platform.
Yu, Scanlan & Wills (2007)	Operation of aircrafts	Comparing agent-based model with a traditional discrete-event model, benchmarking.
Yuan et al. (2005)	Electricity market	Literature survey on computational economics

Appendix B

Survey on agent-based systems

B.1 Introduction

To identify the level of commonality or divergence among the various perspectives on agent-based modelling a small survey was designed and sent to a group of researchers with a strong interest in, and contribution to, the agent-based systems area[1]. These results were previously published in concise form in van Dam, Adhitya, Srinivasan & Lukszo (2008).

B.2 Questionnaire

The following questions were used to find out how modellers, developers and other researchers interpret the concept of an "agent" and "agent-based systems":

1. What is your definition of an agent?
2. When can a model of a system be seen or defined as an agent-based model? In other words, what are the necessary and sufficient characteristics of an agent-based model?
3. What is the difference between agent-based control and distributed control?
4. Please give one example of a system or problem for which the agent-based model **is suitable** and one example of a system or problem for which it is **not suitable**.

B.3 List of answers

Table B.1 shows the answers given by the participants of the survey to the questions listed Section B.2. The answers have been anonymised before analysis and they are presented with only minor textual editing.

[1]Note that the aim is not to get a complete overview of all different views in the community, which would require a larger scale survey and a comprehensive literature review.

Table B.1 – *Answers to the survey on agent-based systems*

Respondent	Q1: Agent definition	Q2: Characteristics	Q3: Distributed control	Q4: Example domains
1	A software-entity capable of decision-making	That specific parts of the system can be described as agents	I think that distributed control problems can be modeled via the agent-based paradigm. As distributed control involves separate decision-making entities, I think every distributed control problem can be seen as a agent-based model.	Suitable to be modeled as agent-based model: a group people negotiating to set a date for a drink. Not suitable to be modeled as agent-based model: diffusion model of a gas in a membrane (differential equations better).
2	Agents are a programming technique, stemming from AI. It is based on the analogy with humans. The essential difference with simple straightforward programs or even object-oriented programs, is that agents are supposed to act simultaneously.	The agents must interact and run simultaneously, or simultaneously in a simulated way with a scheduler; then it's an agent-based system.	Agent-based control (which is the same as multi-agent control) is indeed quite similar to distributed control. In the latter there is more than one instance that can take control decisions. For instance, all the traffic signals in a town are a form of distributed control of urban traffic. The instances involved can be seen as agents. And they influence each other's behaviour via the traffic. The control system then has all the typical characteristics of an agent-based system.	Not suitable for agent based modelling: That are systems in which one cannot discern more than one typical agent. A system with only one actuator: the Maeslantkering in the Nieuwe Waterweg, for instance. Or the game of chess: the game is already linearised by its rules, only one piece moves at a time. Then agents do not help.

Table continued on next page…

Table B.1 continued from previous page...

Respondent	Q1: Agent definition	Q2: Characteristics	Q3: Distributed control	Q4: Example domains
3	An agent is a bundle of sensors, decision makers and actuators	An agent-based model is a reduced form description (model) of a multi-agent system. An agent-based economic model might use software agents in a software environment to model firms in a market.	Agent-based modelling can be used to simulate an actual distributed control system. In this case the "plant" is a mathematical model rather than a physical system. In a real distributed control system agents interact with a real plant, as opposed to a simulated one. I would consider any distributed controller an agent, so long as it has sensors, decision-makers and actuators. Some distributed controllers are intelligent (such as a sophisticated robot), whereas others are very simple (a thermostat for example).	A system that can be modeled using agent-based modelling: Firms competing to sell goods in a market. The firms are agents. The environment is the market. A system that is not readily amenable to agent-based modelling: A transistor. It might be possible to model individual atoms in a transistor as autonomous agents, but in most cases there would be little benefit to doing so.

Table continued on next page...

Table B.1 continued from previous page...

Respondent	Q1: Agent definition	Q2: Characteristics	Q3: Distributed control	Q4: Example domains
4	Autonomous entity with sensing and actuating capabilities, which strives for choosing its actions such that its objectives are achieved in its environment. Frequently an agent is a member of a collection of agents (a multi-agent system). In that case, in addition to sensing and actuating capabilities, the agent also has communication capabilities.	A model of a system is agent-based if agents are explicitly used to represent the dynamics of the system, i.e. the autonomous entities in the system are explicitly implemented and taken into account in the model. In an agent-based model at least 1 agent has to be considered.	An agent-based model is a model. Distributed control is a way of controlling a system. A model of a system can be used to determine which actions have to be taken to adequately control to system. If you say 'agent-based control' instead of 'an agent-based model', then they overlap to some extent. Agent-based control has the connotation of additional intelligence and autonomy of the local controllers/control agents. Also, agents can more easily be added or removed to the system.	Examples of multi-agent system large-scale systems: traffic network, electricity network, water network, gas network,... Examples not suited for multi-agent: If there is only one control input or highly coupled control inputs and outputs, or if there is only one centralised control agent, or if optimal performance is required for small-scale system
5	An agent can be defined as a software system that communicates and cooperates with other software systems to solve a complex problem that is beyond of the capability of each individual software system.	A software system (or subsystem) must have some level of "autonomy" in order to be considered as an "agent". Another important agent characteristic in my research domain is "proactiveness".	Distributed control system: distribution/decentralisation of data/information/knowledge. Agent-based control system: in addition to the distribution /decentralisation of data / information / knowledge, its control is also decentralised.	Agent-based approaches are particularly suitable for "real-time/dynamic scheduling" in order to achieve "zero response time" and "zero disruption" (to the regular operation); but they are not suitable for advance scheduling (scheduling of thousands of jobs/tasks over hundreds/thousands of resources for a period of a week or a month).

Table continued on next page...

Table B.1 continued from previous page...

Respondent	Q1: Agent definition	Q2: Characteristics	Q3: Distributed control	Q4: Example domains
6	Software agent is a complex software entity that is capable of behaving with a certain degree of autonomy, proactivity and an ability to communicate in peer to peer fashion.	Agent based models should consist of dynamically interacting agents. These interacting agents usually have goal-driven behaviour. Thus interacting behaviours create (or model) system's complexity. On the other hand, system's decomposition into agents is the way to reduce the complexity of its modelling and control.	Essentially, agent-based control implies dynamic decentralised interaction of autonomous entities.	Suitable: any distributed system (decentralised in nature) with autonomic behaviour of its components, like holonic manufacturing. Not-suitable: any centralised system without the need for autonomous actions and interactions of its components.
7	My original idea of agent comes from Micheal Woodridge, who defines it as an autonomous system which can perceive the environment around it, and keeps an internal state depending on it, while also has certain extent of intelligence to act to the changes of the environment according to its goal.	Philosophically, agent based modelling is an ontology. It is the superior level of object oriented programming. You can view anything through a pair of agent spectacle. While agent based modelling can be applied on any system, its real strength is the convenience on modelling complex interactive system.	Distributed control is the task, while agent based modelling is the methodology. Agent based model is the appropriate hardware/-software architecture for distributed control.	Given that all the clocks are exact and can go for ever, a clock in Belgium that needs to be switched between winter time and summer time is suitable to be modeled as an agent, which has interaction with other agents. While, a clock in China that does not needs to be switched is not necessary to be modeled as an agent.
8	An agent has at least semi-autonomy, a goal or objective, interactions with an external environment and flexibility to be both proactive and reactive.	A model is agent-based if it attempts to mimic the multiple entities that constitute a system separately from one another while including their interactions and allowing each entity within the system some form of freedom.	The difference in agent-based control and distributed control lies in the autonomy of the individuals and forms of interactions between them.	One example of a system where ABM is suitable is in the management and control of distinct entities. ABMs are not suitable for purely continuous physical systems such as fluid flow.

Table continued on next page...

Table B.1 continued from previous page...

Respondent	Q1: Agent definition	Q2: Characteristics	Q3: Distributed control	Q4: Example domains
9	Agent: any entity which "autonomously" performs actions of various types with the (possible) purpose to reach some goals.	I consider the above-mentioned autonomy a necessary and sufficient condition. Other concepts usually attached to agency (mobility, pro-activity, collaboration) are secondary.	I'd say it's the autonomous-looking behaviour that every program running at a control node shows.	The usefulness of an agent model to solve problems is rather disputable, especially now that after several years no "killer app" has been found yet. The only application in which apparent autonomy is not only suitable, but rather requested is gaming. Autonomous behaviour is rather resource consuming to implement, so I'd consider good practice leaving agents out of any hard real time system (e.g. air traffic control, biomedical systems).
10	An agent is an autonomous entity with simple decision rules and specific objectives to achieve.	A model of a system can be seen as an agent-based model when the modeller /analyst wants to address the heterogeneity of system elements (even geographic characteristics) in a particular system rather than the existing structure of it. The level of aggregation of information plays an important role.	A distributed control is mostly used in production/manufacturing systems for controlling equipment. An agent-based model is mostly used as an experimentation environment in order to observe emergent behaviour or structure.	The same system could be modelled as continuous or discrete (agent-based) depending on the focus of the problem. If we would like to focus on the traffic flow and jams in the Netherlands it is possible to work with a continuous model. If the focus is on the drivers' behaviour then an agent-based model will suit better the purpose.

Table continued on next page...

Table B.1 continued from previous page...

Respondent	Q1: Agent definition	Q2: Characteristics	Q3: Distributed control	Q4: Example domains
11	I like Stuart Kauffman's definition that "an agent is a thing that does things to other things". In a sense, an agent is like a noun, and we give it different verbs which it can use to connect to other nouns in its environment. More formally, an agent is a means to encapsulate our knowledge about actors and their behaviour within a discrete entity without explicitly defining the resulting structure of the overall system.	An agent based model should consist of discrete entities that are capable of perceiving and interacting with their environment, given the sets of rules defined for each agent.	An agent-based model isn't always about control. Sometimes it is exploratory and tries to see where the system "wants" to go given different parameters. A valid outcome could include finding that there is no way to control the system due to its internal dynamics. Depending on the implementation, distributed control could be seen as a subset of types of agent-based modelling.	The suitability of an ABM relates to the nature of a problem. If the mechanisms of a system can be accurately defined mathematically, then creating an ABM of the system becomes an academic exercise in showing that another pathway to a solution exists. ABM is valuable when we realize that it is impossible or too difficult to reduce the system to a set of equations. For instance, this occurs when we have problems of understanding how the behaviour of diverse actors (with different interests) is aggregated into the overall system behaviour. This is a natural fit for economic and industrial systems.

B.4 List of participants

The author of this thesis would like to express gratitude to the following people for contributing to the survey on perspectives on agent based systems: Bart De Schutter, Bri-Mathias Hodge, Catherine Chiong Meza, Chris Davis, Geert Deconinck, Igor Nikolic, Jos Vrancken, Leonid Sheremetov, Mario Verdicchio, Michiel Houwing, Paul Hines, Rudy Negenborn, Rui Duan and Weiming Shen.

Appendix C

Models built with the framework

C.1 Brief description of models

In this section a list of models that were built with the framework and ontology for socio-technical systems from Chapter 3 is presented. This list serves as an illustration of the wide scopes of systems and problems that can be addressed with the framework. Note that in some cases the project is still ongoing and has not been finalised yet. The models are listed in no particular order.

SchedulingDGModel A proof-of-concept model of distributed electricity generation units and exchange of power with the grid.

ChocolateGameModel An illustrative model of a chocolate bar production network with a supply chain for different products.

ElectricityMarketSimulationGame An educational serious game of the electricity sector, involving bidding on the power exchange and investing in new production capacity.

FailureModeAvoidance A model of the interaction between possible failures in products to discover how sensitive a system is to possible problems.

FlowBasedEvolution An illustrative model of the evolution of industrial clusters, in which industries connect to others based on the raw materials they need and products they can produce.

FreightHubModel A model of a multi-modal (sea, rail and road) freight transport system incorporating a freight hub.

HouseholdEnergyModel A model of electricity consumption and production at the household level, incorporating distributed technologies such as micro-Combined Heat and Power (μCHP).

IncomeDistributionModel An abstract model of the global economy to demonstrate inequality in income distribution.

IndustryInfrastructureCoEvolutionModel A detailed model of the development of large-scale socio-technical systems and the co-evolution between the technical system and the infrastructure. Extensive work on reasoning, for example with risk.

BulkBiochemicalsCase A model of bio-fuel and bulk chemicals clusters in The Netherlands and the impact of different prices and the introduction of new technologies to such a cluster.

CostaDueCase A detailed model of a regional industrial cluster in the port area of Groningen, the Netherlands, and the possible paths for a transition from petrochemicals to a bio-based cluster.

MetalsNetwork A model of a copper and aluminium production infrastructure through the full life-cycle, including different technologies for mining, smelting and converting.

SjoerdsGraduationVersion.1.1 A large scale model of the development of industrial clusters to increase understanding of the regional development of such clusters, including possible paths for a transition to a more sustainable future.

LubeOilModel A supply chain model of a lube additive production chain with multiple production sites, including scheduling activities for profit optimisation.

RefinerySupplyChainModel A detailed model of the operational behaviour of various actors in an oil refinery supply chain, including different operational departments within a single oil refinery company.

CarbonTaxationAndEmissionTrading-all-scenarios Full-scale model of different scenarios for carbon taxation and emission trading schemes and their effects on electricity generation portfolios.

ComparingCarbonPoliciesEffectOnPowerGeneration A model of the electricity generation infrastructure to study the impact of different carbon reduction policies.

EmissionTradingImpactOnPowerGeneration A model of investment decisions in the electricity sector and how they are influenced by emission trading.

TransitionInConsumerLighting A model of consumer behaviour in choice of lighting, to study a transition to more energy efficient lights and consumer adaption of new technologies.

TransitionInLNGMarkets A model of the transition to a spot market for liquid natural gas, as opposed to the current long term large volume market, including an implementation of how such a spot market could work.

CO2MarketTechnologicalInnovationModel A model of learning curves in technological developments and the introduction of new technologies to the market.

CO2MarketWindfallProfitsModel A model to study how *windfall profits* resulting from free emission trading rights can be prevented, so that the right incentives for sustainable generation of electricity can be introduced.

RobinsonCrusoe A model of scarcity and the effects on price, in which food is considered as an abstraction of oil and other natural resources.

OweSim A model of possible scenarios for large scale offshore wind energy, including actors such as wind farm installation companies, electricity contractors, wind farm developers and the permit office.

InvestmentStrategyEUETS A model of the carbon emission trading scheme as is currently considered by the European Union and its impact on risk and strategic decision making in companies.

C.2 Time planning

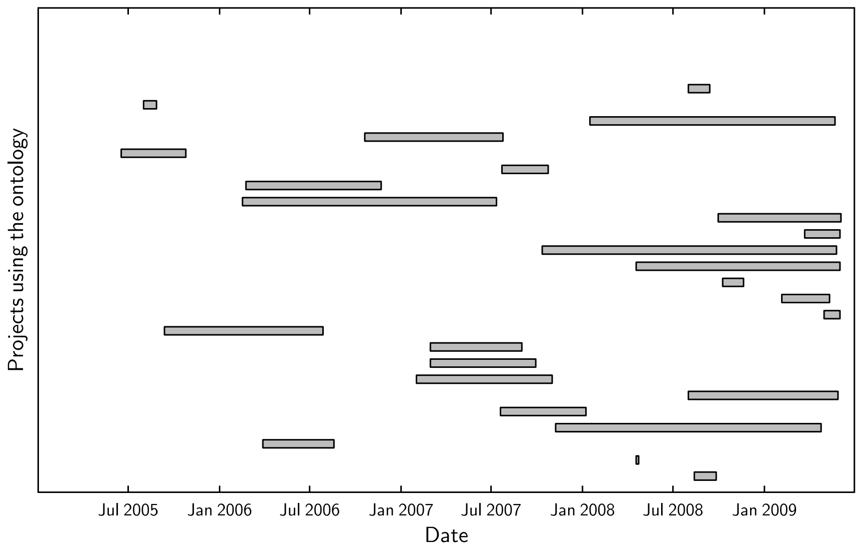


Figure C.1 – *Time planning for cases that use the shared ontology between July 2005 and June 2009. Every row is a separate model developed for a specific problem, with the bar indicating approximately when the project started and ended*

Figure C.1 shows the time planning for the projects listed in Section C.1. In Figure 5.2b a part of this image was already shown for the time period that was considered in Section 5.5.

Appendix D

Classes added to the ontology

D.1 List of classes added

Table D.1 shows a list of classes that were added to, or removed from, the shared ontology. Revision numbers that are not listed contain no changes in the class definitions (e.g. because only instances were added).

Table D.1 – *Classed added to, or removed from, the shared knowledge base*

Revision	Added/Removed	Classes
195	added	EnergyContent, CoordinateTuple, PhysicalEdge, SocialEdge, LiquidAssets, Plot, Content, OxygenContent, MoistureContent, CreationTime, PhysicalNode, CapitalAssets, SocialNode, CarbonContent, MetalContent, Ownership
195	removed	Flow, Optimisation
198	added	Time, DestructionTime
199	added	CriteriaTuple, PhysicalFlowContract, MultiCriteriaAnalysis
199	removed	NonUnique, TradeContract, Unique
206	added	MetalsNetwork
210	added	EnvironmentalEmission, Distance, BioElectricity
212	added	Origin, Fossil, Renewable
215	added	EnergyComposition, Composition, MassComposition, CompositionTuple
215	removed	EnergyContent, Content, OxygenContent, MoistureContent, CarbonContent, MetalContent
224	added	Components
224	removed	Composition
235	added	BiobasedBulkChemicals
256	added	Cluster2, Cluster3, Cluster1
258	added	BioBasedBulkChemicals
258	removed	BiobasedBulkChemicals
262	added	BiobasedBulkChemicals
262	removed	BioBasedBulkChemicals
274	removed	Cluster2, Cluster3, Cluster1
276	added	Cluster2, Cluster3, Cluster1
282	added	RecursiveNegotiationCase

Table continued on next page...

Table D.1 continued from previous page...

Revision	Added/Removed	Classes
287	added	S03PolicyStimulation, Te5NumberOfPossibleRawMaterials, Te6ProcessComplexity, S02AwarenessOfProduct, S05ImageOfProducts, S04ProcessRisk
290	added	MCA, Knows
297	added	User, Firm, FailureMode, EnvironmentalFactor, FailureRecord, ProductDevelopmentProcess, Designer, ComponentInterface, Environment, EnvironmentalFactorInteraction, Component, Product, TestEnvironment, Coutermeasure
299	removed	User, Firm, FailureMode, EnvironmentalFactor, FailureRecord, ProductDevelopmentProcess, Designer, ComponentInterface, Environment, EnvironmentalFactorInteraction, Component, Product, TestEnvironment, Coutermeasure
373	added	OweSim
413	added	EcoInventTuple
414	added	PowerPlant
418	added	RefinerySupplyChain
428	added	CreationTimeStamp
454	added	ProcessInstallation, StorageInstallation
460	added	Model, Player, Game, SimulationModel
465	added	ShoulderLoad, Scenario, TimeSeriesTuple, Operational, ElectricityMarketSimulationScenarioSpace, Dismantled, BidTuple, ElectricityMarketSimulationGameLabel, GoodPriceSequence, BidForm, ElectricityMarketSimulationRound, Status, Operator, ElectricityMarketSimulationScenario, PeakLoad, UnderConstruction, IdentityTuple, Unavailable, BaseLoad, ScenarioSpace, Fuel, NewsTuple, ElectricityMarketSimulationGame, UnavailableNextRound, Round, ElectricityMarketSimulationPlayer
487	added	Team, ElectricityMarketSimulationTeam
487	removed	Player, ElectricityMarketSimulationPlayer
489	added	Player
491	added	FuelConsumption, DefaultLabel, Loan, Availability, LevelizedCapitalCost, MarketLabel, FixedOperatingAndMaintenanceCost
492	added	EnergyEfficiency
492	removed	FuelConsumption
500	added	ConstructionRoundNumber, plantTuple, TimeSeriesGoods
500	removed	GoodPriceSequence, ScenarioSpace
501	added	PlantTuple
501	removed	plantTuple
535	added	LNGSpotMarket, TransportInstallation
537	added	Option, RealOptions, Possibility
557	added	SupplyTrend, LNGSpotMarketScenarioSpace, DemandTrend, PriceTrend, LNGSpotMarketScenario
564	added	RiskAttitude
583	added	RangeNormalization, NormalizationLabel, GoodNeutralNormalization
600	added	AverageSpeed, Vehicle, MaximumSpeed
600	removed	TransportInstallation
629	added	RobinsonCrusoe
698	added	ExpectedValueCriterion, MostProbableFutureCriterion, Risk, BusinessAsUsualCriterion, PavlovCriterion, AspirationLevelCriterion
699	added	RiskDecisionCriteria
699	removed	Risk

Table continued on next page...

Table D.1 continued from previous page...

Revision	Added/Removed	Classes
715	added	TransitionsConsumerLighting
764	added	MobilePhoneRecyclingNetworkTestConfiguration
767	added	Condition, EnergyAndIndustryKnowledgeBase_Class20045, Used, Functional, NonFunctional, New, ReadyForCollection, Functionality, Unspecified, Refurbished, Situation, YearOfManufacture, ReadyForConsumption
770	added	CanBeRefurbished, CannotBeRefurbished, PossibilityForRefurbishing
812	added	Failed, Available
812	removed	EnergyAndIndustryKnowledgeBase_Class20045

Bibliography

Aldea, A., Bañares Alcántara, R., Jiménez, L., Moreno, A., Martínez, J. & Riaño, D. (2004). The scope of application of multi-agent systems in the process industry: Three case studies, *Expert Systems with Applications* **26**(1): 39–47.

Atkinson, C., Eldabi, T., Paul, R. & Pouloudi, A. (2001). Investigating integrated socio-technical approaches to health informatics, *Proceedings of the Hawaii International Conference on System Sciences*, p. 153.

Awadallah, M. A. & Morcos, M. M. (2006). Automatic diagnosis and location of open-switch fault in brushless dc motor drives using wavelets and neuro-fuzzy systems, *IEEE Transactions on Energy Conversion* **21**(1): 104–111.

Baker, P. T. (1979). The use of human ecological models in biological anthropology: examples from the Andes, *Collegium Antropologicum* **3**(2): 157–171.

Barrett, C. L., Bisset, K., Eubank, S., Fox, E. A., Ma, Y., Marathe, M. & Zhang, X. (2007). *A scalable data management tool to support epidemiological modeling of large Urban regions*, Vol. 4675 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg.

Basnyat, S., Palanque, P., Schupp, B. & Wright, P. (2007). Formal socio-technical barrier modelling for safety-critical interactive systems design, *Safety Science* **45**(5): 545–565.

Bechhofer, S., Horrocks, I., Goble, C. A. & Stevens, R. (2001). OilEd: A reason-able ontology editor for the semantic web, *in* F. Baader, G. Brewka & T. Eiter (eds), *KI/ÖGAI*, Vol. 2174 of *Lecture Notes in Computer Science*, Springer, pp. 396–408. ISBN 3-540-42612-4.

Beck, J., Kempener, R., Cohen, B. & Petrie, J. (2008). A complex systems approach to planning, optimization and decision making for energy networks, *Energy Policy* **36**(8): 2803–2813.

Behdani, B., Lukszo, Z., Adhitya, A. & Srinivasan, R. (2009). Agent based model for performance analysis of a global chemical supply chain during normal and abnormal situations, *in* J. Jezowski & J. Thullie (eds), *Proceedings of to the European Symposium on Computer Aided Process Engineering ESCAPE-19*, Vol. 26 of *Computer-Aided Chemical Engineering*, Elsevier, Krakow, Poland.

Bergman, N., Haxeltine, A., Whitmarsh, L., Köhler, J., Schilperoord, M. & Rotmans, J. (2008). Modelling socio-technical transition patterns and pathways, *Journal of Artificial Societies and Social Simulation (JASSS)* 11(3).

Billari, F. G., Fent, T., Prskawetz, A. & Scheffran, J. (eds) (2006). *Agent-Based Computational Modelling: applications in Demography, Social, Economic and Environmental Sciences*, Contributions to Economics, Physica-Verlag, Heidelberg, Germany. ISBN 379081640X.

Birta, L. G. & Özmizrak, F. N. (1996). A knowledge-based approach for the validation of simulation models: the foundation, *ACM Trans. Model. Comput. Simul.* **6**(1): 76–98.

Blokker, S. B. (2006). *Co-evolution in the process industry: an agent based model for the eemshaven chemical cluster*, Master's thesis, Delft University of Technology.

Boero, R., Castellani, M. & Squazzoni, F. (2004). Micro behavioural attitudes and macro technological adaptation in industrial districts: an agent-based prototype, *Journal of Artificial Societies and Social Simulation (JASSS)* 7(2).

Böhmann, T. & Loser, K. U. (2005). Towards a service agility assessment — modeling the composition and coupling of modular business services, *Proceedings of the 7th IEEE International Conference on E-Commerce Technology Workshops*, Vol. 2005, pp. 140–148.

Bontekoning, Y. M., Macharis, C. & Trip, J. J. (2004). Is a new applied transportation research field emerging? — a review of intermodal rail-truck freight transport literature, *Transportation Research Part A: Policy and Practice* 38(1): 1–34.

Borodin, A. (1975). *Computational complexity of algebraic and numeric problems*, Vol. 1 of *Theory of computation series*, Elsevier computer science library.

Borrie, D. & Özveren, C. S. (2003). Modelling the role of the system operator under the new energy trading arrangements, *Proceedings of the Universities Power Engineering Conference*, Vol. 38, pp. 296–299.

Borshchev, A. & Filippov, A. (2004). From system dynamics and discrete event to practical agent based modeling: reasons, techniques,tools, *Proceedings of the 22nd International Conference of the System Dynamics Society*, Oxford, England.

Botterud, A., Mahalik, M. R., Veselka, T. D., Ryu, H. S. & Sohn, K. W. (2007). Multi-agent simulation of generation expansion in electricity markets, *Proceedings of the IEEE Power Engineering Society General Meeting, PES2007*.

Bower, J. & Bunn, D. W. (2000). Model-based comparisons of pool and bilateral markets for electricity, *Energy Journal* 21(3): 1–29.

Bradshaw, J. (1996). KAoS: an open agent architecture supporting reuse, interoperability, and extensibility, *Proceedings of the 1996 Knowledge Acquisition Workshop (KAW96)*, Banff, Alberta, Canada, University of Calgary.

Brandt, D., Hartmann, E., Sander, C. & Strina, G. (1999). Designing and simulating sociotechnical systems: Concepts and strategies, *Human Factors and Ergonomics In Manufacturing* **9**(3): 245–252.

Bratman, M. (1999). *Intention, plans, and practical reason*, CSLI Publications, Stanford, CA. ISBN 978-1575861920.

Braunschweig, B. & Joulia, X. (eds) (2008). *Proceedings of the European Symposium on Computer Aided Process Engineering ESCAPE 18*, Vol. 25 of *Computer-Aided Chemical Engineering*, Elsevier, Amsterdam, the Netherlands.

Bunn, D. W. & Martoccia, M. (2008). Analyzing the price-effects of vertical and horizontal market power with agent based simulation, *Proceedings of the IEEE Power and Energy Society 2008 General Meeting: Conversion and Delivery of Electrical Energy in the 21st Century, PES2008*.

Bussmann, S., Jennings, N. R. & Wooldridge, M. (2004). *Multi-agent systems for manufacturing control : a design methodology*, Series on Agent Technology, Springer-Verlag, Berlin, Germany.

Cameron, I. T. & Ingram, G. D. (2008). A survey of industrial process modelling across the product and process lifecycle, *Computers and Chemical Engineering* 32(3): 420–438.

Carley, K. M. (2002). Computational organizational science and organizational engineering, *Simulation Modelling Practice and Theory* **10**(5-7): 253–269.

Cavalieri, S., Macchi, M. & Valckenaers, P. (2003). Benchmarking the performance of manufacturing control systems: design principles for a web-based simulated testbed, *Journal of Intelligent Manufacturing* 14(1): 43–58.

Chaib-draa, B. & Müller, J. (eds) (2006). *Multiagent based supply chain management*, Vol. 28 of *Studies in Computational Intelligence*, Springer.

Chappin, E. J. L. (2006). *Carbon dioxide emission trade impact on power generation portfolio, agent-based modelling to elucidate influences of emission trading on investments in Dutch electricity generation*, Master's thesis, Delft University of Technology.

Chappin, E. J. L. & Dijkema, G. P. J. (2009). On the impact of co2 emission-trading on power generation emissions, *Technological Forecasting and Social Change* **76**: 358–370.

Chappin, E. J. L., Dijkema, G. P. J., van Dam, K. H. & Lukszo, Z. (2007). Modeling strategic and operational decision-making — an agent-based model of electricity producers, *Proceedings of the 21st annual European Simulation and Modelling Conference (ESM2007)*, St. Julian's, Malta.

Chappin, E. J. L., Dijkema, G. P. J. & Vries, L. J. d. (2009). Carbon policies: do they deliver in the long run?, *in* P. Sioshansi (ed.), *Carbon Constrained: Future of Electricity*, Global Energy Policy and Economic Series, Elsevier. in press.

Chatfield, D. C., Hayya, J. C. & Harrison, T. P. (2007). A multi-formalism architecture for agent-based, order-centric supply chain simulation, *Simulation Modelling Practice and Theory* 15: 153–174.

Chen, H., Yang, Y., Zhang, Y., Wang, Y., Jing, Z. & Chen, Q. (2008). Realization of decision-making module in agent-based simulation of power markets, *Dianli Xitong Zidonghua/Automation of Electric Power Systems* 32(20): 22–26.

Cirillo, R., Thimmapuram, P. R., Veselka, T. D., Koritarov, V., Conzelmann, G., Macal, C., Boyd, G., North, M., Overbye, T. & Cheng, X. (2006). Evaluating the potential impact of transmission constraints on the operation of a competitive electricity market in Illinois, *Technical Report ANL-06/16*, Argonne National Laboratory, Argonne, IL, USA.

Clancey, W. J. (1993). Knowledge level reinterpreted: modeling socio-technical systems, *International Journal of Intelligent Systems* 8(1): 33–49.

Clegg, C. W. & Frese, M. (1996). Integrating organizational and cognitive approaches towards computer-based systems, *Behaviour and Information Technology* 15(4): 203–204.

Cole, S. (2006). Portfolio theory for regional disaster analysis and policy, *Journal of Security Education* 1(4): 27–38.

Cooke, D. L. & Rohleder, T. R. (2006). Learning from incidents: from normal accidents to high reliability, *System Dynamics Review* 22(3): 213–239.

Darwin, C. R. (1859). *On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life*, John Murray.

Davenport, E. & Hall, H. (2001). New knowledge and micro-level online organization: 'communities of practice' as a development framework, *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, p. 92.

Davidsson, P. & Wernstedt, F. (2002). Software agents for bioprocess monitoring and control, *J. of Chemical Technology and Biotechnology* 77(7): 761–766.

de Vries, L. J. & Chappin, E. J. L. (2009). Electricity market simulation game, http://emg.tudelft.nl/.

de Vries, L. J., Subramahnian, E. & Chappin, E. J. L. (2009). Power games: using an electricity market simulation game to convey research results, *Proceedings of the second International Conference on Infrastructure Systems 2009 (INFRA 2009): Developing 21st Century Infrastructure Networks*, Chennai, India. submitted for review.

Delgadillo, A., Gallego, L., Duarte, O., Jimenez, D. & Camargo, M. (2008). Agent learning methodology for generators in an electricity market, *Proceedings of the IEEE Power and Energy Society 2008 General Meeting: Conversion and Delivery of Electrical Energy in the 21st Century, PES2008*.

Dennet, D. (1987). *The Intentional Stance*, MIT Press, Cambridge. ISBN 0-262-04093-X.

Diehl, A. E. (1974). Computer simulation model to analyze occupational safety and health problems and countermeasures., pp. 92–95.

Dignum, F., Dignum, V., & Jonker, C. M. (2008). Towards agents for policy making, *Proceedings of the 9th International Workshop on Multi-Agent-Based Simulation (MABS)*, Estoril, Portugal.

Donzelli, P. & Bresciani, P. (2003). Domain ontology analysis in agent-oriented requirements engineering, *Knowledge-Based Intelligent Information and Engineering Systems*, Vol. 2773 LNCS part 1, pp. 1372–1379.

Donzelli, P., Setola, R. & Tucci, S. (2004). Identifying and evaluating critical infrastructures — a goal-driven dependability analysis framework, *Communications in Computing*, pp. 245–254.

Edgar, T. F., Himmelblau, D. M. & Lasdon, L. S. (2006). *Optimization of chemical processes*, 2nd edn, McGraw-Hill, New York, USA. ISBN 0-07-118977-7.

Edwards, D., Srivastava, S. K., Cartes, D. A., Simmons, S. & Wilde, N. (2007). Implementation and validation of a multi-level security model architecture, *2007 International Conference on Intelligent Systems Applications to Power Systems, ISAP*.

Ehlen, M. A., Scholand, A. J. & Stamber, K. L. (2007). The effects of residential real-time pricing contracts on transco loads, pricing, and profitability: Simulations using the N-ABLE agent-based model, *Energy Economics* **29**(2): 211–227.

El-Hassan, O. & Fiadeiro, J. L. (2007). Role-based architectural modelling of socio-technical systems, *Electronic Notes in Theoretical Computer Science* **181**(1): 5–17.

El-Seoud, S. A. & El-Khouly, M. M. (2004). Information system design methodologies: software development and methods of integration, *Proceedings of the 2004 International Conference on Information and Communication Technologies: From Theory to Applications*, pp. 573–574.

Eliasson, B. E. & Persson, A. (1996). Prototype support system for generation of switching instructions, *Proceedings of the Fourth International Conference on Power System Control and Management*, pp. 165–170.

Emond, B. & West, R. L. (2003). Cyberpsychology: a human-interaction perspective based on cognitive modeling, *Cyberpsychology and Behavior* **6**(5): 527–536.

Eo, S. Y., Chang, T. S., Shin, D. & Yoon, E. S. (2000). Cooperative problem solving in diagnostic agents for chemical processes, *J. Computers and Chemical Engineering* **24**(2–7): 729–734.

Epstein, J. (1999). Agent-based computational models and generative social science, *Complexity* **4**(5).

Epstein, J. (2007). *Generative Social Science: Studies in Agent-Based Computational Modeling*, Princeton University Press, Princeton, NJ. ISBN 0691125473.

Epstein, J. M. (2008). Why model?, *Journal of Artificial Societies and Social Simulation (JASSS)* **11**(4 12).

Faro, A. & Giordan, D. (2003). Design memories as evolutionary systems socio-technical architecture and genetics, *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC)*, Vol. 5, pp. 4334–4339.

Fekete, K., Nikolovski, S., Puzak, D., Slipac, G. & Keko, H. (2008). Agent-based modelling application possibilities for croatian electricity market simulation, *5th International Conference on the European Electricity Market, EEM2008*.

Fisher, R. A. (1935). *The design of experiments*, Oliver & Boyd. ISBN 0-02-844690-9.

Forgy, C. L. (1982). Rete: a fast algorithm for the many pattern/many object pattern match problem, *Artificial Intelligence* 19(1): 17–37.

Frezzi, P., Garcés, F. & Haubrich, H. J. (2007). Analysis of short-term bidding strategies in power markets, *2007 IEEE Lausanne POWERTECH, Proceedings*, pp. 971–976.

Gaines, B. R. & Norrie, D. H. (1995). Knowledge systematization in the international IMS research program, *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC): Intelligent Systems for the 21st Century*, Vol. 1, pp. 958–963.

Gamma, E., Helm, R., Johnson, R. & Vlissides, J. (1995). *Design patterns: elements of reusable object-oriented software*, Addison Wesley. ISBN 0-201-63361-2.

Gani, R. & Grossmann, I. E. (2007). Process systems engineering and CAPE — what next?, *in* V. Plesu & P. Agachi (eds), *Proceedings of the European Symposium on Computer Aided Process Engineering ESCAPE 17*, Vol. 24 of *Computer-Aided Chemical Engineering*, Elsevier, Bucharest, Romania.

Garcia-Flores, R. & Wang, X. Z. (2002). A multi-agent system for chemical supply chain simulation and management support, *Operations Research Spectrum* 24(3): 343–370.

Geels, F. W. (ed.) (2002). *Understanding the dynamics of technological transitions: a co-evolutionary and socio-technical analysis*, Twente University Press, Enschede, the Netherlands.

Geiger, R. (1992). Science, universities, and national defense, 1945-1970, *Osiris* 7: 26–48.

Gennari, J. H., Musen, M. A., Fergerson, R. W., Grosso, W. E., Crubezy, M., Eriksson, H., Noy, N. F. & Tu, S. W. (2003). The evolution of Protege: an environment for knowledge-based systems development, *International Journal of Human-Computer Studies* 58(1): 89–123.

Gjerdrum, J., Shah, N. & Papageorgiou, L. G. (2000). A combined optimisation and agent-based approach for supply chain modelling and performance assessment, *Production Planning and Control* 12: 81–88.

Godbersen, H. P. (1983). Simulation with 'FUN'., *Angewandte Informatik, Applied Informatics* 25(5): 213–219.

Goossenaerts, J. (1993). Enterprise formulae and information infrastructures for manufacturing, *IFIP Transactions B: Computer Applications in Technology*, number B-14, pp. 123–140.

Govindaraj, T. (2008). Characterizing performance in socio-technical systems: a modeling framework in the domain of nuclear power, *Omega* 36(1): 10–21.

Gregoriades, A. & Sutcliffe, A. (2008). A socio-technical approach to business process simulation, *Decision Support Systems* 45(4): 1017–1030.

Gregoriades, A. & Sutcliffe, A. G. (2006). Automated assistance for human factors analysis in complex systems, *Ergonomics* 49(12-13): 1265–1287.

Grohn, M., Jalkanen, J., Haho, P., Nieminen, M. & Smeds, R. (1999). Visualizing human communication in business process simulations, *Proceedings of SPIE - The International Society for Optical Engineering* 3643: 140–148.

Grossmann, I. E. & Westerberg, A. W. (2000). Perspectives on process systems engineering, *AIChE Journal* 46(9): 1700–1703.

Gruber, T. R. (1993). A translation approach to portable ontology specifications, *Knowledge Acquisition* 5(2): 199–220.

Guerci, E., Rastegar, M. A., Cincotti, S., Delfino, F., Procopio, R. & Ruga, M. (2008). Supply-side gaming on electricity markets with physical constrained transmission network, *5th International Conference on the European Electricity Market, EEM2008*.

Hansman, R. J., Magee, C., Neufville, R. D., Robins, R. & Roos, D. (2006). Research agenda for an integrated approach to infrastructure planning, design, and management, *International Journal of Critical Infrastructures* 2(2/3): 146–159.

Haxeltine, A., Whitmarsh, L., Bergman, N., Rotmans, J., Schilperoord, M. & Köhler, J. (2008). A conceptual framework for transition modelling, *International Journal of Innovation and Sustainable Development* 3(1-2): 93–114.

Herrmann, T. & Loser, K.-U. (1999). Vagueness in models of socio-technical systems, *Behaviour & Information Technology* 18(5): 313–323.

Hines, P. & Talukdar, S. (2007). Controlling cascading failures with cooperative autonomous agents, *International Journal of Critical Infrastructures* 3(1/2): 192–220.

Hodge, B.-M., Aydogan-Cremaschi, S., Blau, G., Pekny, J. & Reklaitis, G. (2008). A prototype agent-based modeling approach for energy systems analysis, *in* Braunschweig & Joulia (2008), pp. 1071–1076.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*, University of Michigan Press, Ann Arbor.

Houwing, M., Heijnen, P. & Bouwmans, I. (2007). Socio-technical complexity in energy infrastructures — conceptual framework to study the impact of domestic level energy generation storage and exchange, *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, Taipei, Taiwan, pp. 906–911.

Huang, M., Tong, Z. S., Zhu, Y. L. & Ma, Y. G. (2004). Research on agent and xml based power plant information integrating system, *Power System Technology* 28(23): 66–70.

Huang, W., Huang, H. H., Yang, T. W. & Chen, W. R. (2005). Research on multi-agent based remote distributed substation video monitoring system, *Dianli Zidonghua Shebei / Electric Power Automation Equipment* 25(4): 5.

Hughes, T. P. (1987). The evolution of large technological systems, *in* W. E. Bijker, T. Hughes & T. J. Pinch (eds), *The social construction of technological systems. New directions in the sociology and history of technology*, MIT Press, Cambridge, pp. 51–82.

Iivari, J. & Hirschheim, R. (1996). Analyzing information systems development: a comparison and analysis of eight is development approaches, *Information Systems* **21**(7): 551–575.

Jackson, J. (2007). Are US utility standby rates inhibiting diffusion of customer-owned generating systems?, *Energy Policy* **35**(3): 1896–1908.

Jarman, A. & Kouzmin, A. (1990). Decision pathways from crisis — a contingency-theory simulation heuristic for the Challenger shuttle disaster (1983–1988), *Contemporary Crises* **14**(4): 399–433.

Jenkins, D. P., Stanton, N. A., Salmon, P. M., Walker, G. H. & Young, M. S. (2008). Using cognitive work analysis to explore activity allocation within military domains, *Ergonomics* **51**(6): 798–815.

Jennings, N. R. (2000). On agent based software engineering, *Artificial Intelligence* **117**(2): 277–296.

Jiang, J. J., Kang, C. Q. & Xia, Q. (2005a). Bid acceptance probability and learning model of genco agent in electricity market simulation, *Power System Technology* **29**(13): 26–31+39.

Jiang, J. J., Kang, C. Q. & Xia, Q. (2005b). A new belief-based bid decision model of genco agent in electricity market simulation, *Power System Technology* **29**(14): 4–9.

Jo, K. Y. & Dockery, J. T. (1998). Virtual network representations of information warfare battlespace, *Proceedings of the IEEE Military Communications Conference MILCOM 98*, Vol. 2, pp. 457–461.

Johnson, J. (2008). Science and policy in designing complex futures, *Futures* **40**(6): 521–537.

Julka, N., Karimi, I. & Srinivasan, R. (2002). Agent-based supply chain management — 2: a refinery application, *Computers and Chemical Engineering* **26**(12): 1771–1781.

Julka, N., Srinivasan, R. & Karimi, I. (2002). Agent-based supply chain management — 1: Framework, *Computers and Chemical Engineering* **26**(12): 1755–1769.

Kaelbling, L. P. (1991). A situated-automata approach to the design of embedded agents, *SIGART Bulletin* **2**(4): 85–88.

Katare, S. & Venkatasubramanian, V. (2001). An agent-based learning framework for modeling microbial growth, *Journal of Engineering Applications of Artficial Intelligence* **14**(6): 715–726.

Keirstead, J. & Leach, M. (2008). Bridging the gaps between theory and practice: a service niche approach to urban sustainability indicators, *Sustainable Development* **16**: 329–340.

Keirstead, J., Samsatli, N. & Shah, N. (2009). SynCity: an integrated tool kit for urban energy systems modeling, *Proceedings of the 5th Urban Research Symposium*, Marseille, France.

Kember, P. & Murray, H. (1988). Towards socio-technical prototyping of work systems, *International Journal of Production Research* **26**(1): 133–142.

Klatt, K. U. & Marquardt, W. (2009). Perspectives for process systems engineering — personal views from academia and industry, *Computers and Chemical Engineering* 33(3): 536–550.

Kling, R., McKim, G. & King, A. (2003). A bit more to it: scholarly communication forums as socio-technical interaction networks, *Journal of the American Society for Information Science and Technology* 54(1): 47–67.

Kolan, P. & Dantu, R. (2007). Socio-technical defense against voice spamming, *ACM Transactions on Autonomous and Adaptive Systems* 2(1): Article No. 2.

Koo, L. Y., Adhitya, A., Srinivasan, R. & Karimi, I. A. (2008). Decision support for integrated refinery supply chains — 2. design and operation, *Computers and Chemical Engineering* 32(11): 2787–2800.

Krause, T., Beck, E. V., Cherkaoui, R., Germond, A., Andersson, G. & Ernst, D. (2006). A comparison of nash equilibria analysis and agent-based modelling for power markets, *International Journal of Electrical Power and Energy Systems* 28(9 SPEC. ISS.): 599–607.

Lekov, A., Lutz, J., Whitehead, C. D. & McMahon, J. E. (2000). Payback analysis of design options for residential water heaters, *Proceedings of the ACEEE Summer Study on Energy Efficiency in Buildings* 1: 1163–1174.

Leonard, A. P. (1992). Managing change with VSM and STS: an experiential approach, *Transactions of the Institute of Measurement and Control* 14(1): 46–50.

LeRouge, C., Hevner, A., Collins, R., Garfield, M. & Law, D. (2004). Telemedicine encounter quality: comparing patient and provider perspectives of a socio-technical system, *Proceedings of the 37th Hawaii International Conference on System Sciences*.

Li, H., Sun, J. & Tesfatsion, L. (2008). Dynamic LMP response under alternative price-cap and price-sensitive demand scenarios, *Proceedings of the IEEE Power and Energy Society 2008 General Meeting: Conversion and Delivery of Electrical Energy in the 21st Century, PES2008*.

Lincoln, R. W., Galloway, S., Burt, G. M. & McDonald, J. R. (2006). Agent-based simulation of short-term energy markets for highly distributed power systems, *41st International Universities Power Engineering Conference, UPEC 2006, Conference Procedings*, Vol. 1, pp. 198–202.

Little, R. G. (2005). Organizational culture and the performance of critical infrastructure: modeling and simulation in socio-technological systems, *Proceedings of the Annual Hawaii International Conference on System Sciences*, p. 63.

Little, R. G., Birkland, T. A., Wallace, W. A. & Herabat, P. (2007). Socio-technological systems integration to support tsunami warning and evacuation, *Proceedings of the Annual Hawaii International Conference on System Sciences*.

Liu, D., Chen, Y., Shen, G. & Fan, Y. (2005). A multi-agent based approach for modeling and simulation of bulk power system restoration, *Proceedings of the IEEE Power Engineering Society Transmission and Distribution Conference*, Vol. 2005, pp. 1–6.

Liu, J., Yoshikawa, H. & Zhou, Y. (2005). Study of visualized simulation and analysis of nuclear fuel cycle system based on multilevel flow model, *Nuclear Science and Techniques/Hewuli* **16**(6): 358–370.

Liu, M. Z., Yang, L. & Gan, D. Q. (2005). A survey on agent based electricity market simulation, *Power System Technology* **29**(4): 76–80.

Lu, L., Murray-Smith, D. J. & Thomson, D. G. (2008). Issues of numerical accuracy and stability in inverse simulation, *Simulation Modelling Practice and Theory* **16**: 1350–1364.

Lu, S. C. & Cai, J. (2000). STARS: A socio-technical framework for integrating design knowledge over the internet, *IEEE Internet Computing* 4(5): 54–62.

Luck, M., McBurney, P., Shehory, O. & Willmott, S. (2005). *Agent Technology: Computing as Interaction (A Roadmap for Agent Based Computing)*, AgentLink.

Lukszo, Z., van Dam, K. H., Weijnen, M. P. C. & Dijkema, G. P. J. (2008). Agent-based models for crisis management, *in* H. Bouwman, R. Bons, M. Hoogewegen, M. Janssen & H. Pronk (eds), *Let a Thousand Flowers Bloom, Essays in commemoration of prof.dr. René Wagenaar*, IOS Press, pp. 281–299. ISBN 978-1-58603-879-3.

Ma, C., Feng, J. Q., Yang, Z., Wu, Q. H. & Fitch, J. (2007). A broker agent for remote control of distributed power systems, *Proceedings of the 2006 International Conference on Power System Technology, POWERCON2006*.

Macal, C. M. & North, M. J. (2002). Beer, bullwhips, and agents. The Supply Chain Simulation Workgroup — Santa Fe Institute Business Network.

Macal, C. M. & North, M. J. (2005). Tutorial on agent-based modelling and simulation, *in* M. Euhl, N. Steiger, F. Armstrong & J. Joines (eds), *Proceedings of the 2005 Winter Simulation Conference*, Orlando, USA.

Maciol, A. & Stawowy, A. (1993). Intelligent modeling of decision-making process in information systems, *IFIP Transactions A: Computer Science and Technology* (A-36): 157–163.

Maes, P. (1991). The agent network architecture (ANA), *SIGART Bulletin* 2(4): 115–120.

Maiden, N., Ncube, C., Kamali, S., Seyff, N. & Grünbacher, P. (2007). Exploring scenario forms and ways of use to discover requirements on airports that minimize environmental impact, *Proceedings of the 15th IEEE International conference on Requirements Engineering RE '07*, pp. 29–38.

Masys, A. J. (2007). Understanding climate change through modelling and simulation: a case for verification, validation and accreditation, *Proceedings of the 2006 IEEE EIC Climate Change Technology Conference, EICCCC 2006*.

MathWorks (1996). Matlab, using Simulink version 2.

MathWorks (2008). Matlab 7 — programming fundamentals.

McCown, R. L. (2002). Locating agricultural decision support systems in the troubled past and socio-technical complexity of 'models for management', *Agricultural Systems* 74(1): 11–25.

McIntosh, B. S., Jeffrey, P., Lemon, M. & Winder, N. (2005). On the design of computer-based models for integrated environmental science, *Environmental Management* 35(6): 741–752.

McNally, P. & Heavey, C. (2004). Developing simulation as a desktop resource, *International Journal of Computer Integrated Manufacturing* 17(5): 435–450.

McNeese, M. D., Perusich, K. & Rentsch, J. R. (2000). Advancing socio-technical systems design via the living laboratory, *Proceedings of the XIVth Triennial Congress of the International Ergonomics Association and 44th Annual Meeting of the Human Factors and Ergonomics Association, 'Ergonomics for the New Millennium'*, pp. 610–613.

Mele, F. D., Guillen, G., Espuna, A. & Puigjaner, L. (2005). An agent-based approach for supply chain retrofitting under uncertainty, *Computers and Chemical Engineering* 31(5–6): 722–735.

Meyrick and Associates (2006). National intermodal terminal study. http://www.infrastructure.gov.au/transport/publications/pdf/10781_MA_DOTARS_NIMTS_Final_report_13020_Final.pdf.

Mikulecky, D. C. (2001). The emergence of complexity: science coming of age or science growing old?, *Computers and Chemistry* 25(4): 341–348.

Monch, L. (2007). Simulation-based benchmarking of production control schemes for complex manufacturing systems, *Control Engineering Practice* 17: 1381–1393.

Montgomery, D. C. (2008). *Design and analysis of experiments*, 7th edn, John Wiley and Son. ISBN 9780470128664.

Morais, H., Cardoso, M., Khodr, H., Praça, I. & Vale, Z. (2008). Virtual power producers market strategies, *2008 5th International Conference on the European Electricity Market, EEM*.

Moscoso, P. G., Wäfler, T. & Windischer, A. (1999). A modelling framework for complementary design of production planning and control systems, *Ergonomics* 42(11): 1492–1506.

Müller, M., Sensfuß, F. & Wietschel, M. (2007). Simulation of current pricing-tendencies in the german electricity market for private consumption, *Energy Policy* 35(8): 4283–4294.

Navarre, D., Palanque, P., Dragicevic, P. & Bastide, R. (2006). An approach integrating two complementary model based environments for the construction of multimodal interactive applications, *Interacting with Computers* 18(5): 910–941.

Negenborn, R. R. (2007). *Multi-agent model predictive control with applications to power networks*, PhD thesis, Delft University of Technology, the Netherlands. ISBN 978-90-5584-093-9.

Negenborn, R. R. & De Schutter, B. (2008). A distributed model predictive control approach for the control of irrigation canals, *Proceedings of the International Conference on Infrastructure Systems 2008 (INFRA 2008)*, Rotterdam, The Netherlands.

Negenborn, R. R., van Overloop, P. J., Keviczky, T. & De Schutter, B. (2009). Distributed model predictive control for irrigation canals, *Networks and Heterogeneous Media* 4(2): 359–380.

Nelder, J. A. & Mead, R. (1965). A simplex method for function minimization, *Computer Journal* 7(4): 308–313.

Nikitaev, V. V. (1991). Semiotic approach to simulation, *Automation and Remote Control* 52(5 pt 2): 697–704.

Nikolai, C. & Madey, G. (2009). Tools of the trade: a survey of various agent based modeling platforms, *Journal of Artificial Societies and Social Simulation (JASSS)* 12(2 2).

Nikolic, I. (2009). *Socially inclusive, open source evolutionary method for modelling Large Scale Socio-Technical system evolution and several models that demonstrate the method*, PhD thesis, Delft University of Technology, the Netherlands. to be defended.

Nikolic, I., Dijkema, G. P. J. & van Dam, K. H. (2009). Understanding and shaping the evolution of sustainable large-scale socio-technical systems — towards a framework for action oriented industrial ecology, *in* M. Ruth & B. Davidsdottir (eds), *The Dynamics of Regions and Networks in Industrial Ecosystems*, Edward Elgar, pp. 156–178. ISBN 978 1 84720 742 5.

Nikolic, I., Dijkema, G. P. J., van Dam, K. H. & Lukszo, Z. (2006). General methodology for action-oriented industrial ecology: complex systems approach applied to the rotterdam industrial cluster, *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, Ft. Lauderdale, Florida, USA, pp. 831–836.

Noble, T. (2000). *Social theory and social change*, Palgrave Macmillan.

North, M., Collier, N. & Vos, J. (2006). Experiences creating three implementations of the Repast agent modeling toolkit, *ACM Transactions on Modeling and Computer Simulation* 16(1): 1–25. ACM, New York, USA.

Noy, N. F. & McGuinness, D. L. (2001). Ontology development 101: a guide to creating your first ontology, *Technical Report SMI-2001-0880*, SMI. http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html.

Nuutinen, M., Savioja, P. & Sonninen, S. (2007). Challenges of developing the complex socio-technical system: realising the present, acknowledging the past, and envisaging the future of vessel traffic services, *Applied Ergonomics* 38(5): 513–524.

Oh, H. & Thomas, R. (2007). A diffusion-model-based supply-side offer agent, *Proceedings of the 2007 IEEE Power Engineering Society General Meeting, PES2007*.

Ortega-Vazquez, M. A. & Kirschen, D. S. (2008). Assessment of generation expansion mechanisms using multi-agent systems, *Proceedings of the IEEE Power and Energy Society 2008 General Meeting: Conversion and Delivery of Electrical Energy in the 21st Century, PES2008*, Baltimore, Maryland, USA, pp. 1–7.

Ottens, M., Franssen, M., Kroes, P. & van de Poel, I. (2006). Modelling infrastructures as socio-technical systems, *International Journal of Critical Infrastructures* 2(2/3): 133–145.

Ottens, M. & Marchau, V. (2005). ITS: Beyond designing the technology, *Proceedings of the IEEE Conference on Intelligent Transportation Systems ITSC 2005*, Vol. 2005, pp. 1033–1038.

Ouria, A. & Toufigh, M. M. (2009). Application of Nelder-Mead simplex method for unconfined seepage problems, *Applied Mathematical Modelling* 33: 3589–3598.

Oxford English Dictionary Online (2009). Oxford University Press. http://dictionary.oed.com/.

Parunak, H. V. D., Savit, R. & Riolo, R. L. (1998). Agent-based modeling vs. equation-based modeling: a case study and users' guide, *Proceedings of the Workshop on Modelling Agent Based Systems MABS98*, Paris, France.

Pasmore, W. A. & Khalsa, G. S. (1993). The contributions of Eric Trist to the social engagement of social science, *The Academy of Management Review* 18(3): 546–569.

Peerenboom, J. (2001). Energy infrastructure interdependencies program, *Proceedings of the IEEE Power Engineering Society Transmission and Distribution Conference*, Vol. 1, p. 194.

Pitty, S. S., Li, W., Adhitya, A., Srinivasan, R. & Karimi, I. (2008). Decision support for integrated refinery supply chains — 1. dynamic simulation, *Computers and Chemical Engineering* 32(11): 2767–2786.

Proctor, M., Neale, M., Frandsen, M., Griffith, Jr., S., Tirelli, E., Meyer, F. & Verlaenen, K. (2008). Drools — JBoss rules user guide, *Technical report*, JBoss. http://www.jboss.org/drools/documentation.html.

Qudrat-Ullah, H. (2008). Structural validation of simulation models: an illustration, *Proceedings of the UKSim 10th International Conference on Computer Modelling and Simulation, EUROSIM/UKSim2008*, pp. 537–542.

Ramanna, S., Skowron, A. & Peters, J. F. (2007). *Approximation space-based socio-technical conflict model*, Vol. 4481, Springer, Berlin / Heidelberg.

Ramaswamy, V., Thulasidasan, S., Romero, P., Eidenbenz, S. & Cuéllar, L. (2007). Simulating the national telephone network: a socio-technical approach to assessing infrastructure criticality, *Proceedings of the IEEE Military Communications Conference MILCOM*.

Rammelt, C. F., Nikolic, I., Boes, J. & van Dam, K. H. (2005). Complex systems approach to 'development aid', *Proceedings of the 11th Annual International Sustainable Development Research Conference*, Helsinki, Finland.

Rosenkranz, C. & Holten, R. (2007). Combining cybernetics and conceptual modeling: The concept of variety in organizational engineering, *Proceedings of the ACM Symposium on Applied Computing* pp. 1228–1233.

Saeed, K. (1987). A re-evaluation of the effort to alleviate poverty and hunger, *Socio-Economic Planning Sciences* 21(5): 291–304.

Saguan, M., Keseric, N., Glachant, J. M., Bastard, P. & Haas, R. (2005). The effects of transmission capacities on market outcomes: an agent-based approach, *Proceedings of the 2005 IEEE Russia Power Tech, PowerTech*.

Savoyant, A. & Ladure, P. (1972). A method of approach to safety problems in a socio technical system, *Travail Humain* 35(2): 219–232.

Scheidt, D. H. & Pekala, M. J. (2007). Model-based agents, *Proceedings of the IEEE Power Engineering Society General Meeting, PES2007*.

Shah, A. P. & Pritchett, A. R. (2005). *Work-environment analysis: environment centric multi-agent simulation for design of socio-technical systems*, Vol. 3415, Springer, Berlin / Heidelberg.

Sheng, G. H., Jiang, X. C., Fan, X. H. & Zeng, Y. (2005). Implementation and performance analysis of multi-agent based secondary voltage control system, *Implementation and performance analysis of multi-agent based secondary voltage control system* 29(24): 42–48.

Shin, D. (2004). Public networks for SME applications: implications for future SME networks, *Human Systems Management* 23(3): 151–159.

Shin, J. E., Sutcliffe, A. G. & Gregoriades, A. (2005). Scenario advisor tool for requirements engineering, *Requirements Engineering* 10(2): 132–145.

Siirola, J. D., Hauan, S. & Westerberg, A. W. (2003). Toward agent-based process systems engineering: proposed framework and application to non-convex optimization, *Computers and Chemical Engineering* 27(12): 1801–1811.

Siirola, J. D., Hauan, S. & Westerberg, A. W. (2004). Computing pareto fronts using distributed agents, *Computers & Chemical Engineering* 29(1): 113–126.

Simon, C. (2006). Presentation at the complexity in industrial ecology workshop, University of Michigan, Ann Arbor.

Simon, H. A. (1962). The architecture of complexity, *Proceedings of the American Philosophical Society*, Vol. 106, pp. 467–482.

Simone, C. (1989). Identification and estimation of parameters for a computer-based organization, *IFAC Proceedings Series*, Vol. 2, pp. 873–878.

Sims, B. & Henke, C. (2007). Stockpile stewardship and the reliable replacement warhead: Socio-technical repair in the U.S. nuclear weapons complex, *IEEE International Symposium on Technology and Society ISTAS 2007*, pp. 1–6.

Sirikijpanichkul, A., L. F. (2006). Modelling intermodal freight hub location decisions, *Proceedings of the 2006 IEEE International Conference on Systems, Man, and Cybernetics*, The Grand Hotel, Taipei, Taiwan, pp. 890–895.

Sirikijpanichkul, A., van Dam, K. H., Ferreira, L. & Lukszo, Z. (2007). Optimizing the location of intermodal freight hubs: an overview of the agent based modelling approach, *Journal of Transportation Systems Engineering and Information Technology* 7(4): 71–81.

Smajgl, A., Izquierdo, L. R. & Huigen, M. (2008). Modeling endogenous rule changes in an institutional context: the adico sequence, *Advances in Complex Systems* 11(2): 199–215.

Sterman, J. (1989). Testing behavioral simulation models by direct experiment, *Management Science* 33(12): 1572–1592.

Sterman, J. (2000). *Business dynamics. Systems thinking and modeling for a complex world*, Irwin McGraw-Hill, Boston.

Sun, J. & Tesfatsion, L. (2007). An agent-based computational laboratory for wholesale power market design, *Proceedings of the IEEE Power Engineering Society General Meeting, PES2007*.

Sure, Y., Erdmann, M., Angele, J., Staab, S., Studer, R. & Wenke, D. (2001). OntoEdit: Collaborative ontology engineering for the semantic web, *Proceedings of the first International Semantic Web Conference 2002 (ISWC 2002)*, Vol. 2342 of *Lecture Notes in Computer Science*, Springer.

Sutcliffe, A., Chang, W. & Neville, R. S. (2007). Applying evolutionary computing to complex systems design, *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans* 37(5): 770–779.

Tàbara, D. J., Roca, E., Madrid, C., Valkering, P., Wallman, P. & Weaver, P. M. (2008). Integrated sustainability assessment of water systems: lessons from the Ebro river basin, *International Journal of Innovation and Sustainable Development* 3(1-2): 48–69.

Tàbara, J. D., Elmqvist, B., Ilhan, A., Madrid, C., Olson, L., Schilperoord, M., Valkering, P., Wallman, P. & Weaver, P. (2007). Participatory modelling for the integrated sustainability assessment of water: the world cellular model and the MATISSE project, *in* J. Jäger & P. M. Weaver (eds), *MATISSE Working Papers*, 9, http://www.matisse-project.net/.

Tang, Y., Parsons, S. & Sklar, E. (2006). Modeling human education data: From equation-based modeling to agent-based modeling, *in* L. Antunes & K. Takadama (eds), *MABS 2006, LNAI 4442*, Springer-Verlag Berlin Heidelberg, pp. 41–56.

Tellidou, A. C. & Bakirtzis, A. G. (2007). Agent-based analysis of monopoly power in electricity markets, *Proceedings of the 2007 International Conference on Intelligent Systems Applications to Power Systems, ISAP*, pp. 1–6.

Terna, P. (1998). Simulation tools for social scientists: building agent based models with SWARM, *Journal of Artificial Societies and Social Simulation (JASSS)* 1(2).

Thimmapuram, P., Veselka, T. D., Koritarov, V., Vilela, S., Pereira, R. & Silva, R. F. (2008). Modeling hydro power plants in deregulated electricity markets: integration and application of EMCAS and VALORAGUA, *2008 5th International Conference on the European Electricity Market, EEM*.

Thissen, W. A. & Herder, P. M. (2003). Critical infrastructures: Challenges for systems engineering, *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 2042–2047.

Tong, X., Wang, X. & Ding, L. (2008). Study of information model for wide-area backup protection agent in substation based on IEC61850, *3rd International Conference on Deregulation and Restructuring and Power Technologies, DRPT 2008*, pp. 2212–2216.

Trigo, P. & Marques, P. (2008). The electricity market as a multi-agent system, *Proceedings of the 5th International Conference on the European Electricity Market, EEM*.

Trist, E. L. (1981). The sociotechnical perspective: the evolution of sociotechnical systems as a conceptual framework and as an action research program, *in* A. H. van de Ven & W. F. Joyce (eds), *Perspectives on organization design and behavior: 19-75*, Wiley, New York.

Tsoukalas, L. H. (2001). Modeling the grid as a customer-driven system: emerging challenges and opportunities for self-healing infrastructures, *Proceedings of the IEEE Power Engineering Society Transmission and Distribution Conference*, Vol. 1, p. 158.

van Dam, K. H. (2002). *Design of a generic ontology translator — implemented for JADE*, Master's thesis, Vrije Universiteit, Amsterdam, the Netherlands.

van Dam, K. H., Adhitya, A., Srinivasan, R. & Lukszo, Z. (2008a). Benchmarking numerical and agent-based models of an oil refinery supply chain, *in* Braunschweig & Joulia (2008), pp. 623–628.

van Dam, K. H., Adhitya, A., Srinivasan, R. & Lukszo, Z. (2008b). Nuances of benchmarking agent-based and equation-based models of an oil refinery supply chain, *Proceedings of the First International Conference on Infrastructure Systems (INFRA 2008): Building Networks for a Brighter Future*, Rotterdam, The Netherlands.

van Dam, K. H., Adhitya, A., Srinivasan, R. & Lukszo, Z. (2009). Critical evaluation of paradigms for modelling integrated supply chains, *Journal of Computers and Chemical Engineering* 33(10): 1711–1726. doi:10.1016/j.compchemeng.2009.01.023.

van Dam, K. H. & Lukszo, Z. (2006). Modelling energy and transport infrastructures as a multi-agent system using a generic ontology, *Proceedings of the 2006 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, The Grand Hotel, Taipei, Taiwan, pp. 890–895.

van Dam, K. H. & Lukszo, Z. (2009). Model factory for socio-technical infrastructure systems, *in* R. R. Negenborn, Z. Lukszo & J. Hellendoorn (eds), *Intelligent Infrastructures*, Intelligent Systems, Control and Automation: Science and Engineering, Springer. in print.

van Dam, K. H., Lukszo, Z. & Srinivasan, R. (2009). Agent-based decision support for failure-prone networked infrastructures, *International Journal of Critical Infrastructures* 9. in print.

van Dam, K. H., Nikolic, I., Lukszo, Z. & Dijkema, G. P. J. (2006). Towards a generic approach for analyzing the efficiency of complex networks, *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, Ft. Lauderdale, Florida, USA.

van Dam, K. H., Verwater-Lukszo, Z., Ottjes, J. A. & Lodewijks, G. (2006). Distributed intelligence in autonomous multi-vehicle systems, *International Journal of Critical Infrastructures* 2(2/3): 261–272.

van Oosterhout, E. M. W., Talmon, J. L., De Clercq, P. A., Schouten, H. C., Tange, H. J. & Hasman, A. (2005). Three-layer model for the design of a protocol support system, *International Journal of Medical Informatics* 74(2-4): 101–110.

Vartanian, C., Law, D., Hines, P., Yinger, R., Hamilton, S. & Feliachi, A. (2008). Agent-based distribution circuit automation, *Proceedings of the IEEE Power and Energy Society 2008 General Meeting: Conversion and Delivery of Electrical Energy in the 21st Century, PES2008*.

Verwater-Lukszo, Z. (1996). *FRIS — A practical approach to recipe improvement and optimization in the batch processing industry*, PhD thesis, Eindhoven University of Technology.

Waldorp, M. (1992). *Complexity: the emerging science at the edge of order and chaos*, Simon and Schuster, New York.

Wang, H. H., Noy, N., Rector, A., Musen, M., Redmond, T., Rubin, D., Tu, S., Tudorache, T., Drummond, N., Horridge, M. & Seidenberg, J. (2006). Frames and OWL side by side, *9th International Protégé Conference*, Stanford, California, USA.

Wang, J., Botterud, A., Conzelmann, G. & Korintarov, V. S. (2008). Market power analysis in the EEX electricity market: an agent-based simulation approach, *Proceedings of the IEEE Power and Energy Society 2008 General Meeting: Conversion and Delivery of Electrical Energy in the 21st Century, PES2008*.

Waupotitisch, R., Eidenbenz, S., Smith, J. & Kroc, L. (2006). Multi-scale integrated information and telecommunication systems (MIITS): first results from a large-scale end-to-end network simulator, *Proceedings of the Winter Simulation Conference 2006*, pp. 2132–2139.

Weaver, P. M. & Jordan, A. (2008). What roles are there for sustainability assessment in the policy process?, *International Journal of Innovation and Sustainable Development* 3(1-2): 9–32.

Weidlich, A. & Veit, D. (2008a). Analyzing interrelated markets in the electricity sector — the case of wholesale power trading in Germany, *Proceedings of the IEEE Power and Energy Society 2008 General Meeting: Conversion and Delivery of Electrical Energy in the 21st Century, PES2008*.

Weidlich, A. & Veit, D. (2008b). A critical survey of agent-based wholesale electricity market models, *Energy Economics* **30**(4): 1728–1759.

Weijnen, M. P. C. & Bouwmans, I. (2006). Innovation in networked infrastructures: coping with complexity, *International Journal of Critical Infrastructures* 2(2/3): 121–132.

Weiss, G. (1999). *Multi-agent systems — a modern approach to distributed Artificial Intelligence*, MIT Press, Cambridge.

Whitmarsh, L. & Nykvist, B. (2008). Integrated sustainability assessment of mobility transitions: simulating stakeholders' visions of and pathways to sustainable land-based mobility, *International Journal of Innovation and Sustainable Development* 3(1-2): 115–127.

Wong, C. W. K., Adhitya, A. & Srinivasan, R. (2008). Dynamic modeling of a global specialty chemicals supply chain, *Proceedings of the AIChE Annual Meeting 2008*, Philadelphia, PA. paper 448b.

Wooldridge, M. (2009). *An Introduction to Multi-agent Systems*, 2nd edn, John Wiley & Sons, Chichester, England. ISBN 0470519460.

Wooldridge, M. J. & Jennings, N. R. (1995). Intelligent agents: theory and practice, *Knowledge Engineering Review* **10**(2): 115–152.

Wright, M. H. (1995). Direct search methods: once scorned, now respectable, *in* D. F. Griffiths & G. A. Watson (eds), *Numerical analysis*, Addison-Wesley, Longman, Harlow, UK.

Xu, M., Wu, J., Hu, Z. & Zhang, Z. (2008). Multi-agent based multi-objective network planning in power market environment, *Dianli Zidonghua Shebei / Electric Power Automation Equipment* **28**(1): 33–37.

Yahja, A. & Carley, K. M. (2005). WIZER: An automated intelligent tool for model improvement of multi-aegent social-network systems, *Proceedings of the Eighteenth International Florida Artificial Intelligence Research Society Conference, FLAIRS 2005 - Recent Advances in Artifical Intelligence*, pp. 44–49.

Ydstie, B. E. (2004). Distributed decision making in complex organizations: the adaptive enterprise, *Computers and Chemical Engineering* **29**(1): 11–27.

Ye, T. & Xiong, C. H. (2008). Geometric parameter optimization in multi-axis machining, *Computer-Aided Design* **40**: 879–890.

Ye, Y. & Fischer, G. (2007). *Designing for participation in socio-technical software systems*, Vol. 4554 - part 1 of *Lecture Notes in Computer Science*, Springer, Berlin / Heidelberg.

Yilmaz, L. (2007). *Modelling software processes as human-centered adaptive work systems*, Vol. 4764 of *Lecture Notes in Computer Science*, Springer, Berlin / Heidelberg.

Yu, N. & Liu, C. C. (2008). Multi-agent systems and electricity markets: state-of-the-art and the future, *Proceedings of the IEEE Power and Energy Society 2008 General Meeting: Conversion and Delivery of Electrical Energy in the 21st Century, PES2008*.

Yu, T. T., Scanlan, J. P. & Wills, G. B. (2007). Agent-based and discrete-event modeling: a quantitative comparison, *Collection of Technical Papers — 7th AIAA Aviation Technology, Integration, and Operations Conference*, Vol. 2, pp. 1284–1295.

Yuan, J. H., Ding, W. & Hu, Z. G. (2005). A critical study of agent based computational economics and its application in research of electricity market theory, *Power System Technology* **29**(7): 47–51.

Zandi, M. (2008). *Financial Shock: a 360° Look at the Subprime Mortgage Implosion, and How to Avoid the Next Financial Crisis*, Financial Times Press. ISBN 0137142900.

Zarboutis, N. & Marmaras, N. (2007). Design of formative evacuation plans using agent-based simulation, *Safety Science* **45**(9): 920–940.

Zee, D. J. v. d. (2006). Modeling decision making and control in manufacturing simulation, *International Journal of Production Economics* **100**(1): 155–167.

Glossary

$\bar{d}$	Disturbances.
$\bar{x}$	Degrees of freedom.
μCHP	Micro-Combined Heat and Power.
ABM	Agent-based model(ling).
AI	Artificial Intelligence.
BDI	Belief, Desire and Intention.
CAS	Chemical Abstracts Service number.
CCS	Carbon Capture and Storage.
CDU	Crude Distillation Unit.
CVS	Concurrent Versions System.
EMCAS	Electricity Market Complex Adaptive System.
ETCS	European Train Control System.
GIS	Geographical Information System.
GUI	Graphical User Interface.
IDE	Integrated Development Environment.
kbbl	A volume equivalent to a thousand oil barrels of approximately 160 liters each.
MATISSE	Methods and Tools for Integrated Sustainability Assessment.
Model E	Model of the refinery supply chain developed in Microsoft Excel.
Model M	Model of the refinery supply chain developed in MATLAB/Simulink.

Model R	Model of the refinery supply chain developed in Repast.
OWL	Web Ontology Language.
PPP	Private Public Partnership.
RDFS	Resource Description Framework Schema.
SVN	Subversion.
VLCC	Very Large Crude Carrier.
W3C	The World Wide Web Consortium.
XML	Extensible Markup Language.

Summary

Capturing socio-technical systems with agent-based modelling

What is a suitable modelling approach for socio-technical systems? The answer to this question is of great importance to strategic decision makers in large scale interconnected network systems. Typical examples are the regional, national, continental and global networks found in the public utility sectors and network industries which provide, for example, energy, telecommunication and transportation services. The behaviour of such systems is determined by many actors including regulators, asset owners, operators, service providers and users. Each decision making entity is situated in a dynamic, multi-actor, multi-objective and multi-level jungle: it is part of a bigger system which is constantly changing, it has to cope with the actions of other actors who may have conflicting interests and values, and who operate on different levels of hierarchy. Which models could support such an actor to explore different scenarios and to learn about the possible consequences of different actions through simulations? Successful models should be able to capture both the physical and social reality of the system, their interactions with one another and the external dynamic environment, and they must allow users to experiment with *changes* in both the physical and the social network configuration. In other words, socio-technical systems pose a formidable challenge for modellers.

Existing tools to deal with either the physical (e.g. models of industrial processes) or the social network (e.g. economic market models) are available, but these worlds have yet to be brought together in an integrated modelling approach for socio-technical systems. That is the ambition of this thesis. The additional challenge is to meet this objective not just for one specific domain, such as energy or industry, but to set up a modelling infrastructure that is able to deal with today's reality of socio-technical network systems that are interconnected across domains. This thesis aims at contributing to an integrated framework for socio-technical systems to help modellers build better models and ultimately provide better decision support to actors involved in regulating, operating or otherwise using these systems.

This thesis covers two different story lines, which will be addressed below. The first starts with an illustration of the problems and challenges in socio-technical systems and the need for a flexible bottom-up approach to modelling, resulting in a modelling framework that fulfils these criteria. The framework can then be applied to a number of case studies, with each subsequent case study contributing to the generic framework. For this purpose the *agent-based* paradigm turned out as most promising. This story line could be

denoted as the 'framework for agent-based models of socio-technical systems'.

The second story line starts with the need of modellers to justify the choice of the selected modelling paradigm, as well as with the scientific challenge to objectively analyse the framework developed in this thesis. After a methodology for systematically performing such a comparison is given, a benchmarking exercise of modelling paradigms is done on a number of case studies. The evaluation of the framework results in rules of thumb for the applicability and its usefulness. Two models developed with the framework are then deployed to support a problem owner, demonstrating how real-life decision problems can be solved with agent-based models. This second story line could be labelled the 'critical evaluation of agent-based models of socio-technical systems'.

Framework for agent-based models of socio-technical systems

To deal with the challenges that arise from socio-technical complexity a generic agent-based modelling framework has been developed (Chapter 3). This framework aims at supporting the modeller in quickly setting up new applications by re-using building blocks and allowing the connection of existing models to one another. The framework consists of the following three types of elements:

- **Interface** definition between components, between models, between developers and between developers and problem-owners.
- **Library** of source code that can be re-used.
- **Procedures** on how to use the library and interface to build models.

In an agent-based model the system is described in terms of agents and their behaviour, where an agent is a model of a decision making entity at various levels of aggregation, from an individual to a collective. Agents are considered as software entities that are autonomous, re-active, pro-active and capable of social behaviour. The agent-based paradigm is particularly suitable to model socio-technical systems because it allows the modeller to describe the social elements of the system through algorithms. Furthermore, it offers a flexible bottom-up approach which is needed for performing experiments with changing elements (leading to different configurations of the system) to study the effect on overall system behaviour.

The cornerstone of the framework is a shared language formalised in an *ontology*, which is a formal specification of concepts. The ontology forms the interface needed to bring different aspects of the system (both social and physical) together and to interconnect different models. Besides interconnectivity, the ontology offers *interoperability*. The ontology provides a set of (abstract) classes and properties with which (concrete) instances can be defined. The instances are the system elements — or the facts — included in the model which are stored in a shared *knowledge base*. Furthermore, the concepts from the ontology (i.e. the words in the shared language) are also used to define the behaviour of the agents. Finally, a shared language also helps find a common ground when communicating with experts and users from different domains.

The modelling infrastructure can help to set up new models of socio-technical systems by following a sequence of modelling steps and, where possible, re-using already existing "building blocks" (e.g. facts, procedures, agents or technologies) from models developed during previous applications. When new elements are created for a specific case they can be fed back into the shared framework with the result that they are available for re-use. A basic set of class definitions for socio-technical systems was developed from a number of initial case studies and refined through subsequent applications.

The approach presented in this thesis has been applied to a number of case studies (Chapter 4). Applications include a model of an intermodal freight hub, an oil refinery supply chain and a chocolate production cluster. With a description of the development of these models, the model-building procedures were demonstrated and it was illustrated how the framework supports modellers. These procedures include the conceptualisation of the system in terms of agents and physical elements, refining the generic ontology for case-specific concepts, the creation of concrete instances and the implementation of agent behaviour. Furthermore, these procedures were demonstrated to be applicable by other modellers to various case-specific problems in a wide range of infrastructure domains.

The development of the framework over time, through application and refinement cycles and with contributions from many users, is well documented and the development trajectory itself has been studied and analysed (Chapter 5). The completeness, correctness and usability of the framework were tested. It is concluded that the ontology is complete for the scope that was defined and that it can be successfully expanded for new problems. After the initial development phase none of the key concepts have been changed or replaced which, together with their widespread use, indicates that the ontology is adequate. Finally, it was demonstrated that through re-use of instances in the shared knowledge base as well as of model source code, building new models for new cases using the framework requires less work. After the initial investment in the first generation of models for a number of initial cases, new applications can be developed more efficiently.

Critical evaluation of agent-based models of socio-technical systems

A critical evaluation of the advantages and disadvantages of the framework and a detailed comparison with other modelling paradigms is called for. Comparing modelling paradigms based only on the conceptual model specifications is not enough; rather a well-defined benchmarking process and experiments are required. By building different models and analysing how they are built and how they can be expanded, a well-founded justification for the choice of modelling paradigm can be made and recommendations and guidelines on which paradigm is more suitable for which problem can be given.

One of the main problems in comparing modelling paradigms lies in the definition of what is encompassed in each of the paradigms in the study (Chapter 2). A distinction between agent-based models and equation-based models found in literature overlooks the fact there is a fuzzy boundary and that both *labels* can be interpreted in different ways. The concept is not black-and-white, rather there is a continuous scale or a spectrum in the modelling space. There are two main axes on which models can differ: The *model elements* axis and the *system description elements* axis. The former deals with *what* is modelled and the constituents of the model, the latter with *how* their structure and behaviour are formally described. The constituents of the model range from individuals (i.e. decision making entities) to system level observables and the system description elements from strictly equations to algorithms. This nuance allows the conclusions of a benchmarking study to be generalised beyond the specific *models* that are compared, to the advantages and shortcomings of *modelling paradigms*.

A general scheme to compare modelling paradigms is proposed (Chapter 6), with special emphasis on the identification of what is to be benchmarked, the evaluation if objects of study are comparable and the description of well-structured experiments. This way fair and balanced conclusions can be drawn. The benchmarking scheme is then used to compare different models of an oil refinery supply chain, developed using different modelling paradigms; one using a numerical tool and the other using an agent-based platform. It was demonstrated that different modelling paradigms and tools can be used to successfully create a model of the *same* socio-technical system with comparable results. By analysing the efforts required to expand the models to allow new scenarios to be tested, the strengths of the paradigms were identified in the context of supply chain modelling. Ease of expressing the problem, ease of extending the models, ease of re-use and ease of explaining the results were used as performance indicators.

The results of the benchmarking study can, within the context in which the comparison was performed, be expanded from the specific models to the modelling paradigms. Production processes and technological aspects are well catered for by equations, while the decision making aspects can only be captured in algorithms. The complete system can, however, be fully expressed in both modelling paradigms that are compared. One can say that equation-based models, in general, are more suitable for representing the physical elements in the system whereas the (dynamic) interaction between the actors is best captured by the agent-based model. For extending or adjusting the models the general rule is that if something is only indirectly captured in the model it requires more effort to be changed. This means that for the agent-based model, where the configuration of the system is dynamic and not fixed in the model structure, adding new actors, new physical elements and, consequently, introducing new possible relationships was easy. However, adjustments in the way the technical system itself works were more easily done through adjusting equations. If a new model is built based on earlier work, for the equation-based model the conceptualisation could be re-used, but none of the actual equations could be copied. From the bottom-up agent-based approach, on the other hand, also specific building blocks could be re-used or extended. Finally, when explaining the model and the model results to stakeholders, the agent-paradigm offers a natural representation of the decision making processes and interactions between the entities in the system, while equations, in front of the right audience, have an edge when explaining the technical processes.

After performing the benchmarking study and learning about the advantages and disadvantages of agent-based modelling, it is demonstrated how simulation models developed with the framework presented in this thesis can support a problem-owner by solving a specific decision problem (Chapter 7). These problems can often be characterised by the fact that they are multi-actor, multi-criteria and multi-level problems. A decision model, formulated for a specific purpose and question, is built for a simulation model of the system and different search strategies can be defined to solve the problem.

To show how agent-based models can be applied as decision support tools two illustrative case studies with the agent-based simulation models of an oil refinery supply chain and an intermodal freight transport system, both inspired by real-life problems, are presented. It is demonstrated how a search strategy from the field of Operations Research, such as the Nelder-Mead optimisation method, can be applied to a decision problem with disturbances within the supply chain to choose the right response to abnormal situations in such a highly complex system. In another case study it is shown how different tax incentives can be used to encourage different stakeholders to agree on the location of a new freight hub. As such, it is demonstrated that agent-based models developed with the generic framework presented in this thesis can support decision makers to solve real-life problems.

Conclusions

The framework developed in this thesis presents a suitable generic modelling approach for socio-technical systems. Agent-based models are particularly appropriate to experiment with different scenarios and to answer *what if* questions. This gives valuable support for decision makers in dealing with, for example, disturbances in the physical system or with new regulations imposed.

The models built and used in this thesis were developed in a bottom-up fashion, making it possible to change the social configuration so new actors can be included in the system (e.g. different users of the freight hub or more suppliers with different prices and lead times in the supply chain) or to adjust the physical configuration (e.g. additional transport links in the intermodal freight network or extra storage tanks for the refinery). The framework was designed from the start to be able to deal with a variety of infrastructures and other socio-technical networks so that lessons learnt in one domain can be translated to other domains and (parts of) models of different infrastructure systems can be connected.

The framework development is an ongoing process through ongoing use; new modellers are using the approach for new cases and as such contribute to the shared framework. This is one of the key strengths of the approach: the more it is used, the more that can be re-used. Hereby the reader is invited to start thinking about challenges in the infrastructure domain from a socio-technical and agent-based perspective and to map the system's elements onto the ontology presented here, so that the modelling infrastructure can be used to effectively build better models and make the modelling process more efficient.

Koen Haziël van Dam

Samenvatting

Grip krijgen op socio-technische systemen met agent-gebaseerd modelleren

Wat is een geschikte modelleeraanpak voor socio-technische systemen? Het antwoord op deze vraag is van groot belang voor strategische beslissers in grootschalige systemen die gekenmerkt worden door technische complexiteit èn een multi-actor karakter. Typische voorbeelden zijn de regionale, nationale, continentale en mondiale netwerken die energie, telecommunicatie- en transportdiensten leveren of industriële productieketens en netwerken. Het gedrag van zulke systemen wordt bepaald door vele actoren, waaronder toezichthouders, eigenaren van fysieke installaties, producenten, dienstverleners en gebruikers. Elke actor opereert in een dynamische omgeving en in interactie met andere actoren die verschillende doelstellingen en waardensystemen kunnen hebben. Hoe verschillend transportinfrastructuren, energie-infrastructuren of industriële netwerken in eerste aanblik ook mogen lijken, op het socio-technische systeemniveau spelen vergelijkbare problemen die voortkomen uit de technische complexiteit van het systeem en de multi-actor complexiteit.

Welke modellen kunnen actoren in dergelijke systemen ondersteunen bij het verkennen van scenario's en hen in staat stellen te experimenteren met alternatieve handelingsstrategieën, door middel van simulaties? Zijn er simulatiemodellen die inzicht geven in de effecten van individuele acties op het geheel? Succesvolle modellen moeten zowel de fysieke als de sociale werkelijkheid van het systeem "vangen". Ze moeten gebruikers ook de mogelijkheid bieden om te experimenteren met *veranderingen* in zowel de fysieke als de sociale configuratie van het netwerk. Met andere woorden, socio-technische systemen stellen modelleurs voor een grote uitdaging.

Er zijn de nodige instrumenten beschikbaar om ofwel het fysieke ofwel het sociale netwerk te modelleren, maar socio-technische systemen vragen om een geïntegreerde modelleeraanpak die beide werelden bijeenbrengt. Dat is de ambitie van dit proefschrift. Een bijkomende uitdaging is om dit doel niet voor slechts een enkel domein, zoals energie of industrie, te bereiken maar om een modelleerinfrastructuur op te zetten die sectoroverschrijdend is en die om kan gaan met onderling verbonden systemen; denk bijvoorbeeld aan de verwevenheid van energie- en transportsystemen, of aan de verwevenheid van de telecommunicatie- en elektriciteitsinfrastructuur. Dit proefschrift biedt een geïntegreerd raamwerk voor het modelleren van socio-technische systemen. Deze aanpak helpt modelleurs bij het ontwikkelen van betere modellen om daarmee het besluitvormingsproces te ondersteunen van actoren die betrokken zijn bij het reguleren, aansturen, of gebruiken van deze systemen.

Dit proefschrift omvat twee verhaallijnen. De eerste start met een illustratie van de problemen en uitdagingen in het modelleren van socio-technische systemen en de noodzaak van een flexibele, bottom-up wijze van modelleren. Voor dit doel is het *agent-gebaseerde* paradigma het meest veelbelovend. Het resultaat is een modelleer-raamwerk dat aan deze voorwaarden voldoet. De aanpak is toegepast op een aantal cases, waarbij elke volgende casus het generieke raamwerk verrijkt. Deze verhaallijn kan worden aangeduid als het 'raamwerk voor agent-gebaseerde modellen van socio-technische systemen'.

De tweede verhaallijn wordt ingegeven door de behoefte van modelleurs om de keuze voor het gekozen modelleerparadigma te verantwoorden, alsmede door de wetenschappelijke uitdaging om op een objectieve manier het raamwerk ontwikkeld in dit proefschrift te analyseren. Nadat een methode voor het systematisch uitvoeren van zo'n vergelijking is gepresenteerd, wordt de op agenten gebaseerde aanpak vergeleken met andere modelleerparadigma's. De evaluatie van het raamwerk resulteert in vuistregels voor de toepasbaarheid en de bruikbaarheid. Vervolgens worden twee modellen, ontwikkeld met het raamwerk, ingezet om een probleemeigenaar te ondersteunen bij het oplossen van problemen. Deze tweede verhaallijn kan worden bestempeld als de 'kritische evaluatie van agent-gebaseerde modellen voor socio-technische systemen'.

Raamwerk voor agent-gebaseerde modellen van socio-technische systemen

Om de socio-technische complexiteit hanteerbaar te maken voor actoren is een generiek agent-gebaseerd raamwerk ontwikkeld (Hoofdstuk 3).Het raamwerk is opgebouwd uit de volgende drie typen onderdelen:

- **Interface** tussen onderdelen van modellen, tussen modellen, tussen ontwikkelaars en tussen ontwikkelaars en probleemeigenaren.
- **Bibliotheek** van broncode die kan worden hergebruikt.
- **Procedures** die beschrijven hoe de bibliotheek en interface gebruikt worden om modellen te bouwen.

In een op agenten gebaseerd model wordt het systeem beschreven door het uit te drukken in agenten en hun gedrag, waarbij een agent een model is van een beslissingnemende entiteit op verschillende niveaus van aggregatie, van een individu tot een collectief. Agenten worden beschouwd als software-entiteiten die autonoom zijn, kunnen reageren op de omgeving, pro-actief gedrag en sociaal gedrag kunnen vertonen. Het op agenten gebaseerde paradigma is geschikt om het sociale gedrag van entiteiten te beschrijven. Bovendien biedt het een flexibele bottom-up aanpak die nodig is om experimenten uit te voeren met veranderingen in elementen (die weer leiden tot een veranderende configuratie van het systeem) om de effecten op het globale systeemgedrag te bestuderen.

De hoeksteen van het raamwerk is een gedeelde taal die geformaliseerd is in een ontologie, hetgeen een formele specificatie van concepten is. De ontologie vormt de interface die nodig is om verschillende elementen van het systeem (zowel sociale als fysieke) samen te brengen en om verschillende modellen aan elkaar te koppelen. Naast interconnectiviteit, biedt de ontologie *interoperability*, wat betekent dat verschillende onderdelen van een systeem samen kunnen werken. De ontologie voorziet in een verzameling van (abstracte) categorieën en eigenschappen waarmee (concrete) instanties kunnen worden gedefinieerd. De instanties zijn de systeemelementen — of de feiten — die in het model zijn opgenomen en worden opgeslagen in een gedeelde kennisbank. Bovendien worden de concepten uit de ontologie (dat wil zeggen, de woorden in de gedeelde taal) ook gebruikt om het gedrag van agenten te definiëren. Tenslotte is een gemeenschappelijke taal onontbeerlijk voor betekenisvolle communicatie tussen inhoudsdeskundigen uit verschillende disciplines.

Het raamwerk kan helpen bij het opzetten van nieuwe modellen van socio-technische systemen door het volgen van een aantal modelleerstappen en, waar mogelijk, hergebruik van reeds bestaande bouwstenen (bv. feiten, procedures, agenten of technologieën) uit modellen die eerder zijn ontwikkeld. Als nieuwe elementen zijn gemaakt voor een specifieke studie kunnen ze vervolgens worden teruggevoerd in het gedeelde raamwerk, met als resultaat dat ze beschikbaar komen voor hergebruik. Een basisverzameling met definities van categorieën is ontwikkeld voor een aantal initiële cases en verfijnd door verdere toepassingen.

De aanpak die in dit proefschrift wordt gepresenteerd is toegepast op een aantal casestudy's van socio-technische systemen (Hoofdstuk 4). Toepassingen zijn onder andere een multi-modale hub voor vrachttransport, een productieketen van een olieraffinaderij en van een chocoladeproducent. Met een beschrijving van de ontwikkeling van die modellen worden de modelbouwprocedures gedemonstreerd en wordt getoond hoe het raamwerk modelleurs ondersteunt. Deze procedures behelzen de conceptualisatie van het systeem door agenten en fysieke entiteiten te definiëren, het verfijnen van de generieke ontologie met probleem-specifieke concepten, het toevoegen van concrete instanties en het implementeren van het gedrag van de agenten. Vervolgens is aangetoond dat deze procedures door andere modelleurs toepasbaar zijn in een verscheidenheid aan problemen in diverse infrastructuurdomeinen.

De ontwikkeling van het raamwerk over de tijd, door cycli van toepassing en verfijning alsmede door bijdragen van vele gebruikers, is goed gedocumenteerd en het ontwerptraject zelf is bestudeerd en geanalyseerd (Hoofdstuk 5). De compleetheid, correctheid en toepasbaarheid van het raamwerk zijn getest. De conclusie is dat de ontologie compleet is voor het bestek waarvoor zij is gedefinieerd en eenvoudig kan worden uitgebreid voor nieuwe problemen. Na de initiële ontwikkelingsfase zijn geen van de voornaamste concepten veranderd of vervangen hetgeen, samen met een wijdverspreid gebruik door meerdere modelleurs, aangeeft dat de ontologie adequaat is. Tenslotte is aangetoond dat door hergebruik van instanties in de gedeelde kennisbank, alsmede de broncode van modellen, het oplossen van nieuwe problemen gebruikmakend van het raamwerk minder tijdrovend is. Na de aanloopkosten van het ontwikkelen van de eerste generaties modellen voor de initiële cases, kunnen toepassingen voor nieuwe cases steeds efficiënter worden gerealiseerd.

Kritische evaluatie van agent-gebaseerde modellen voor socio-technische systemen

Er is behoefte aan een kritische evaluatie van de voor- en nadelen van het raamwerk en een gedetailleerde vergelijking met andere modelleerparadigma's. Het vergelijken van modelleerparadigma's op basis van slechts de conceptuele modelspecificaties is niet voldoende; in plaats daarvan zijn een goed-gedefinieerd proces voor benchmarken alsook het uitvoeren van experimenten noodzakelijk. Door het bouwen van verschillende modellen en te analyseren hoe ze zijn ontwikkeld en hoe ze kunnen worden uitgebreid, kan een goed-gefundeerde verantwoording van de keuze voor een modelleerparadigma gemaakt worden, en kunnen aanbevelingen en vuistregels worden opgesteld over welk paradigma geschikter is voor welk probleem.

Een van de grootste uitdagingen in het vergelijken van modelleerparadigma's ligt in de definitie van wat elk van de paradigma's in de studie omvat (Hoofdstuk 2). Een onderscheid tussen agent-gebaseerde modellen en op stelsels vergelijkingen gebaseerde modellen, zoals gevonden in de literatuur, gaat voorbij aan het feit dat beide *labels* op verschillende manieren kunnen worden geïnterpreteerd. Het onderscheid is niet zwart-wit; veeleer is er sprake van een continu spectrum in de modelleerruimte. Er zijn twee assen waarop modellen kunnen verschillen: de *modelelementen* as en de *systeembeschrijvingselementen* as. De eerste specificeert *wat* is gemodelleerd en wat de bestanddelen van het model zijn, terwijl de tweede specificeert *hoe* de structuur en het gedrag formeel zijn beschreven. De bestanddelen variëren van individuen (dat wil zeggen, beslissende entiteiten) tot op systeemniveau waarneembare zaken. De elementen die het systeem beschrijven bevinden zich ergens op de as tussen puur gebruik van wiskundige vergelijkingen en het uitdrukken van gedrag in algoritmes. Deze nuance maakt het mogelijk dat de conclusies van het benchmarken kunnen worden gegeneraliseerd van de specifieke modellen die worden vergeleken tot de voor- en nadelen van modelleerparadigma's.

Een algemeen schema om modelleerparadigma's te vergelijken wordt in dit proefschrift voorgesteld (Hoofdstuk 6), met speciale nadruk op het identificeren van wat er wordt vergeleken, de evaluatie van vergelijkbaarheid van studieobjecten en het beschrijven van goed-gestructureerde experimenten. Op deze manier kunnen eerlijke en gebalanceerde conclusies worden getrokken. De aanpak is daarna gebruikt om verschillende modellen van de productieketen van een olieraffinaderij te vergelijken. Elk model is ontwikkeld gebruikmakend van een ander modelleerparadigma: het ene model gebruikt een numeriek softwarepakket en het andere een agent-gebaseerd platform. Aangetoond wordt dat de beide modelleerparadigma's en hun — zeer verschillende — modelleergereedschap succesvol gebruikt kunnen worden om een model van hetzelfde systeem te maken met vergelijkbare resultaten. Door het analyseren van de inspanning benodigd voor het uitbreiden van de modellen om nieuwe scenario's te kunnen uitproberen worden de sterke en zwakke punten van beide paradigma's geïdentificeerd binnen de context van het modelleren van productieketens. Het gemak van het uitdrukken van het probleem, de uitbreidbaarheid en de herbruikbaarheid van (onderdelen van) het simulatiemodel zijn daarbij de prestatieindicatoren.

De resultaten van het benchmarken kunnen, binnen de context waarin de vergelijking is uitgevoerd, worden uitgebreid van de specifieke modellen tot de paradigma's. Productieprocessen en de technologische aspecten worden goed gekarakteriseerd door wiskundige vergelijkingen, terwijl de besluitvorming door actoren slechts met algoritmes kan worden gevat. Het complete systeem kan echter volledig worden uitgedrukt door middel van beide paradigma's die werden vergeleken. Gesteld kan worden dat modellen gebaseerd op stelsels van vergelijkingen, in het algemeen, meer geschikt zijn om de fysieke elementen in het systeem te beschrijven terwijl de (dynamische) interactie tussen de actoren het beste kan worden gerepresenteerd door agent-gebaseerde modellen. Voor het uitbreiden of aanpassen van modellen geldt de algemene regel dat het veranderen van indirecte eigenschappen in het model meer moeite kost. Dit betekent dat voor het agent-gebaseerde model, waarin de inrichting van het systeem dynamisch is en niet vastligt in de structuur van het model, het gemakkelijk is om nieuwe actoren, nieuwe fysieke elementen en als consequentie nieuwe mogelijke relaties toe te voegen. Echter, veranderingen in de manier waarop het technische systeem zelf werkt worden gemakkelijker doorgevoerd door aanpassing van wiskundige vergelijkingen. Als een nieuw model wordt gebouwd gebaseerd op eerder werk, kan voor het wiskundige model de conceptualisatie worden hergebruikt maar geen enkele van de vergelijkingen zelf. In het geval van het bottom-up agent-gebaseerde model kunnen ook specifieke bouwstenen worden hergebruikt of uitgebreid. Ten slotte, bij het uitleggen van het model en de modelresultaten aan belanghebbenden biedt het agent-paradigma een natuurlijke representatie van het besluitvormingsproces alsook van de interacties tussen de entiteiten in het systeem. Een model op basis van wiskundige vergelijkingen biedt inzicht in de technische processen en heeft daarmee een voorsprong voor een technisch deskundig publiek.

Na de benchmarking en de daaruit volgende lessen over de voordelen en gebreken van agent-gebaseerd modelleren is gedemonstreerd hoe simulatiemodellen ontwikkeld met het raamwerk uit dit proefschrift een probleemeigenaar kunnen ondersteunen door het oplossen van een specifiek multi-actor, multi-criteria en multi-level beslissingsprobleem (Hoofdstuk 7). Met een beslismodel, geformuleerd voor een specifiek doel en een specifieke vraag, kan een probleem worden opgelost door middel van verschillende zoekstrategieën en gebruikmakend van een simulatiemodel van het systeem.

Om te laten zien hoe agent-gebaseerde modellen kunnen worden toegepast als beslissingsondersteunend systeem worden twee illustratieve toepassingen gepresenteerd, beiden door echte problemen geïnspireerd. De eerste gebruikt het agent-gebaseerde model van een productieketen van een olieraffinaderij en de tweede het model van de multimodale hub voor vrachttransport. Er is aangetoond hoe een zoekstrategie uit de beslis-kunde, zoals de Nelder-Mead optimalisatiemethode, kan worden toegepast op een beslissingsprobleem betreffende verstoringen in de productieketen van een olieraffinaderij. In een andere studie wordt getoond hoe verschillende belastingmaatregelen kunnen helpen om actoren met verschillende belangen te bewegen tot instemming met een bepaalde locatie voor een nieuwe multi-modale transporthub. Zodoende is aangetoond dat agent-gebaseerde modellen ontwikkeld met behulp van het in dit proefschrift beschreven raamwerk gebruikt kunnen worden om beslissers te ondersteunen.

Conclusies

Het raamwerk gepresenteerd in dit proefschrift is een adequate generieke modelleeraanpak voor socio-technische systemen. Agent-gebaseerde modellen zijn in het bijzonder geschikt om te experimenteren met verschillende scenario's en om *wat als* vragen aan te pakken. Dit biedt waardevolle ondersteuning aan beslissers om adequaat te reageren op bijvoorbeeld verstoringen in het fysieke systeem, nieuwe wetgeving of nieuwe regulering.

De modellen die in het kader van dit proefschrift gebouwd en beproefd zijn, zijn ontwikkeld op een bottom-up wijze, wat het relatief gemakkelijk maakt om sociale of de fysieke configuratie van het gemodelleerde systeem te veranderen. Zo kunnen er nieuwe actoren aan het systeem worden toegevoegd (bv. andere gebruikers van de transporthub of nieuwe leveranciers met andere prijzen en levertijden voor de olieraffinaderij) of kan de fysieke inrichting worden aangepast (bv. nieuwe transportverbindingen in een vrachtsysteem of extra opslagtanks voor de raffinaderij). Het raamwerk is vanaf het begin generiek opgezet. Dat maakt het mogelijk om een verscheidenheid aan infrastructuren en andere socio-technische systemen te modelleren, en lessen uit het ene domein te vertalen naar andere domeinen. Bij het analyseren van over sectorgrenzen heen verbonden netwerken kunnen (delen van) modellen van verschillende infrastructuursystemen aan elkaar worden gekoppeld.

Het doorontwikkelen van het raamwerk is een iteratief proces door voortdurend (her)gebruik; nieuwe modelleurs passen de aanpak toe op nieuwe cases en problemen en dragen op die manier bij aan het gedeelde raamwerk. Dit is een van de belangrijkste winstpunten van de agent-gebaseerde aanpak: hoe meer het raamwerk wordt gebruikt, hoe meer er kan worden hergebruikt. Bij deze is de lezer uitgenodigd om de uitdagingen voor de infrastructuursystemen van de toekomst in een socio-technisch en agent-gebaseerd perspectief te plaatsen en om de systeemelementen te representeren in de hier gepresenteerde ontologie. De modelleeraanpak van dit proefschrift kan zo worden gebruikt om effectief betere modellen te maken en om het modelleerproces efficiënter te maken.

Koen Haziël van Dam

Curriculum vitae

Koen Haziël van Dam was born on the 7th of March 1978 in Delft, the Netherlands. After completing his pre-university education (VWO) in 1996 at the Goois Lyceum in Bussum, he studied for his Master's degree in Artificial Intelligence at the faculty of Mathematics and Computer Science of the Vrije Universiteit in Amsterdam. With an internship at the Netherlands Institute for Research (TNO) and a specialisation in knowledge engineering, he graduated in 2002 with his MSc thesis titled "Design of a generic ontology translator — implemented for JADE".

In 2003 and 2004 Koen worked as a researcher at the faculty of Mechanical Engineering of the Delft University of Technology (TU Delft) on the control of automated guided vehicles before starting as a doctoral candidate at the faculty of Technology, Policy and Management of TU Delft in 2005. As a member of the Energy and Industry group and participant in the Intelligent Infrastructures subprogramme of the Next Generation Infrastructures Foundation, he performed research on agent-based modelling of socio-technical infrastructure systems. In 2008 he spent four months as a visiting researcher at the Department of Chemical and Biomolecular Engineering of the National University of Singapore.

During his PhD project his main research interests were agent-based modelling, benchmarking modelling paradigms, ontology design, energy and transport infrastructures (in particular oil refineries) and complex systems. The results of his work have been published in over twenty articles and presented at a dozen international conferences and seminars in Europe, Asia, and the United States of America. The project came to a conclusions with his PhD thesis "Capturing socio-technical systems with agent-based modelling", supervised by prof. dr. ir. Weijnen and dr. ir. Lukszo and to be defended on the 30th of October 2009.

Alongside his work as a researcher, Koen has been active as board member of various organisations that represent researchers. First at the university level, in 2004 as member of the PhD Council of research school TRAIL and then from 2004 to 2005 as vice-president of the representative body of doctoral candidates at TU Delft (Promood). This was followed by a position as board member of the national PhD Network of the Netherlands (PNN) from 2005 to 2007. Finally, he was elected president of the European Council for Doctoral Candidates and Young Researchers (Eurodoc) in 2007/2008, representing early stage researchers at the European level.

List of publications

van Dam, K. H. (2002). *Design of a generic ontology translator — implemented for JADE*, Master's thesis, Vrije Universiteit, Amsterdam, the Netherlands.

van Dam, K. H., Ottjes, J. A., Lodewijks, G., Verwater-Lukszo, Z. & Wagenaar, R. (2004). Intelligent infrastructures: Distributed intelligence in transport system control — an illustrative example., *Proceedings of the 2004 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, the Hague, the Netherlands, pp. 4650–4654.

Rammelt, C. F., Nikolic, I., Boes, J. & van Dam, K. H. (2005). Complex systems approach to 'development aid', *Proceedings of the 11th Annual International Sustainable Development Research Conference*, Helsinki, Finland.

van Dam, K. H., Nikolic, I., Lukszo, Z. & Dijkema, G. P. (2006). Towards a generic approach for analyzing the efficiency of complex networks, *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, Ft. Lauderdale, Florida, USA.

Nikolic, I., Dijkema, G. P., van Dam, K. H. & Lukszo, Z. (2006). General methodology for action-oriented industrial ecology: Complex systems approach applied to the rotterdam industrial cluster, *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, Ft. Lauderdale, Florida, USA, pp. 831–836.

van Dam, K. H., Verwater-Lukszo, Z., Ottjes, J. A. & Lodewijks, G. (2006). Distributed intelligence in autonomous multi-vehicle systems, *International Journal of Critical Infrastructures* 2(2/3): 261–272.

van Dam, K. H., Houwing, M., Lukszo, Z. & Bouwmans, I. (2006). Modelling an electricity infrastructure as a multi-agent system — lessons learnt from manufacturing control, *in* W. Marquardt & C. Pantelides (eds), *Proceedings of the 16th European Symposium on Computer Aided Process Engineering and 9th international symposium on Process Systems Engineering (ESCAPE-16 and PSE2006)*, Vol. 21 of *Computer-Aided Chemical Engineering*, Elsevier, Garmisch-Partenkirchen, Germany.

van Dam, K. H. & Lukszo, Z. (2006). Modelling energy and transport infrastructures as a multi-agent system using a generic ontology, *Proceedings of the 2006 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, The Grand Hotel, Taipei, Taiwan, pp. 890–895.

van Dam, K. H., Lukszo, Z., Ferreira, L. & Sirikijpanichkul, A. (2007). Planning the location of intermodal freight hubs: an agent based approach, *Proceedings of the 2007 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, London, England.

van Dam, K. H., Lukszo, Z., Srinivasan, R. & Karimi, I. A. (2007). Opportunities for agent-based models in computer aided process engineering, *in* V. Plesu & P. Agachi (eds), *Proceedings of the European Symposium on Computer Aided Process Engineering ESCAPE 17*, Vol. 24 of *Computer-Aided Chemical Engineering*, Elsevier, Bucharest, Romania.

Chappin, E. J., Dijkema, G. P., van Dam, K. H. & Lukszo, Z. (2007). Modeling strategic and operational decision-making — an agent-based model of electricity producers, *Proceedings of the 21st annual European Simulation and Modelling Conference (ESM2007)*, St. Julian's, Malta.

Sirikijpanichkul, A., van Dam, K. H., Ferreira, L. & Lukszo, Z. (2007). Optimizing the location of intermodal freight hubs: An overview of the agent based modelling approach, *Journal of Transportation Systems Engineering and Information Technology* 7(4): 71–81.

Lukszo, Z., van Dam, K. H., Weijnen, M. P. & Dijkema, G. P. (2008). Agent-based models for crisis management, *in* H. Bouwman, R. Bons, M. Hoogewegen, M. Janssen & H. Pronk (eds), *Let a Thousand Flowers Bloom, Essays in commemoration of prof.dr. René Wagenaar*, IOS Press, pp. 281–299.

van Dam, K. H., Adhitya, A., Srinivasan, R. & Lukszo, Z. (2008). Benchmarking numerical and agent-based models of an oil refinery supply chain, *in* B. Braunschweig & X. Joulia (eds), *Proceedings of the European Symposium on Computer Aided Process Engineering ESCAPE 18*, Vol. 25 of *Computer-Aided Chemical Engineering*, Elsevier, Lyon, France, pp. 623–628.

van Dam, K. H., Houwing, M., Lukszo, Z. & Bouwmans, I. (2008). Agent-based control of distributed electricity generation with micro combined heat and power — cross-sectoral learning for process and infrastructure engineers, *Journal of Computers and Chemical Engineering* 21(1–2): 205–217.

van Dam, K. H., Adhitya, A., Srinivasan, R. & Lukszo, Z. (2008). Nuances of benchmarking agent-based and equation-based models of an oil refinery supply chain, *Proceedings of the First International Conference on Infrastructure Systems: Building Networks for a Brighter Future*, Rotterdam, The Netherlands.

Nikolic, I., Dijkema, G. P. & van Dam, K. H. (2009). Understanding and shaping the evolution of sustainable large-scale socio-technical systems — towards a framework for action oriented industrial ecology, *in* M. Ruth & B. Davidsdottir (eds), *The Dynamics of Regions and Networks in Industrial Ecosystems*, Edward Elgar.

van Dam, K. H., Lukszo, Z. & Srinivasan, R. (2009). Abnormal situation management in a refinery supply chain supported by an agent-based simulation model, *in* R. M. de Brito Alves, C. A. O. do Nascimento & E.C. Biscaia,Jr. (eds), *Proceedings of the*

10th International Symposium on Process Systems Engineering — PSE2009, Salvador, Bahia, Brazil.

van Dam, K. H., Adhitya, A., Srinivasan, R. & Lukszo, Z. (2009). Critical evaluation of paradigms for modelling integrated supply chains, *Journal of Computers and Chemical Engineering* 33(10): 1711–1726.

van Dam, K. H., Lukszo, Z. & Srinivasan, R. (2009). Agent-based decision support for failure-prone networked infrastructures, *International Journal of Critical Infrastructures*. in print.

van Dam, K. H. & Lukszo, Z. (2009) Model factory for socio-technical infrastructure systems, *in* R.R. Negenborn, Z. Lukszo & J. Hellendoorn (eds.), *Intelligent Infrastructures*, Intelligent Systems, Control and Automation: Science and Engineering, Springer. in print.

van Dam, K. H. (2009). *Capturing socio-technical systems with agent-based modelling*, PhD thesis, Delft University of Technology, Delft, the Netherlands.

NGInfra PhD thesis series on infrastructures

1. Strategic behavior and regulatory styles in the Netherlands energy industry
   Martijn Kuit, 2002, Delft University of Technology, the Netherlands.

2. Securing the public interest in electricity generation markets — The myths of the invisible hand and the copper plate
   Laurens de Vries, 2004, Delft University of Technology, the Netherlands.

3. Quality of service routing in the internet — Theory, complexity and algorithms
   Fernando Kuipers, 2004, Delft University of Technology, the Netherlands.

4. The role of power exchanges for the creation of a single European electricity market — Market design and market regulation
   François Boisseleau, 2004, Delft University of Technology, the Netherlands, and University of Paris IX Dauphine, France.

5. The ecology of metals
   Ewoud Verhoef, 2004, Delft University of Technology, the Netherlands.

6. MEDUSA — Survivable information security in critical infrastructures
   Semir Daskapan, 2005, Delft University of Technology, the Netherlands.

7. Transport infrastructure slot allocation
   Kaspar Koolstra, 2005, Delft University of Technology, the Netherlands.

8. Understanding open source communities — An organizational perspective
   Ruben van Wendel de Joode, 2005, Delft University of Technology, the Netherlands.

9. Regulating beyond price — Integrated price-quality regulation for electricity distribution networks
   Viren Ajodhia, 2006, Delft University of Technology, the Netherlands.

10. Networked reliability — Institutional fragmentation and the reliability of service provision in critical infrastructures
    Mark de Bruijne, 2006, Delft University of Technology, the Netherlands.

11. Regional regulation as a new form of telecom sector governance — The interactions with technological socio-economic systems and market performance
Andrew Barendse, 2006, Delft University of Technology, the Netherlands.

12. The internet bubble — The impact on the development path of the telecommunications sector
Wolter Lemstra, 2006, Delft University of Technology, the Netherlands.

13. Multi-agent model predictive control with applications to power networks
Rudy Negenborn, 2007, Delft University of Technology, the Netherlands.

14. Dynamic bi-level optimal toll design approach for dynamic traffic networks
Dusica Joksimović, 2007, Delft University of Technology, the Netherlands.

15. Intertwining uncertainty analysis and decision-making about drinking water infrastructure
Machtelt Meijer, 2007, Delft University of Technology, the Netherlands.

16. The new EU approach to sector regulation in the network infrastructure industries
Richard Cawley, 2007, Delft University of Technology, the Netherlands.

17. A functional legal design for reliable electricity supply — How technology affects law
Hamilcar Knops, 2008, Delft University of Technology, the Netherlands and Leiden University, the Netherlands.

18. Improving real-time train dispatching — Models, algorithms and applications
Andrea D'Ariano, 2008, Delft University of Technology, the Netherlands.

19. Exploratory modelling and analysis — A promising method to deal with deep uncertainty
Datu Buyung Agusdinata, 2008, Delft University of Technology, the Netherlands.

20. Characterization of complex networks — Application to robustness analysis
Almerima Jamaković, 2008, Delft University of Technology, Delft, the Netherlands.

21. Shedding light on the black hole — The roll-out of broadband access networks by private operators
Marieke Fijnvandraat, 2008, Delft University of Technology, Delft, the Netherlands.

22. On Stackelberg and inverse Stackelberg games & their applications in the optimal toll design problem, the energy markets liberalization problem, and in the theory of incentives
Kateřina Staňková, 2009, Delft University of Technology, Delft, the Netherlands.

23. On the conceptual design of large-scale process & energy infrastructure systems — Integrating flexibility, reliability, availability, maintainability and economics (FRAME) performance metrics
Austine Ajah, 2009, Delft University of Technology, Delft, the Netherlands.

24. Comprehensive models for security analysis of critical infrastructure as complex systems
    Fei Xue, 2009, Politecnico di Torino, Torino, Italy.

25. Towards a single European electricity market — A structured approach for regulatory mode decision-making
    Hanneke de Jong, 2009, Delft University of Technology, the Netherlands.

26. Co-evolutionary process for modelling large scale socio-technical systems evolution
    Igor Nikolić, 2009, Delft University of Technology, the Netherlands.

27. Regulation in splendid isolation — A framework to promote effective and efficient performance of the electricity industry in small isolated monopoly systems
    Steven Martina, 2009, Delft University of Technology, the Netherlands.

28. Reliability-based dynamic network design with stochastic networks
    Hao Li, 2009, Delft University of Technology, the Netherlands.

29. Coping with competing public values
    Bauke Steenhuisen, 2009, Delft University of Technology, the Netherlands.

30. Innovative contracting practices in the road sector — Cross-national lessons in dealing with opportunistic behaviour
    Mónica Altamirano, 2009, Delft University of Technology, the Netherlands.

31. Reliability in urban public transport network assessment and design
    Shahram Tahmasseby, 2009, Delft University of Technology, the Netherlands.

32. Capturing socio-technical systems with agent-based modelling
    Koen van Dam, 2009, Delft University of Technology, the Netherlands.

Order information: info@nginfra.nl

Koen Haziël van Dam

Capturing socio-technical systems with agent-based modelling

What is a suitable modelling approach for socio-technical systems? The answer to this question is of great importance to decision makers in large scale interconnected network systems. The behaviour of these systems is determined by many actors, situated in a dynamic, multi-actor, multi-objective and multi-level jungle. Models to support such an actor should be able to capture both the physical and social reality of the system, their interactions with one another and the external dynamic environment. Moreover, they must allow users to experiment with changes in both the physical and the social network configuration.

To deal with these challenges a generic *agent-based* modelling framework for socio-technical systems is developed in this thesis. The cornerstone of the framework is a shared language formalised in an *ontology*, which forms the interface needed to bring different elements of the system (both social and physical) together, to interconnect different models and ensure interoperability. The re-usability of building blocks helps modellers build new models more efficiently. The models developed with the new framework are shown to offer valuable decision support in case studies of an oil refinery supply chain and an intermodal freight hub.

The Next Generation Infrastructures Foundation
represents an international consortium of knowledge institutions, market players and governmental bodies, which joined forces to cope with the challenges faced by today's and tomorrow's infrastructure systems. The consortium cuts across infrastructure sectors, across discplinary borders and across national borders, as infrastructure systems themselves do. With the strong participation of practitioners in a concerted knowledge effort with social and engineering scientists, the Foundation seeks to ensure the conditions for utilization of the research results by infrastructure policy makers, regulators and the infrastructure industries.

www.nginfra.nl

NEXT GENERATION INFRASTRUCTURES FOUNDATION

9 789079 787128