

Advanced Information and Knowledge Processing

Longbing Cao

Metasynthetic Computing and Engineering of Complex Systems



Springer

Advanced Information and Knowledge Processing

Series Editors

Lakhmi C. Jain

University of Canberra and University of South Australia

Xindong Wu

University of Vermont

Information systems and intelligent knowledge processing are playing an increasing role in business, science and technology. Recently, advanced information systems have evolved to facilitate the co-evolution of human and information networks within communities. These advanced information systems use various paradigms including artificial intelligence, knowledge management, and neural science as well as conventional information processing paradigms. The aim of this series is to publish books on new designs and applications of advanced information and knowledge processing paradigms in areas including but not limited to aviation, business, security, education, engineering, health, management, and science. Books in the series should have a strong focus on information processing - preferably combined with, or extended by, new results from adjacent sciences. Proposals for research monographs, reference books, coherently integrated multi-author edited books, and handbooks will be considered for the series and each proposal will be reviewed by the Series Editors, with additional reviews from the editorial board and independent reviewers where appropriate. Titles published within the Advanced Information and Knowledge Processing series are included in Thomson Reuters' Book Citation Index.

More information about this series at <http://www.springer.com/series/4738>

Longbing Cao

Metasynthetic Computing and Engineering of Complex Systems

 Springer

Longbing Cao
Advanced Analytics Institute
University of Technology, Sydney
Sydney, New South Wales
Australia

ISSN 1610-3947
Advanced Information and Knowledge Processing
ISBN 978-1-4471-6550-7 ISBN 978-1-4471-6551-4 (eBook)
DOI 10.1007/978-1-4471-6551-4

Library of Congress Control Number: 2015941316

Springer London Heidelberg New York Dordrecht
© Springer-Verlag London 2015

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

Springer-Verlag London Ltd. is part of Springer Science+Business Media (www.springer.com)

Preface

We are in the era of complex systems.

An increasingly critical research issue facing researchers in different disciplines is how to compute and engineer complex systems and how to build corresponding problem-solving systems.

Various system metaphors have accordingly been proposed to address the intricacies of complex systems. This effort has become increasingly focused in recent decades as complex systems have become omnipresent in our everyday, social, business, and web worlds.

Reductionism has proved to be a very effective philosophy for tackling complexities in physical and chemical systems, in particular by dedicating priority to the parts of a system rather than the whole. Its principal ideas have widely influenced the research and engineering of complex scientific problems in many domains and business applications.

In recent decades, holism has emerged, bringing with it a focus on systems thinking and an emphasis on understanding complexities in the whole rather than merely in part. System sciences and cognitive sciences have a particular interest in holism. The science of complexity urges the exploration of holism for understanding complex natural and social systems.

Complementing the challenges of applying reductionism and holism for complex systems, systematology combines top-down holistic methodologies with bottom-up reductionistic approaches to consider the complexities in the whole system and its parts, as well as their respective connections. Systematology appears to be really important for addressing the complexities of open complex giant systems, in which it may not be effective to apply only reductionism or holism.

Faced with open complex giant systems like the Internet, we often do not know what we do not know. A qualitative-to-quantitative metasynthesis which explores the synergy of human intelligence and machine intelligence in a human-centered, human-machine-cooperative way may be helpful in understanding system complexities and building problem-solving systems.

Metasynthetic computing and engineering take human-centered, human-machine-cooperative, qualitative-to-quantitative metasynthesis as the main guide for understanding open complex systems. In this book, we outline the corresponding aspects: Chap. 1 presents an overview of complex systems, Chap. 2 considers ubiquitous intelligence in complex systems, Chap. 3 examines system methodologies, Chap. 4 explores computing paradigms, Chap. 5 focuses on metasynthesis, Chap. 6 looks at organization- and service-oriented methodology for engineering complex systems, Chaps. 7, 8, and 9 investigate visual modeling, formal modeling, and integrative modeling of complex systems, Chaps. 10 and 11 discuss the architecture and detailed design of complex problem-solving systems, Chap. 12 addresses ontological engineering, and Chaps. 13, 14, and 15 detail several case studies in building problem-solving systems for actionable knowledge discovery and understanding complex behavior and social data.

The overarching goal of compiling this book is to encourage and inspire discussion and reflection, and the implementation of effective methodologies and tools for computing and engineering open complex systems and problems, while synergizing ubiquitous intelligence, including human, domain, social, network, data, and machine intelligence, during the respective problem-solving processes and in problem-solving systems.

Sydney, Australia

Longbing Cao

Contents

- 1 Complex Systems 1**
 - 1.1 Introduction 1
 - 1.2 System Complexities 1
 - 1.3 System Transparency 5
 - 1.3.1 Black Boxes 5
 - 1.3.2 White Boxes 6
 - 1.3.3 Glass Boxes 6
 - 1.3.4 Grey Boxes 6
 - 1.4 System Classification 7
 - 1.5 Complex Agent Systems 9
 - 1.5.1 Multiagent Systems 9
 - 1.5.2 Large-Scale Systems 13
 - 1.5.3 Large-Scale Multiagent Systems 13
 - 1.5.4 Open Complex Agent Systems 17
 - 1.6 Hybrid Intelligent Systems 19
 - 1.6.1 Concept 19
 - 1.6.2 Hybridization Strategies 20
 - 1.6.3 Design Strategies 22
 - 1.6.4 Typical Hybrid Applications 23
 - 1.7 Evolution of Intelligent Systems 25
 - 1.8 Open Giant Intelligent Systems 29
 - 1.9 Computing and Engineering Complex Systems 31
 - 1.10 Summary 33
 - References 33
- 2 Ubiquitous Intelligence 37**
 - 2.1 Introduction 37
 - 2.2 Data Intelligence 38
 - 2.2.1 What Is Data Intelligence? 38
 - 2.2.2 Aims of Involving Data Intelligence 38
 - 2.2.3 Aspects of Data Intelligence 39

2.3	Domain Intelligence	40
2.3.1	What Is Domain Intelligence?	40
2.3.2	Aims of Involving Domain Intelligence	40
2.3.3	Aspects of Domain Intelligence	41
2.4	Network Intelligence	41
2.4.1	What Is Network Intelligence?	41
2.4.2	Aims of Involving Network Intelligence	42
2.4.3	Aspects of Network Intelligence	42
2.5	Human Intelligence	43
2.5.1	What Is Human Intelligence?	43
2.5.2	Aims of Involving Human Intelligence	43
2.5.3	Aspects of Human Intelligence	44
2.6	Organizational Intelligence	45
2.6.1	What Is Organizational Intelligence?	45
2.6.2	Aims of Involving Organizational Intelligence	45
2.6.3	Aspects of Organizational Intelligence	46
2.7	Social Intelligence	46
2.7.1	What Is Social Intelligence?	46
2.7.2	Aims of Involving Social Intelligence	47
2.7.3	Aspects of Social Intelligence	47
2.8	Metasynthesis of Ubiquitous Intelligence	48
2.9	Summary	49
	References	49
3	System Methodologies	51
3.1	Introduction	51
3.2	Reductionism	52
3.3	Holism	53
3.4	Systematology	53
3.5	Summary	55
	References	55
4	Computing Paradigms	57
4.1	Introduction	57
4.2	Objects and Object-Oriented Methodology	58
4.3	Components and Component-Based Methodology	58
4.4	Services and Service-Oriented Methodology	59
4.5	Agents and Agent-Oriented Methodology	60
4.5.1	Goal-Oriented Requirements Analysis	61
4.5.2	Agent-Oriented Software Engineering	62
4.5.3	Issues in Agent-Oriented Software Engineering	65
4.6	Relations Among Agents, Objects, Components, and Services	66
4.7	Autonomic Computing	67
4.8	Organizational Computing	70

4.9	Behavior Computing	71
4.10	Social Computing	74
4.11	Cloud/Service Computing	77
4.12	Metasynthetic Computing	78
	References	78
5	Metasynthesis	81
5.1	Introduction	81
5.2	Open Complex Giant Systems	81
5.3	OCGS System Complexities	85
5.4	Knowledge and Intelligence Emergence	87
5.5	Theoretical Framework of Metasynthesis	93
5.6	Problem-Solving Process in M-Space	95
5.7	Social Cognitive Intelligence Emergence in M-Space	98
	5.7.1 Individual Cognitive Model	98
	5.7.2 Social Cognitive Interaction Model	99
	5.7.3 Cognitive Intelligence Emergence	101
5.8	Thinking Pitfalls in M-Interactions	102
5.9	M-Computing: Engineering OCGS	105
5.10	Discussions	106
	References	108
6	OSOAD Methodology	111
6.1	Introduction	111
6.2	Organizational Abstraction	111
	6.2.1 Actors	112
	6.2.2 Environment	113
	6.2.3 Interaction	113
	6.2.4 Organizational Rules	114
	6.2.5 Organizational Structure	114
	6.2.6 Organizational Goal	115
	6.2.7 Organizational Dynamics	115
6.3	Organization-Oriented Analysis	116
	6.3.1 Challenges for Current Organization-Related Software Engineering	116
	6.3.2 What is Organization-Oriented Analysis?	117
6.4	Agent Service-Oriented Design	119
	6.4.1 Agent Service, Services of Agent, and Services of Service	119
	6.4.2 Why Agent Service-Oriented Design?	119
	6.4.3 What is Agent Service-Oriented Design?	120
	6.4.4 Agent Service-Oriented Architectural Design	123
	6.4.5 Agent Service-Oriented Detailed Design	124
6.5	Building Organization and Service-Oriented Software Engineering	124
6.6	Summary	127
	References	127

7	Visual Modeling	131
7.1	Introduction	131
7.2	Actor Model	131
7.2.1	Actor Classification	131
7.2.2	Role Model	133
7.3	Environment Model	135
7.3.1	Characteristics of Agent Environment	135
7.3.2	Classification of Agent Environment	136
7.3.3	POMDP _{AEI} Model	138
7.4	Modeling Organizational Rules	138
7.4.1	Structural Rules	138
7.4.2	Problem-Solving Rules	139
7.4.3	Rule Combinations	141
7.5	Modeling Organizational Structure	142
7.5.1	GAIRE Model	142
7.6	Organizational Dynamics Analysis	145
7.7	Interaction Ontology	147
7.7.1	Interaction Protocols	147
7.7.2	Organizational Patterns	148
7.7.3	Interaction Levels	150
7.7.4	Interaction Rules	150
7.8	Interaction Protocols Engineering	150
7.8.1	Analysis	151
7.8.2	Interaction Protocol Ontology	151
7.8.3	Specifications of Interaction Protocol	153
7.8.4	Interaction Metaprotocols	154
7.9	Modeling Interaction Patterns	156
7.9.1	Pattern Description Template	156
7.9.2	Case Study: Contract Net Protocol	157
7.10	Agent–Environment Interaction	157
7.10.1	What is Agent–Environment Interaction?	157
7.10.2	Modeling Based on Markov Decision Process	160
7.10.3	Modeling Based on the Science of Complexity	162
7.10.4	Dynamic System Theory	164
7.10.5	Case Study: Markov State Chain	165
7.11	Summary	166
	References	166
8	Formal Modeling	169
8.1	Introduction	169
8.2	First-Order Linear-Time Temporal Logics	169
8.2.1	Formal Assertions	170
8.2.2	Real-Time Temporal Logics	171
8.3	Temporal Specification	172

8.4	Formulae for Organizational Abstraction	173
8.4.1	Actor	173
8.4.2	Environment	174
8.4.3	Rule	174
8.4.4	Properties and Keywords	175
8.5	Modeling Roles	177
8.6	Modeling Interaction	178
8.7	FIPA ACL Message Specifications	180
8.7.1	ACL Protocol Description Language	180
8.7.2	Modeling ACL Messages	181
8.8	Modeling Organizational Goal	182
8.9	Summary	184
	References	184
9	Integrative Modeling	185
9.1	Introduction	185
9.2	Integrating Functional and Nonfunctional Requirements	185
9.2.1	Functional Requirements Analysis	185
9.2.2	Nonfunctional Requirements Analysis	186
9.2.3	Analyzing Integrative Requirements	186
9.3	Visual Modeling	187
9.3.1	Goal-Oriented Visual Modeling	187
9.4	Formal Specifications	189
9.5	Integrative Modeling Framework	190
9.5.1	Business-Oriented Functional Requirements	190
9.5.2	Business-Oriented Nonfunctional Requirements	191
9.5.3	Integrative Modeling	191
9.6	Summary	192
	References	193
10	Agent Service-Oriented Architectural Design	195
10.1	Introduction	195
10.2	Agent Service Model	195
10.2.1	Agent Model	196
10.2.2	Service Model	197
10.3	Agent Service Design Patterns	198
10.3.1	Agent Architecture Patterns	198
10.3.2	Structural and Functional Service Patterns	202
10.4	Agent Service-Oriented Integration Architectures	202
10.4.1	Integration Levels and Techniques	202
10.4.2	Architectures for Application Integration	205
10.5	Agent Service-Oriented Integration Strategies	207
10.5.1	Multiagent + Web Services	207
10.5.2	Multiagent + Service-Oriented Computing	209
10.6	Agent Service Management and Communications	210

10.7	Agent Service Coordination	211
10.7.1	Coordination Methods	211
10.7.2	Coordination Modeling and Patterns	214
10.8	Case Study	217
10.9	Summary	218
	References	218
11	Agent Service-Oriented Detailed Design	221
11.1	Introduction	221
11.2	Agent Service Ontology	221
11.2.1	Extracting Problem-Solving Ontology	221
11.2.2	Developing Agent Service Ontology	223
11.3	Representation of Agent Services	224
11.3.1	Agent Service Specification	224
11.3.2	Case Study: Algorithm Registration Agent Service	226
11.4	Agent Service Endpoint Interfaces	227
11.4.1	Designing Agent Service Interfaces	227
11.4.2	Case Study: Algorithm Service Interface	229
11.5	Directory of Agent Services	231
11.6	Communication of Agent Services	234
11.7	Transport of Agent Services	234
11.8	Mediation of Agent Services	235
11.9	Discovery of Agent Services	236
11.10	Modeling Coordination	237
11.11	Other Strategic Issues	240
11.11.1	Design with Agent Service-Oriented Principles	240
11.11.2	Create a Custom Ontological Directory	240
11.11.3	Define a Schema Management Strategy	240
11.11.4	Always Relate XML to Data	241
11.12	Summary	241
	References	241
12	Ontological Engineering	243
12.1	Introduction	243
12.2	Ontology Profiles	244
12.2.1	From Ontology to Ontological Engineering	244
12.2.2	Domain-Specific Business Ontology	245
12.2.3	Problem-Solving Ontology	247
12.2.4	Ontological Commitment	252
12.3	Ontological Semantic Relationships	254
12.4	Ontological Representation	256
12.4.1	Ontology Modeling Techniques	256
12.4.2	Representing Domain Ontologies	258
12.4.3	Representing Problem-Solving Ontologies	259

12.5	Ontological Semantic Aggregation and Transformation	
	Cross Domains	261
12.5.1	Semantic Aggregation of Semantic Relationships	261
12.5.2	Semantic Aggregation of Ontological Items	262
12.5.3	Transformation Between Ontological Items	263
12.6	Summary	264
	References	265
13	OSOAD Case Study	267
13.1	Organization-Oriented System Analysis	267
13.2	Organizational Relationship Model	268
13.3	Organizational Rationale Model	271
13.4	Formal Analysis	273
13.5	Formal Refinement Using Scenario-Based Analysis	275
13.6	Agent Service-Driven Plug and Play	278
	13.6.1 Plug and Play Modeling	278
	13.6.2 Agent Service-Driven Plug and Play	279
	13.6.3 Implementation	281
13.7	M-Space for Macroeconomic Decision Support	282
13.8	Summary	285
	References	286
14	Actionable Knowledge Discovery and Delivery	287
14.1	Introduction	287
14.2	Issues with Existing KDD	288
14.3	Gap Analysis	290
	14.3.1 Gaps Between Delivered and Desired	290
	14.3.2 Aspects for Narrowing Gaps	292
14.4	An AKD Framework	293
	14.4.1 AKD Problem Statement	294
	14.4.2 Actionability Computing	296
	14.4.3 AKD Concept Map	298
	14.4.4 Ubiquitous Intelligence	298
14.5	Deployment	300
	14.5.1 Opportunities	300
	14.5.2 AKD Architectures	302
	14.5.3 AKD Implementation	302
	14.5.4 Knowledge Delivery	305
14.6	An Example	306
14.7	Summary	308
	References	309

15	Learning Complex Behavioral and Social Data	313
15.1	Introduction	313
15.2	Complex Behavioral and Social Problems	314
15.2.1	Behavioral and Social System and Intelligence	314
15.2.2	Complexity of Behavioral and Social Systems	316
15.3	Non-IID Behavioral and Social Problems	317
15.3.1	Coupling	317
15.3.2	Heterogeneity	318
15.4	Issues in Classic Behavioral and Social Learning	320
15.4.1	Classic Behavior Analysis	320
15.4.2	Classic Social Media and Recommendation Systems	321
15.4.3	Classic Social Network Analysis	322
15.5	Non-IIDness Learning	323
15.6	Non-IIDness Learning Case Studies	325
15.6.1	Coupled Behavior Analysis	326
15.6.2	Coupled Item Recommendation	329
15.6.3	Term Coupling-Based Document Analysis	331
15.7	Summary	332
	References	334
16	Opportunities and Prospects	337
16.1	About Open Complex System Studies	337
16.2	About Metasynthetic Computing and Engineering	338
	References	340
	Index	343