Studies in Computational Intelligence

Volume 680

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series "Studies in Computational Intelligence" (SCI) publishes new developments and advances in the various areas of computational intelligence—quickly and with a high quality. The intent is to cover the theory, applications, and design methods of computational intelligence, as embedded in the fields of engineering, computer science, physics and life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in computational intelligence spanning the areas of neural networks, connectionist systems, genetic algorithms, evolutionary computation, artificial intelligence, cellular automata, self-organizing systems, soft computing, fuzzy systems, and hybrid intelligent systems. Of particular value to both the contributors and the readership are the short publication timeframe and the worldwide distribution, which enable both wide and rapid dissemination of research output.

More information about this series at http://www.springer.com/series/7092

Aleksander Byrski · Marek Kisiel-Dorohinicki

Evolutionary Multi-Agent Systems

From Inspirations to Applications

Springer

Aleksander Byrski
Faculty of Computer Science, Electronics and Telecommunications, Department of Computer Science
AGH University of Science and Technology
Kraków
Poland

Marek Kisiel-Dorohinicki
Faculty of Computer Science, Electronics and Telecommunications, Department of Computer Science
AGH University of Science and Technology
Kraków
Poland

ISSN 1860-949X ISSN 1860-9503 (electronic)
Studies in Computational Intelligence
ISBN 978-3-319-51387-4 ISBN 978-3-319-51388-1 (eBook)
DOI 10.1007/978-3-319-51388-1

Library of Congress Control Number: 2016960714

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

To our beloved ones as a proof,
admired ones as a tribute
and adverse ones as a warning.

Preface

For the last 40 years, a growing interest has been observed in the systems where a task to solve is decomposed into smaller parts (subtasks), which are dealt with separately and later they are synthesized into an overall solution. Such an approach may be described as *distributed problem solving*, and is usually easily implemented in parallel environments such as multi-core machines, clusters or grids. It should be noted that multi-agent systems belonging to a popular class of methods in artificial intelligence are an effective implementation of distributed problem solving. Agents are perceived as autonomous beings, which are able to interact with their environment and other agents and bear the features of intelligence. In these systems, a task to solve is usually decomposed into subtasks, which are entrusted to agents. Each agent's goal is to solve its part, and different features of agency affect this process, e.g. autonomy allows for self-adaptation of the agent's strategy.

In 1996, Krzysztof Cetnarowicz proposed an evolutionary multi-agent system (EMAS) dedicated to solving computing problems, with interesting features like distributed selection and lack of global control. Since then the idea of EMAS has been applied to different problems (e.g. single, multi-modal and multi-criteria optimization). This approach still retains high potential possibilities of extension and hybridization (e.g. with cultural or memetic mechanisms) that are researched in Intelligent Information Systems Group at AGH University of Science and Technology in Cracow, Poland.

It is noteworthy that since the inception, EMAS-related research has yielded different modification of this system (utilizing elitist, co-evolutionary or immunological inspirations). Based on these modifications, effective solutions to many difficult problems have been provided such as evolution of neural network architecture, multi-modal optimization and financial optimization to name but a few. EMAS has thus proved to be a versatile optimization mechanism in practical situations.

Multi-agent systems provide a good basis for the development of hybrid search and optimization systems, however it should be noted that in this way, more and more complex computing systems are created. Also, using common sense and remembering *Ockham's razor* rule, one should apply complex search techniques solely to difficult problems. Therefore, metaheuristics, in particular agent-oriented ones, should be treated as the *methods of last resort*, and should not be applied to simple problems.

On the other hand, the need to build complex (hybrid) systems, calls for performing a more in-depth analysis of features of their work. A detailed description of its structure and behavior is required for a full understanding of them, moreover, providing means for stochastic analysis may yield additional, important results such as confirmation if the system works at all (meaning, whether or even when it is able to localize the result).

Several formal models aimed at proving different features of evolutionary metaheuristics have been constructed. One of the first and important models of metaheuristic methods was Michael Vose's model, which proves that for a fixed size population, simple genetic algorithm (SGA) can be modeled by a Markov chain, and after further assumption that the mutation rate is positive, this chain is ergodic. This result formally justifies SGA as a well-defined global optimization algorithm. Other approaches to model evolutionary algorithms to be mentioned are different models for single-population evolutionary algorithms, proposed by Davis, Mahfoud or Rudolph. In particular, Rudolph's model was used to prove the first hitting time for a (1+1) evolution strategy optimizing a convex function. Unfortunately, there is lack of general models, as all those mentioned above are oriented on analysis of particular methods.

In order to successfully build and examine a multi-agent system, making it not only reliable, but also distributed and scalable, flexible and extensible, one must work not only on the implementation itself, but rather try to develop an universal framework, that will be later adapted by the developers creating particular flavor of the system, or even the end user himself (using eg. domain specific languages). Efficient working of such systems applied to solving of complex problem requires both high performance and flexibility connected with possibilities of adapting to dynamically changing needs of the user. Application of intelligent computational techniques supported by novel technologies makes possible adaptation of the structure and organization of the system, creating a new quality in the aspect of services provided and general applicability.

Though the main development of the agent-oriented technology occurred in connection with necessity of making possible exchange of information between cooperating and distributed agents, working in a heterogeneous environment, a significant attention is also drawn to agent-based modeling, connected with simulation-oriented applications, in particular for the process of distributed nature. At the same time, there is a lot to do in the area, which may be generally described as agent-based computing, belonging to computational-intelligence paradigm. In such systems, not the knowledge and planning of a single agent, or more complex communication protocols and interaction schemes are of utmost importance. Rather a holistic sum of single, relatively simple behaviors of individual agents affects the applicability of the whole system. From the implementation point-of-view, the situation is similar to certain simulation models, but in this case the performance becomes an important issue, as single experiments cannot always be repeated because of lack of time or hardware-related constraints.

One should be aware, that since the complexity of both computational-intelligence techniques and agent-based systems is very high, the combination of these two paradigms will pose even higher problems for modeling, design, development, monitoring the experiments and analysis of the results. In the end, design and execution of such systems demands an immense commitment and a large precision during all the stages of their realization. The systems made in *ad hoc* manner quickly become insufficient because of problems in assessment of the obtained results, that may be totally different for small changes introduced in the configuration used.

At the same time due to high complexity of the considered techniques, their applicability becomes justified not earlier as they are applied to task of relatively high complexity. The stochastic computation models used put together high exploration and exploitation capabilities, though the requirements related to the feasibility of the obtained result are satisfied after performing large number of the iterations and repetitions. Moreover, a long user-driven process of trial-and-error is often required, for proper tuning of a large number of their parameters. Performance-related issues very often make the use of classic agent-oriented platform useless in the case of computational intelligence applications.

Over 15 years of experience of the authors in realization of such kind of systems, also because of cooperation and involvement of many persons from different academic institutions have already resulted in preparing of many scientific publications, several doctoral theses (e.g. [91, 39, 262]) and two habilitation dissertations [40, 175], which became a basis for this monograph.

The structure of this monograph is as follows. There are three parts, first one is devoted to the literature review, motivation and definition of the considered systems, also including full formal analysis of EMAS leading to the conclusion that the computing based on this paradigm is formally justified, as such systems are always able to locate the solution to be found. The second part is devoted to design and implementation of the platforms supporting EMAS-like computations, along with the presentation of the AgE platform, that was used during the computations which results are presented in this book. The last part is fully devoted to the experimental results obtained by applying EMAS and some of its modifications to solve discrete and continuous problems, exploring the possibilities of adaptation of particular parameters of the system to solve benchmark problems of varying difficulty.

Chapter 1 presents a systematic state-of-the-art review. It begins with discussion on features of computing systems and their relation to decision support, including identification of difficult problems (so-called "black-box") search problems and justification for the use of complex metaheuristics to solve them. Later, evolutionary and hybrid metaheuristics are reviewed and certain gaps are identified in order to prepare the reader for the next chapters.

Chapter 2 gives a description of agent-based architectures of computing systems is given. Then, a concept of Evolutionary Multi-agent Systems (EMAS) is discussed, including base mechanisms that are used for control and tuning up of its work, along with physical and logical structure of the system that gives the starting point for later architecture-related deliberations. Besides classic EMAS, several of its variations are also presented leveraging different natural inspirations (immunological, co-evolutionary or memetic).

Chapter 3 starts with the formal definition of EMAS and construction of a Markov chain modeling dynamics of EMAS. Then, after necessary assumptions, the ergodic theorem is defined and a full formal proof is given. Later, the same structure of presentation is retained for iEMAS, however, in this case no full formal proof has been constructed, and only necessary conjecture is formulated and the proof outline is described. The chapter is concluded with a short description of actual goals reached in the formal analysis of agent-based metaheuristics.

Chapter 4 builds a technological perspective for the considered class of systems by introducing the most important concepts, ideas and techniques in the field of agent-based systems and component technologies, pointing-out possible relations between these approaches on the development level.

Chapter 5 is opened with the discussion of potential requirements connected with managing of the system, considering its dynamic nature, both because of potential changes of the task to be solved (non-stationary problems), user-related requirements or preferences and the structure of the computing environment. An evaluation of the existing tools was also conducted, in order to check their potential capabilities in realization of agent-based computing systems. Referring to the proposed architectural model and showing the design assumptions *de facto* a new method of implementation of agent-based systems for computational applications was presented.

The most important components of the referential implementation of the agent-based platform and distributed computing environment AgE are described in Chap. 6. Particular attention was given to the previously announced technological requirements and relations to alternative tools described in the literature.

Chapter 7 presents the results of a wide-ranging series of experiments conducted in order to evaluate the efficiency of agent-based metaheuristics, as compared to classical search methods. In the beginning, EMAS is evaluated using selected high-dimensional benchmark functions. After presenting the benchmarks, comparison between EMAS and PEA (parallel evolutionary algorithm) is made, using classical (evolutionary) and memetic versions of these methods. Later, immunological version of EMAS is tested versus the classical EMAS.

The final Chap. 8 focuses on, tuning of selected EMAS parameters is considered. After observing an impact of changing certain EMAS parameters (energy-related and probabilistic), and iEMAS (lymphocyte parameters), the results are summed up, which provides a base for further use in order to adapt these metaheuristics to particular problems.

Kraków, Poland
August 2016

Aleksander Byrski
Marek Kisiel-Dorohinicki

Contents

Part I Concept and Formal Model

1 Contemporary Methods of Computational Intelligence 3
1.1 Computing Systems and Decision Support 4
1.1.1 Formulation of Optimization Problem 4
1.1.2 Decision Support and Computational Intelligence 5
1.1.3 Difficult Search Problems 6
1.1.4 Metaheuristic and Heuristic Search Methods 8
1.2 Evolutionary Metaheuristic Techniques 10
1.2.1 Biological Inspirations. 11
1.2.2 Structure of Evolutionary Algorithm 12
1.2.3 Avoiding the Local Extrema 15
1.2.4 Parallel Evolutionary Algorithms 17
1.3 Sophisticated Population-Based Methods 19
1.3.1 Hybrid Search Methods. 20
1.3.2 Cultural Algorithms. 21
1.3.3 Memetic Algorithms 22
1.3.4 Immunological Techniques 24
1.4 Summary 28

2 Agent-Based Computing 31
2.1 Agency, Architectures, Management and Computing 32
2.1.1 Agents and Multi-agent Systems 32
2.1.2 Architectures of Agent-Based Computing Systems 34
2.1.3 Computing Systems Management by Agent Means 36
2.2 Bringing Together Agent-Based and Evolutionary Computing 37
2.2.1 Non-renewable Resources 38
2.2.2 Integrating Evolutionary Paradigm with Agency 39
2.2.3 Logical and Physical System Structure 40
2.2.4 Representation of the Tasks and Solutions 41

2.3 Evolutionary Multi-agent System . . . 42
2.3.1 Reproduction and Inheritance . . . 42
2.3.2 Selection . . . 43
2.3.3 Computational Features . . . 45
2.3.4 Maintaining Population Diversity . . . 46
2.4 Other Variants of EMAS . . . 48
2.4.1 Memetic EMAS . . . 48
2.4.2 Immunological EMAS . . . 49
2.4.3 Co-evolutionary Multi-agent Systems . . . 51
2.4.4 EMAS in Multi-criteria Optimization . . . 53
2.5 Summary . . . 55

3 Formal Aspects of Agent-Based Metaheuristics . . . 57
3.1 Formal Definition of EMAS . . . 57
3.1.1 EMAS State . . . 59
3.1.2 EMAS Actions . . . 64
3.1.3 EMAS Management . . . 72
3.2 Formal Analysis of EMAS . . . 73
3.2.1 EMAS Dynamics . . . 74
3.2.2 Ergodicity of EMAS . . . 77
3.2.3 Technical Details of EMAS Ergodicity Proof . . . 81
3.3 Formal Perspective on iEMAS . . . 91
3.3.1 iEMAS State . . . 92
3.3.2 iEMAS Management . . . 96
3.3.3 iEMAS Dynamics . . . 98
3.3.4 Ergodicity of iEMAS . . . 101
3.4 Summary . . . 105

Part II Design and Implementation

4 Agents and Components in Software Engineering . . . 109
4.1 Technological Solutions for Agent-Based Systems . . . 109
4.1.1 Agent Platforms . . . 110
4.1.2 Agent-Based Simulation Tools . . . 113
4.2 Component-Based Technologies . . . 115
4.2.1 Components, Dependencies, Contracts . . . 116
4.2.2 Inversion of Control and Dependency Injection . . . 118
4.2.3 Component Environments . . . 118
4.3 Summary . . . 120

5 Towards the Implementation of Agent-Based Computing Systems . . . 123
5.1 Technological Background . . . 123
5.1.1 Referential Solutions . . . 124
5.1.2 A Need for New Computing Platforms . . . 126
5.1.3 Basic Requirements and Assumptions . . . 127
5.2 Agent Design . . . 128
5.2.1 Agents in Object-Oriented Paradigm . . . 128
5.2.2 Component Structure of the Agent Implementation . . . 131
5.3 Computing Environment Architecture . . . 134
5.3.1 Computing Node . . . 134
5.3.2 Distributed Computing . . . 135
5.4 Summary . . . 138

6 AgE Computing Environment . . . 139
6.1 Agent Platform . . . 139
6.1.1 Execution of Agents . . . 141
6.1.2 Life-Cycle Management . . . 143
6.1.3 Communication Facilities . . . 144
6.2 Component Framework . . . 147
6.2.1 Property Mechanism . . . 148
6.2.2 Definitions of Components and Their Dependencies . . . 149
6.2.3 Configuration of the System . . . 150
6.3 Node Architecture . . . 153
6.3.1 Node Services . . . 153
6.3.2 Distributed Environment . . . 155
6.3.3 Migration of the Agents . . . 157
6.4 Summary . . . 157

Part III Experimental Results

7 EMAS in Optimization Problems . . . 161
7.1 Continuous Optimization . . . 161
7.1.1 Benchmarks Problems . . . 161
7.1.2 Classic EMAS and PEA . . . 163
7.1.3 Memetic EMAS and PEA . . . 167
7.1.4 Classic and Immunological EMAS . . . 173
7.2 Discrete Problems . . . 178
7.2.1 Benchmarks Problems . . . 178
7.2.2 Classic EMAS and PEA . . . 180
7.3 Summary . . . 182

8 Tuning of EMAS Parameters . . . 183
8.1 Energy-Related Parameters . . . 184
8.1.1 Energy Exchange Rate . . . 184
8.1.2 Initial Energy Level . . . 185
8.1.3 Minimal Reproduction Energy . . . 186
8.2 Probabilistic Decision Parameters . . . 187
8.2.1 Migration Probability . . . 187
8.2.2 Meeting Probability . . . 187
8.2.3 Immunological Parameters . . . 189
8.3 Summary . . . 193

Final Remarks . . . 195

References . . . 197