Evolutionary Organizational Search (Extended Abstract)

Boyang Li, Han Yu, Zhiqi Shen, Chunyan Miao
Nanyang Technological University
50 Nanyang Avenue, Singapore 639798

(+65) 67906968 byli@ntu.edu.sg
(+65) 67906968 yuhan@pmail.ntu.edu.sg
(+65) 67904386 zqshen@ntu.edu.sg
(+65) 67906197 ascymiao@ntu.edu.sg

ABSTRACT
In this paper, we proposed Evolutionary Organizational Search (EOS), an optimization method for the organizational control of multi-agent systems (MASs) based on genetic programming (GP). EOS adds to the existing armory a metaheuristic extension, which is capable of efficient search and less vulnerable to stalling at local optima than greedy methods due to its stochastic nature. EOS employs a flexible genotype which can be applied to a wide range of tree-shaped organizational forms. EOS also considers special constraints of MASs. A novel mutation operator, the redistribution operator, was proposed. Experiments optimizing an information retrieval system illustrated the adaptation of solutions generated by EOS to environmental changes.

Categories and Subject Descriptors
I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence – *Multiagent systems.* G.2.1 [**Discrete Mathematics**]: Combinatorics – *Combinatorial algorithms.*

General Terms
Algorithms, Performance, Design.

Keywords
Experimental, Systems, Biologically-Inspired Approaches, Organizational Planning, Genetic Programming, Multi-Agent Systems.

1. INTRODUCTION
By specifying important aspects of a multi-agent system such as role assignments, hierarchies of authorities and relationships between agents such as information flow and collaboration, organization may significantly influence the efficiency, adaptability and robustness of MASs [2, 3, 6]. Optimal design of MAS organizations has been an active field of research. In this work, we keep our focus in optimizing organizational control rather than operational control. In other words, we are interested in long-term guidelines such as authorities, role assignments and responsibilities rather than exact algorithmic instructions for specific tasks like communication protocols or social norm enforcement.

A number of methodologies, such as Omni [1], etc., attempt to aid the human designer in organizational optimization. They walk designers through the entire design process and cover both organizational control and operational control, but perform only qualitative analysis and ultimately require human judgments. In contrast, with a focus on organizational control, EOS performs automated quantitative analysis, which may produce convincingly justified results and ease the manual burden.

Existing methods for quantitative optimization of MAS organization include [2], [6], and KB-ORG [5]. These methods differ in their use of domain knowledge and scope of search. In [2, 6], domain knowledge is primarily utilized as explicit or implicit organizational templates which provide holistic views of the MAS. Solutions are searched in the generate-and-evaluate approach. In [2], a general modeling language, the Organizational Design Modeling Language (ODML), is proposed and the template is explicitly constructed in it. It performs exact global searches, employing techniques such as backtracking, and equivalence class. In comparison, KB-ORG [5] represents domain knowledge as multiple small fragments to make a series of locally optimal decisions, and is thus greedy. However, as the optimization complexity has been shown to be NEXP-complete [2], it is generally infeasible to apply exact search methods in optimizations of MAS organizations. On the other hand, we may be interested in the global optimum, which greedy algorithms might not provide in nonlinear and non-monotonic search spaces [2]. As a consequence, it is necessary to develop a heuristic global search method for organization optimization.

In this paper we propose an optimization method called Evolutionary Organizational Search (EOS) based on strongly typed genetic programming (STGP) [4] as an addition to the exact search methods in [1]. EOS optimizes tree-structured organizations and is well suited for hierarchical organizations as well as other tree-like organizational forms such as holarchies and federations. The interested reader is referred to [3] for a survey of MAS paradigms. During the optimization, constraints on resources are taken into account, and special genetic operators are also designed, so as to cope with needs of real-world MAS. Furthermore, due to the stochastic nature of genetic programming, it has the potential to find the global optimum and outperform greedy methods. EOS can be easily implemented as a distributed algorithm, and its execution may be stopped anytime. Hence it is suitable for self-adaptation of multi-agent organizations. While KB-ORG emphasizes a knowledge-based, formal and complete design process and the existing methods in [2] focuses on capturing system performance and building models, EOS fills in the gap in between by proposing a method for efficient search.

In the next section, we describe the representation and genetic operators of EOS. Nevertheless, the space constraint forbids the inclusion of formal definitions and complete pseudo-codes. In

Cite as: Evolutionary Organizational Search, (Extended Abstract), Boyang Li, Han Yu, Zhiqi Shen, Chunyan Miao, *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009),* Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary, pp. 1329–1330

Section 3, we illustrate its adaptability with some experiments. Section 4 draws our conclusion.

2. EVOLUTIONARY ORGANIZATIONAL SEARCH

Employing parse trees as genotypes, genetic programming is a metaheuristic method for evolutionary optimization of programs. STGP adds type consideration to conventional GP. Evolutionary Organizational Search extends STGP to suit the needs of organizational optimization by introducing a modeling method and novel genetic operators.

In genetic programming, each symbol or node is either a terminal or a non-terminal. EOS extends the above two categories to five categories, namely GroupNode, AgentNode, ResourceNode, RoleNode and NullNode. GroupNode and AgentNode are nonterminals. ResourceNode, RoleNode and NullNode are terminals. These categories of nodes provide modeling guidelines, simplify manipulation of the tree structure and ensure its integrity.

A group of agent is represented as a tree/subtree, of which a GroupNode serves as the root. An AgentNode represents an agent which may access or control resources. Thus, a ResourceNode, representing a resource, may be a child of either a GroupNode or an AgentNode. An AgentNode that does not have any ResourceNode children should have a single NullNode child. The RoleNode is used when multiple roles is taken by one agent. It represents a role and points to an AgentNode taking up that role.

In the strongly typed approach, each node is assigned a returned type. Non-terminals can specify types of arguments as a set of allowed returned types for their children. A parent-child relationship can be established only when a matching is found.

EOS supports hard constraints on the number of specific types of nodes. In most applications, computational and other resources are bounded in an MAS. Additionally, we may want to utilize at least a certain number of resources, such as databases in an information retrieval system. In EOS, traditional tree growth, mutation, and crossover operators are modified to account for lower and upper limits on the number of ResourceNode and AgentNode.

A new mutation operator, the redistribution operator, is also proposed to complement the traditional point mutation. In the optimization process, it is often desired to fine-tune the allocation of resources between agents and groups of agents. This operator randomly mutates the combination and groupings of ResourceNode and AgentNode in the tree to accelerate such optimization. Due to space constraints, however, we cannot present detailed algorithms for the genetic operators in this paper.

3. EMPIRICAL STUDIES

In the experiments we optimize the organization of an information retrieval (IR) system. The IR system is hierarchical, consisting of three agent roles, including agents which control databases, aggregators which collect data from lower agents and route messages, and mediators which accept user queries and decide which portion of the system to search. A group of mediators, in charge of aggregators and databases below it, sit at the topmost layer of the system and cooperate to answer user queries. Readers are recommended to read [2] for detailed description of the system and environmental variables.

The system performance is affected by two factors: *recall rate* and *response time*. The recall rate is determined by the relevant data being searched compared to total relevant data in the system. The utility value to be optimized is computed as follows.

$$utility = recall\_rate \times 1000 - response\_time/10$$

Three sets of experiments with different environmental variables were repeated for 50 times. Fifty individuals were evolved for 20 generations. Table 1 shows how the organizational structure of a system managing 15 databases adapted under different environment. Among various organizational features, we are mainly interested in the number of mediators. The results suggest a trade-off between recall rate and response time. When the system was not queried very often (0.05), a single mediator that administered all databases could cope well, but when the query rate was increased to 0.10, multiple mediators that share the workload became necessary. In the third setting, when the system searched only databases under a single mediator, the number of mediators was decreased so as to maintain a good recall rate.

Table 1 Empirical Results

Query Rate	Max Search Set Size	Number of Mediators		Fitness	
		Average	Std Dev	Average	Std Dev
0.05	4	1.00	0.00	851.22	1.47
0.10	4	4.74	0.44	468.17	13.84
0.10	1	2.00	0.00	346.52	4.91

4. CONCLUSIONS

In this paper we proposed a novel framework of evolutionary optimization for agent organizations based on strongly typed GP, which we name EOS. It fills the gap between existing quantitative optimization frameworks by providing a metaheuristic optimization technique. EOS also considers special constraints and requirements of MASs and can be applied to a wide range of tree-shaped organizational forms. Our experiments generated differently structured organizations which adapted to different environments for an information retrieval system.

5. REFERENCES

[1] Dignum, V., Vazquez-Salceda, J. and Dignum, F. Omni: Introducing social structure, norms and ontologies into agent organizations. *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi-Agent Systems* (New York, NY, 2004), 91-102.

[2] Horling, B. *Quantitative Organizational Modeling and Design for Multi-Agent Systems*. PhD Thesis, University of Massachusetts at Amherst, 2006.

[3] Horling, B. and Lesser, V. A survey of multi-agent organizational paradigms. *Knowl. Eng. Rev.*, 19, 4 (Dec. 2004), 281-316.

[4] Montana, D. J. Strongly typed genetic programming. *Evol. Comput.*, 3, 2 (Summer 1995), 199-230.

[5] Sims, M., Corkill, D. and Lesser, V. *Knowledgeable Automated Organization Design for Multi-Agent Systems*. Technical Report 07-43, University of Massachusetts, Amherst, MA, 2007.

[6] So, Y.-p. and Durfee, E. H. Designing organizations for computational agents. In *Simulating Organizations*. AAAI Press/ MIT Press, 47-64, 1998.