

PROFILE-BASED ARCHITECTURE OF EVOLUTIONARY MAS

MAREK KISIEL-DOROHINICKI, GRZEGORZ DOBROWOLSKI
AND EDWARD NAWARECKI

*Department of Computer Science,
University of Mining and Metallurgy,
Mickiewicza 30, 30-059 Cracow, Poland
{doroh, grzela, nawar}@agh.edu.pl*

(Received 18 January 2002; revised manuscript received 30 July 2002)

Abstract: A sub-type of multi-agent systems (MAS) called evolutionary ones (EMAS), special in its features and field of application, needs a dedicated architecture that would be more adequate and easier in design and implementation. The proposed architecture uses the notion of a profile which models strategies and goals of an agent with respect to an aspect of its operation. To make a decision, an agent is equipped with an algorithm that coordinates premises determined in its profiles. The paper presents main ideas of the architecture illustrated with an actual realisation of an EMAS solving the multi-objective optimisation problem.

Keywords: decentralized artificial intelligence, multi-agent systems, evolutionary computation

1. Introduction

Evolutionary computation – a heuristic problem-solving approach based on models of organic evolution – has been successfully used in solving various problems for over 40 years. While different evolutionary algorithms use a specific representation, variation operators and selection scheme, they all employ a similar model of evolution: working on a given number of data structures (population) and repeating the same cycle of processing (generation), which consists of the selection of parents and production of offspring using genetic operators. However, this model of evolution is much simplified and lacks many important phenomena observed in organic evolution [1]: dynamically changing environmental conditions, many criteria under consideration, neither global knowledge nor generational synchronisation assumed, co-evolution of species, evolving genotype-phenotype mapping, *etc.* That is why many variations of classical evolutionary algorithms were proposed, introducing *e.g.* population structure (like in parallel evolutionary algorithms) or specialised selection mechanisms (like fitness sharing).

The idea of decentralised evolutionary computation realised as an *evolutionary multi-agent system* (EMAS) may be considered as an extension of the classical

evolutionary computation covering various specialised techniques in one coherent model. The key idea of EMAS is the incorporation of evolutionary processes into a multi-agent system (MAS) at the population level. This means that, besides interaction mechanisms typical for MAS (such as communication), agents are able to reproduce (generate new agents) and may die (be eliminated from the system). A decisive factor of an agent's activity is its fitness, expressed by amount of the possessed non-renewable resource called life energy. Selection is made in such a way that agents with high energy are more likely to reproduce, while low energy increases possibility of death.

In the paper, a detailed model of an evolutionary multi-agent system is presented. Since existing formalisms for MAS are not easily applicable to this kind of agent systems, a simple yet extensible model of MAS based on M-Agent architecture is first proposed (for further reference see *e.g.* [2]). This constitutes a base for a description of evolutionary phenomena at the level of a single agent and its internal architecture. The described concepts are briefly illustrated with possible application areas of EMAS.

2. Agent systems

During the last decade, the idea of an intelligent autonomous agent has attracted more and more interest, both in the academic community and industry. A constantly increasing number of computer systems is being analysed and designed in terms of agents. Agents play a key role in the integration of AI sub-disciplines and are often necessary to design and build modern intelligent systems. In fact, agent technology is already used in various domains, providing concepts and tools for the development of complex, distributed and decentralised systems [3].

Yet, agent technology is still so immature that there is even no consensus on what an agent really is [4]. Researchers have offered a variety of definitions, which range from simple to lengthy and demanding ones. By now, this should not be considered a problem since "the notion of an agent is meant to be a tool for analyzing systems, not an absolute characterization that divides the world into agents and non-agents" [5]. As a base for further considerations, we will adopt the definition given by Franklin and Graesser [4], which says that "an autonomous agent is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to affect what it senses in the future".

In contrast to the notion of *agent*, the terms *agent-based system* and *multi-agent system* have well-understood meanings and their definitions can be easily formulated. Nevertheless, one ought to remember that the problems discussed above remain in the background all the time.

By an agent-based system, we mean a system in which the key abstraction used is that of an agent. In principle, an agent-based system might be conceptualised in terms of agents, but implemented without any software structures corresponding to agents at all.

There are cases in which a single agent solution is appropriate, *e.g.* expert assistants act to help users attempting to carry out some task in a computer system. A multi-agent system is rather a collection of autonomous agents aiming at solving

a given problem. Of course, several explanations should be added to make this definition usable:

- Possibly, agents exist prior to the problem to be solved.
- Solving the problem remains beyond the individual capabilities or knowledge of each agent.
- Each agent can gain only incomplete information about the problem being solved (data is decentralized).
- Agents act (computation is done) asynchronously.
- There is no global system control.
- Agents may be heterogeneous in nature.

Multi-agent systems are ideally suited to represent problems that have multiple problem solving methods, multiple perspectives and/or multiple problem solving entities [3].

3. Profile-based architecture of MAS

A multi-agent system (see Figure 1) consists of a set of *agents* ($ag \in Ag$) and some *environment* (*env*) they live in:

$$MAS \equiv \langle Ag, env \rangle. \quad (1)$$

The environment may have spatial structure and contain some *information* and/or *resources*, which may be observed by agents:

$$env \equiv \langle Res, Inf, sp \rangle, \quad (2)$$

where *Res* – (global) resources, *Inf* – (global) information, *sp* – the space, *i.e.* locations of agents and local information or resources (most often represented as a graph).

The functionality of each agent is defined by a set of *actions* ($act \in Act$) it is able to perform. Its internal architecture is described in terms of *profiles* ($prf \in Prf$):

$$ag \equiv \langle Act, Prf \rangle. \quad (3)$$

An action is an atomic (indivisible) activity which may be executed by an agent in the system.

Each profile defines the state of an agent from the point of view of a particular aspect of its functionality. The profile may concern some resource possessed by the agent (“physical” or “energetic” profiles):

$$prf^{res} \equiv \langle res, St, Gl \rangle, \quad (4)$$

where *res* – amount of the possessed resource, *St* – set of *strategies* related to this resource, *Gl* – set of *goals* related to this resource.

The profile may also be dedicated to modelling (a part of) the environment and/or (some features of) other agents (“information” or “intellectual” profiles):

$$prf^{inf} \equiv \langle mdl, St, Gl \rangle, \quad (5)$$

where *mdl* – a piece of information representing the agent’s knowledge about the world it lives in (the model of the world), *St* – set of *strategies* related to this model, *Gl* – set of *goals* related to this model.

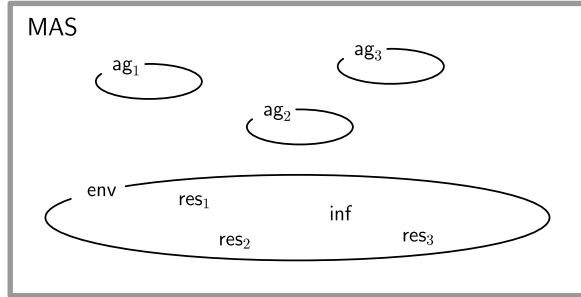


Figure 1. Simplified schema of a multi-agent system according to the proposed model

The model is constructed by an agent using the information acquired via observation of its neighbourhood or from other agents via communication. Of course, this information may (in fact, must) be incomplete and uncertain.

In both cases, St denotes a set of *strategies* ($st \in St$) describing how each action is related to a particular profile. Thus, the strategy st which describes the action act in a physical profile may be defined as:

$$st : res \rightarrow res' \quad (6)$$

and in an intellectual profile, as:

$$st : mdl \rightarrow mdl'. \quad (7)$$

Strategies represent the agent's expectations of the action results. The actual effects of performed actions may differ from these expectations, and this difference may drive a learning process of an agent.

A set of *goals* ($gl \in Gl$) specifies the agent's needs with respect to the resource or model and thus forms a base for a decision-making process. *Active goals* indicate the desired direction of changes, and *conservative goals* define the boundary conditions of amount of the possessed resource, or a state of the model from the point of view of the particular profile.

In this framework the general scheme of MAS operation is that each agent observes (some part of) the system, builds its internal model(s), and acts on (maybe a closer part of) the system according to its goals defined, spending or gaining some resources.

4. Agent decision making process

In the particular case, decision making means the selection of a strategy to realise, and then an action(s) to be performed. Internal architecture of an agent does not enforce specific rules of decision making. Furthermore, without stronger assumptions, this problem is ambiguous because of many profiles and thus various goals to be achieved by an agent at the same time. The most important to achieve seems the selection of an active goal, for which such a strategy exists that actions to be performed do not violate passive goals of remaining profiles.

The proposed model of decision making is related to the concept of a layered agent architecture [6] and assumes some order in the set of profiles $Prf \equiv (Prf, \prec)$, which allows for the definite selection of an action to perform. This order defines

priorities of active goals, as well as the direction of the search for appropriate strategy and its verification by passive goals. On this assumption, the decision making process consists of three stages:

1. selection of an (next) active goal of the lowest priority,
2. search for a strategy which satisfies the selected goal,
3. verification of selected strategy by passive goals of the remaining profiles.

When any stage fails, the process returns to the previous stage and looks for the next element to consider. When stage 1 fails an agent remains idle, *i.e.* performs no actions.

This procedure is illustrated by a simple example in Figure 2: (a) selection of an active goal (stage 1), (b) search for a strategy (stage 2), (c) verification of an action (stage 3), (d) action verification failed (return to stage 2), (e) repeated search for a strategy (stage 2), (f) action successfully verified (stage 3).

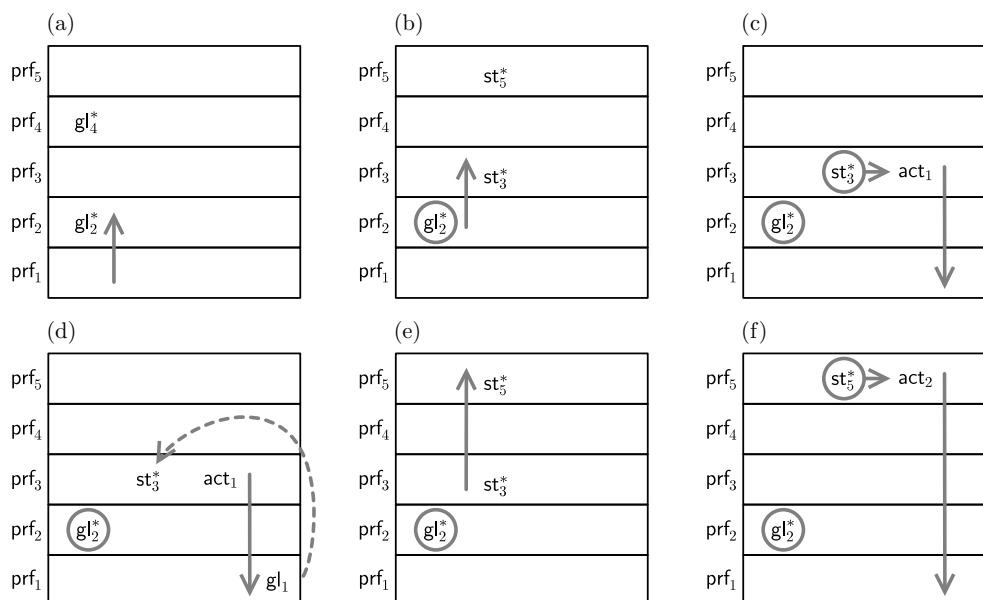


Figure 2. An example of agent's decision making

5. Modelling evolution in MAS

Following neodarwinian paradigms, two main components of the process of evolution are *inheritance* (with random changes of genetic information by means of mutation and recombination) and *selection*. They are realised by the phenomena of death and reproduction, which may be easily modelled as actions executed by agents:

- the action of *death* results in the elimination of an agent from the system,
- the action of *reproduction* is simply the production of a new agent by its parent(s).

Inheritance is to be accomplished by an appropriate definition of reproduction, which is similar to classical evolutionary algorithms. The set of parameters describing

core properties of an agent (genotype) is inherited from its parent(s) with the use of mutation and recombination. Besides, an agent may possess some knowledge acquired during its life, which is not inherited. Both the inherited and acquired information determines the behaviour of an agent in the system (phenotype).

Selection is the most important and most difficult element of the model of evolution employed in EMAS. This is due to the assumed lack of global knowledge (which makes it impossible to evaluate all individuals at the same time) and autonomy of agents (due to which reproduction is achieved asynchronously). In such a situation, selection mechanisms known from classical evolutionary computation cannot be used. The proposed principle of selection corresponds to its natural prototype and is based on the existence of a non-renewable resource called *life energy*. The energy is gained and lost when agents execute actions in the environment. Increase in energy is a reward for “good” behaviour of an agent, decrease – a penalty for “bad” behaviour. Which behaviour is considered “good” or “bad” depends on the particular problem to be solved. At the same time, the level of energy determines the actions that an agent is able to execute. In particular, low energy level should increase possibility of death and high energy level should increase possibility of reproduction.

A more precise description of this technique and its possible applications may be found in [7, 8] and other. In short, EMAS should make the following possible:

- Local selection allows for intensive exploration of the search space, which is similar to parallel evolutionary algorithms.
- The way phenotype (behaviour of an agent) is developed from genotype (inherited information) depends on its interaction with the environment.
- Self-adaptation of population size is possible when appropriate selection mechanisms are used.

Furthermore, explicitly defined living space facilitates implementation in a distributed computational environment.

6. Evolutionary profiles of an agent

To provide a complete description of EMAS in terms of the proposed agent architecture, only a few details reflecting evolutionary nature of the system should be completed, *viz.* mechanisms of selection and reproduction.

As it has already been said in the preceding section, selection in EMAS is based on specific mechanisms, which are realised in *energetic profile* (prf^{eng}) consisting of:

- resource eng – life energy,
- the goal to keep the level of energy above minimal value eng_{\min} ,
- strategies describing all agent’s actions in terms of energy gain and loss, particularly the action of death:

$$\text{st}^{\text{die}} : \text{eng} \rightarrow \text{eng}_{\text{ind}}, \quad (8)$$

which is understood in this profile as a change of the state of life energy to indefinite level $\text{eng}_{\text{ind}} > \text{eng}_{\min}$,

and thus may be described as:

$$\text{prf}^{\text{eng}} = \langle \text{eng}, \text{St}^{\text{eng}} = \{\text{st}^{\text{die}}, \dots\}, \text{Gl}^{\text{eng}} = \{\text{eng} > \text{eng}_{\text{min}}\} \rangle. \quad (9)$$

As long as the level of life energy is above eng_{min} , the goal of an energetic profile is conservative and blocks the realisation of actions which may decrease the amount of eng below this limit. When the energetic state drops below eng_{min} the goal of energetic profile becomes active and triggers the strategy of death.

The agent's striving for reproduction is modelled by *reproductive profile* (prf^{rp}), which consists of:

- resource hr , which determines the agent's ability to reproduce,
- strategy describing the action of reproduction as reducing the level of hr to its minimal value (hr_{min}):

$$\text{st}^{\text{rp}} : \text{hr} \rightarrow \text{hr}_{\text{min}} \quad (10)$$

and, possibly, other strategies related to resource hr ,

- the goal to keep the level of hr below the maximal value $\text{hr}_{\text{max}} > \text{hr}_{\text{min}}$,

and thus may be described as:

$$\text{prf}^{\text{rp}} = \langle \text{hr}, \text{St}^{\text{rp}} = \{\text{st}^{\text{rp}}, \dots\}, \text{Gl}^{\text{rp}} = \{\text{hr} < \text{hr}_{\text{max}}\} \rangle. \quad (11)$$

The amount of resource hr may increase (or decrease) depending on the situation of an agent, *i.e.* its age, interactions with the environment and other agents, *etc.* When it reaches the level of hr_{max} an agent tries to reproduce, expecting that it should lower the level of hr . The reproduction is successful if the state of an agent (*e.g.* amount of life energy) and its neighbourhood allows for the generation of a new agent.

In accordance with (3), an evolving agent is thus described as:

$$\text{ag} = \langle \text{Act} = \{\text{die}, \text{rp}, \dots\}, \text{Prf} = (\text{prf}^{\text{eng}}, \text{prf}^{\text{rp}}, \dots) \rangle. \quad (12)$$

What lacks here is a profile (or profiles) reflecting the problem, which is to be treated (solved) by EMAS, and actions reflecting the solving process. These elements cannot be specified here because they are closely related to a particular application domain. Such specific profile and actions, dedicated to multiobjective optimisation problems, are described in the following section.

7. Applications of EMAS

The proposed model of EMAS was successfully used as a base for a number of applications. Application areas range from typical optimisation problems to data classification and time-series prediction. As an example, a short description of an evolutionary multi-agent system for multicriteria optimisation [8] is presented below.

The particular EMAS should search for a set of points which constitute the approximation of the Pareto frontier for a given multicriteria optimisation problem. In this case, the population of agents represents feasible solutions to the problem defined by a system of objective functions. The energetic reward/punishment mechanism should prefer non-dominated agents. This is done via *domination energy transfer principle* (in short: *domination principle*) forcing dominated agents to give a fixed amount of their energy to the encountered dominants. It may happen when two agents

inhabiting one place communicate with each other and obtain information about their quality with respect to each objective function.

According to (12), each agent in the system may be described as:

$$\text{ag} = \langle \text{Act} = \{\text{die}, \text{rp}, \text{ask}, \text{td}, \text{rd}\}, \text{Prf} = (\text{prf}^{\text{eng}}, \text{prf}^{\text{rp}}, \text{prf}^{\text{opt}}) \rangle. \quad (13)$$

The *domination principle* is realised by three specific actions: **ask** (ask for information about the quality of solution represented by another agent), **td** (transmit domination energy) and **rd** (receive domination energy) described in *optimisation profile* (prf^{opt}). This profile also contains the information about the solution represented by the agent, which is inherited during reproduction. In fact, this is the only component of the agent's genotype and thus a crucial element of the whole process.

The flow of energy connected with *domination principle* causes dominating agents to be more likely to reproduce, whereas dominated agents – likely to die. This way, in successive generations, non-dominated agents should make up better approximations of the Pareto frontier.

8. Instead of conclusion

In the paper, a brief characteristics of an evolutionary multi-agent system is presented. It gives assumptions for the newly proposed so-called *profile-based* architecture, which facilitates design and implementation of systems of this kind. The presentation is supported by a formal model of EMAS. The main elements of the architecture, *i.e.* a profile and decision making algorithm, are described. The reader can also see how the architecture is applied in the construction of an EMAS solving the multi-objective optimisation problem. The exemplary system has been already implemented and tested on several problems.

As the experimental results have demonstrated the usefulness of the proposed architecture, future research should concentrate on further theoretical analysis of EMAS described in terms of profiles and, especially, modelling different interrelations between profiles that will require development of the decision making algorithm. Refining the architecture is also expected, based on analysis of the design and implementation processes of other EMAS to be applied to soft computing problems. It can even lead to creation of rules of specification for such systems.

References

- [1] Bäck T, Hammel U and Schwefel H-P 1997 *IEEE Trans. on Evolutionary Computation* **1** (1) 11
- [2] Cetnarowicz E, Nawarecki E and Cetnarowicz K 1997 *Proc. MCPL'97, IFAC/IFIP Conf.*, Campinas, Brasil, pp. 399–404
- [3] Jennings N R, Sycara K and Wooldridge M 1998 *J. Autonomous Agents and Multi-Agent Systems* **1** (1) 7
- [4] Franklin S and Graesser A 1997 *Intelligent Agents III* (Müller J P, Wooldridge M and Jennings N R, Eds.), Lecture Notes in Artificial Intelligence **1193** 21
- [5] Russell S J and Norvig P 1995 *Artificial Intelligence: A Modern Approach*, Prentice Hall
- [6] Müller J P 1996 *The Design of Intelligent Agents: A Layered Approach*, Lecture Notes in Artificial Intelligence, Vol. 1177
- [7] Cetnarowicz K, Kisiel-Dorohinicki M and Nawarecki E 1996 *Proc. 2nd Int. Conf. on Multi-Agent Systems, ICMA'S'96* (Tokoro M, Ed.), AAAI Press, pp. 383–384
- [8] Dobrowolski G, Kisiel-Dorohinicki M and Nawarecki E 2001 *Proc. IASTED Int. Symp.: Applied Informatics* (Hamza M, Ed.), IASTED/ACTA Press, pp. 295–300