Agent-Oriented Model of Simulated Evolution*

Marek Kisiel-Dorohinicki

Department of Computer Science
University of Mining and Metallurgy, Kraków, Poland
doroh@agh.edu.pl

Abstract. The paper deals with a specific class of multi-agent systems, in principle similar to evolutionary algorithms, but utilising a more complex, since decentralised, model of evolution. The proposed layered architecture uses the notion of a profile that models strategies and goals of an agent with respect to some aspect of its operation. The paper presents main ideas of the architecture illustrated by a concrete realisation that is an evolutionary multi-agent system solving a generic optimisation problem.

1 Introduction

During the last years the idea of an intelligent/autonomous software agent gains more and more applications in various domains. Agent technology provides concepts and tools for development of complex, distributed and decentralised systems [6]. Apparently agents play also key role in integration of AI sub-disciplines, which is often necessary to design and build modern intelligent systems.

Still the literature offers a variety of agent definitions, which range from very simple to lengthy and demanding. In fact this should not be considered a problem since "the notion of an agent is meant to be a tool for analysing systems, not an absolute characterization that divides the world into agents and non-agents" [8]. Indeed the term *multi-agent system* has a well-understood meaning and a corresponding definition can be easily formulated: a multi-agent system (MAS) is simply a collection of agents aiming at solving a given problem. Since usually solving the problem stays beyond the individual capabilities or knowledge of each single agent, the key concept here are intelligent interactions (coordination, cooperation, negotiation). Thus multi-agent systems are ideally suited to representing problems that have multiple problem solving methods, multiple perspectives and/or multiple problem solving entities [6].

Of course a mutli-agent system may be implemented without any software structures corresponding to agents at all. This often happens for simulation systems, where the introduction of agents facilitates modelling of complex phenomena – natural, social, etc. In such cases agents constitute building blocks of the simulation model, which may or may not be implemented with the use of agent

* This work was partially sponsored by State Committee for Scientific Research (KBN) grant no. 4 T11C 027 22.

W.I. Grosky and F. Plášil (Eds.): SOFSEM 2002, LNCS 2540, pp. 253–261, 2002.

technology. An *evolutionary multi-agent system* (EMAS) has analogous characteristics, even though it is rather a computationally intelligent system that may be considered as an extension to classical evolutionary algorithms. The key idea of EMAS is the incorporation of evolutionary processes into a multi-agent system (MAS) at a population level. It 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 possessed non-renewable resource called *life energy*. Selection is realised in such a way that agents with high energy are more likely to reproduce, while low energy increases possibility of death.

Although 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, the model of evolution employed by most evolutionary algorithms is much simplified and lacks many important features observed in organic evolution. This includes dynamically changing environmental conditions, many criteria in consideration, neither global knowledge nor generational synchronisation assumed, co-evolution of species, evolving genotype-fenotype mapping, etc. [1]. That is why many variations of classical evolutionary algorithms were proposed, introducing e.g. some population structure (in parallel evolutionary algorithms) or specialised selection mechanisms (like fitness sharing). The main advantage of EMAS is that it covers various specialised techniques in one coherent model.

Yet EMAS is a very specific, due to its features and fields of application, sub-type of multi-agent systems and thus needs special architecture that would be more adequate and easier for design and implementation. Since existing formalisms for MAS may not be easily applied to this kind of agent systems a simple yet extensible model of MAS based on M-Agent architecture is first proposed in the paper (for further reference see e. g. [2]). This constitutes a base for a description of evolutionary phenomena at a level of a single agent and its internal architecture. The described concepts are shortly illustrated by an application of EMAS in the field of numerical optimisation.

2 General Model of MAS

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

$$MAS \equiv \langle Ag, env \rangle \, . \tag{1}$$

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

$$env \equiv \langle Res, Inf, sp \rangle \tag{2}$$

where Res and Inf represent global (available in the whole system) resources and information, respectively. At the same time sp represents all features of EMAS related to the existence of some space (if any) including:

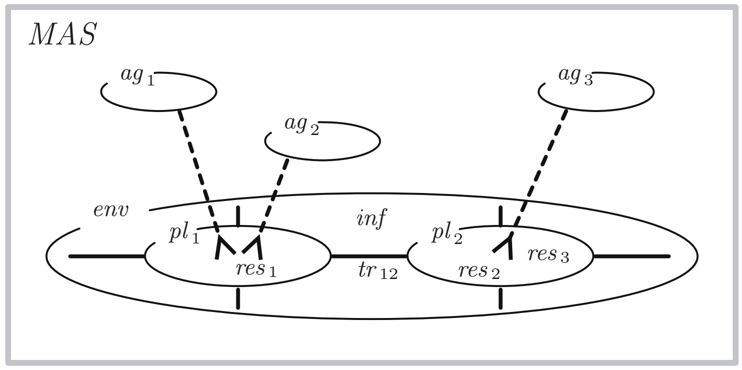

Fig. 1. General structure of a multi-agent system according to the proposed model

- possible locations of agents and local information or resources (topology),
- actual agents' positions,
- information or resources available in specific regions of the space,
- range of observation and/or activity of agents.

The space is most often represented as a graph (Fig. 1), and thus may be described as:

$$sp \equiv \langle Pl, Tr, Loc \rangle \tag{3}$$

where: Pl – set of possible locations: $pl \in Pl$ (nodes of the graph),
Tr – relation of direct neighbourhood between locations: $Tr \subset Pl \times Pl$ (edges of the graph),
Loc – relation representing positions of agents: $Loc\colon Ag \to Pl$.

Each location $pl \in Pl$ may be described in terms of local (available in this location) resources and information:

$$pl \equiv \langle Res^{pl}, Inf^{pl} \rangle\,. \tag{4}$$

The state of local resources Res^{pl} and information Inf^{pl} may be observed and/or changed only by agents, which remain close enough (this should be defined by a particular application) to the location pl.

3 Profile-Based Architecture of an Agent

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* ($prf \in Prf$):

$$ag \equiv \langle Act, Prf \rangle\,. \tag{5}$$

The action is an atomic (indivisible) activity, which may be executed by the 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):

$$prf^{res} \equiv \langle res, St, Gl \rangle \tag{6}$$

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):

$$prf^{inf} \equiv \langle mdl, St, Gl \rangle \tag{7}$$

where: mdl – 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.

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 strategy st, which describes action act, in physical profile may be defined as:

$$st : res \rightarrow res' \tag{8a}$$

and in intelectual profile:

$$st : mdl \rightarrow mdl' \,. \tag{8b}$$

Strategies represent an agent's expectations of the action results, the real effects of the performed action 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 concerning the possessed resource or the state of the model from the point of view of the particular profile.

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

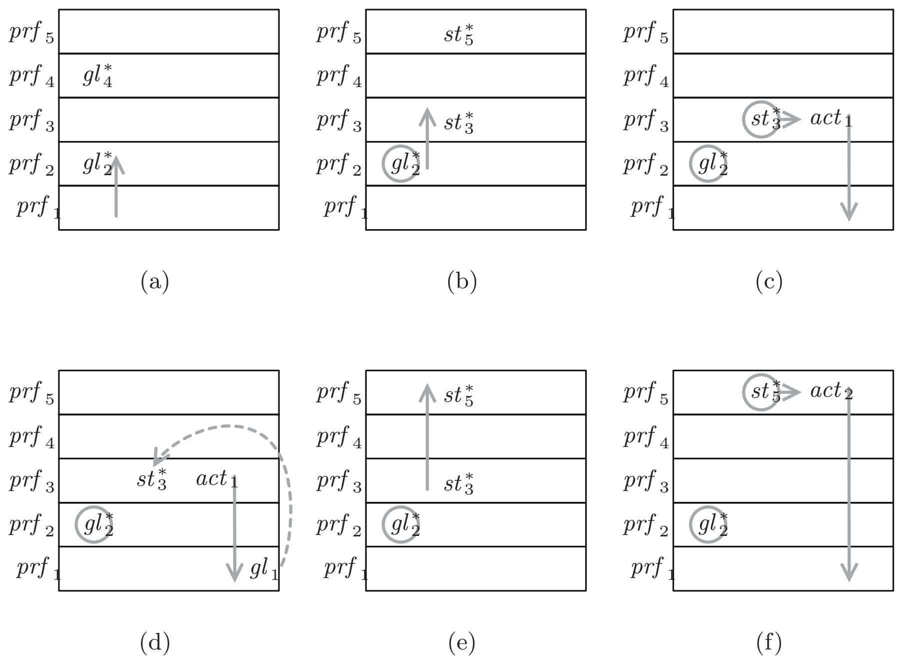

Fig. 2. An example of agent's decision making

In the particular case decision making means selection of the strategy to be realised and then the action(s) to be performed. The internal architecture of an agent does not enforce the specific rules of decision making. What is more, 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 thing to be established 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 [7] and assumes some order in the set of profiles $Prf \equiv (Prf, \prec)$, which allows for definite selection of the action to perform. This order defines priorities of active goals, as well as the direction of search for appropriate strategy and its verification by passive goals. Based on this assumption the decision making process consists of three stages:

1. selection of the (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 remaining profiles.

When any stage fails, the process returns to the previous stage looking 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 Fig. 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).

4 Modelling Phenomena of Evolution

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:

- action of *death* results in the elimination of an agent from the system,
- action of *reproduction* is simply the production of a new agent from 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 assumed lack of global knowledge (which makes it impossible to evaluate all individuals at the same time) and autonomy of agents (which causes that 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 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 actions 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.

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. These are the mechanisms of selection and reproduction described in *energetic* and *reproductive* profiles.

As it was already announced, selection in EMAS is based on specific mechanisms, which are mostly driven by an *energetic profile* (prf^{eng}) consisting of:

- resource eng – life energy,
- 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:

$$st^{die}\colon eng \;\rightarrow\; eng_{ind} \tag{9}$$

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

and thus may be described as:

$$prf^{eng} = \langle eng, St^{eng} = \{st^{die}, \ldots\}, Gl^{eng} = \{eng > eng_{min}\}\rangle\,. \tag{10}$$

As long as the level of life energy is above eng_{min} the goal of 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 energetic profile becomes active and triggers the strategy of death.

The agent's striving for reproduction is modelled by a *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}):

$$st^{rp}\colon hr \;\rightarrow\; hr_{min} \tag{11}$$

  and maybe other strategies related to resource hr,
- goal to keep the level of hr below the maximal value $hr_{max} > hr_{min}$,

and thus may be described as:

$$prf^{rp} = \langle hr, St^{rp} = \{st^{rp}, \ldots\}, Gl^{rp} = \{hr < hr_{max}\}\rangle\,. \tag{12}$$

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

Conforming to (5) an evolving agent is thus described as:

$$ag = \langle Act = \{die, rp, \ldots\}, Prf = (prf^{eng}, prf^{rp}, \ldots)\rangle\,. \tag{13}$$

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 for optimisation problems are shortly described in the next section.

5 EMAS for Numerical Optimisation

As an illustration to above considerations the simplest possible practical application of the proposed architecture will be presented: an evolutionary multi-agent system for numerical optimisation (cf. [4]).

In the particular EMAS the population of agents represents feasible solutions to the problem defined by (a system of) objective function(s). The key issue here is the design of energetic reward/punishment mechanism, which should prefer better (with respect to the defined criterion or criteria) agents. This is done via *energy transfer principle* forcing worse agents to give a fixed amount of their energy to the encountered better agents. This may happen, when two agents inhabiting one place communicate with each other and obtain information about their quality with respect to known objective function(s).

According to (13) each agent in the system may be described as:

$$ag = \langle Act = \{die, rp, ask, td, rd\}, Prf = (prf^{eng}, prf^{rp}, prf^{opt})\rangle \tag{14}$$

where ask, td, rd denote actions realising the *energy transfer principle* and prf^{opt} represents a dedicated *optimisation profile*.

An *optimisation profile* is a problem-dependent profile, which encapsulates information about the solution represented by an agent, which is inherited during reproduction. In fact this is the only component of an agent's genotype and thus the crucial element of the whole process. This profile also contains strategies describing actions of *energy transfer principle*: ask (ask for information about the quality of solution represented by another agent), td (transmit energy), and rd (receive energy). Of course the last two actions must be also described in an energetic profile.

The flow of energy connected with the *transfer principle* causes that better agents are more likely to reproduce, whereas worse ones are more likely to die. This way, in successive generations, the agents should represent better approximations of the solution to the problem.

6 Concluding Remarks

The proposed model of EMAS was successfully used as a base for a number for applications. Application areas range from numerical optimisation to hybrid soft computing systems involving fuzzy systems (e.g. data classification) and neural networks (e.g. time-series prediction). Concerning computational systems, EMAS enables the following:

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

What is more, explicitly defined living space facilitates implementation in a distributed computational environment.

As the experimental results show the usefulness of the proposed model, the future research should lead to refining of the architecture based on analysis of the design and implementation process of EMAS applications in a variety of soft computing problems.

References

[1] T. Bäck, U. Hammel, and H.-P. Schwefel. Evolutionary computation: Comments on the history and current state. *IEEE Trans. on Evolutionary Computation*, 1(1), 1997. 254

[2] E. Cetnarowicz, E. Nawarecki, and K. Cetnarowicz. Agent oriented technology of decentralized systems based on the M-Agent architecture. In *Proc. of the MCPL'97, IFAC/IFIP Conference*, 1997. 254

[3] K. Cetnarowicz, M. Kisiel-Dorohinicki, and E. Nawarecki. The application of evolution process in multi-agent world (MAW) to the prediction system. In M. Tokoro, editor, *Proc. of the 2nd Int. Conf. on Multi-Agent Systems (ICMAS'96)*. AAAI Press, 1996.

[4] G. Dobrowolski, M. Kisiel-Dorohinicki, and E. Nawarecki. Evolutionary multi-agent system in multiobjective optimisation. In M. Hamza, editor, *Proc. of the IASTED Int. Symp.: Applied Informatics*. IASTED/ACTA Press, 2001. 260

[5] S. Franklin and A. Graesser. Is it an agent, or just a program?: A taxonomy for autonomous agents. In J. P. Müller, M. Wooldridge, and N. R. Jennings, editors, *Intelligent Agents III*, volume 1193 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 1997.

[6] N. R. Jennings, K. Sycara, and M. Wooldridge. A roadmap of agent research and development. *Journal of Autonomous Agents and Multi-Agent Systems*, 1(1), 1998. 253

[7] J. P. Müller. *The Design of Intelligent Agents: A Layered Approach*, volume 1177 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 1996. 257

[8] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995. 253