Cycle of the needs satisfaction, and information support of the society development simulation system

Cite as: AIP Conference Proceedings **1836**, 020039 (2017); https://doi.org/10.1063/1.4981979
Published Online: 05 June 2017

B. I. Klebanov, and T. V. Antropov

Cycle of the needs satisfaction, and information support of the society development simulation system

Klebanov B.I.[1, a)] and Antropov T.V.[1, b)]

[1] *Ural Federal University n.a. the first President of Russia B.N. Yeltsin, Yekaterinburg, Russia (620002, Yekaterinburg, 19 Mira Street)*

[a)]Corresponding author: kbi11@yandex.ru
[b)]timofey_tv@mail.ru

Abstract. The report focuses on some of the key points of constructing a model of an artificial society. It is based on the processes of emergence and implementation of the needs of specific agents, depending on the internal processes in the agents, and external environmental factors. Unlike in other approaches, the basis of the modeled system agents' behavior is the concept of needs as the necessity of implementation of the agents' transition from one state to another. This article presents an algorithm of the needs satisfaction of the agent, starting with the event that caused the need for, and ending its satisfaction, or the message that it is impossible to satisfy. The basis of objective knowledge in the system are a model of active and passive agents, as well as recipes meet the needs of the active agents and the relationships between them.

INTRODUCTION

A multi-agent model, developed by us, can be used to analyze the evolution of society as a whole. The model is represented by the physical environment and the different types objects including active agents with the needs and wishes to satisfy them.

This article presents an algorithm of the needs satisfaction of the agent, starting with the event that caused the need for, and ending its satisfaction, or the message that it is impossible to satisfy.

Classical BDI architecture, presented by the authors Rao and Georgeff[1], suggests that the agent reasoning performed using output mechanisms on the basis of the agent mental concepts, that submitted by some structures of knowledge. However, it has a big problem of the actualization and strategies selection, and high computer time resources costs in the planning. It plays an important role if the agent has limited resources for solving complex problems. In EBDI agent architecture developed by the authors[2], [3], includes the emotions. It involves training the agent, but is devoid of the same characteristics as the classic BDI agent model. Most often constructed models are aimed at solving the problems of society such aspects as logistics, transportation[4]. A more detailed description of the mathematical model presented in the paper [5].

Applied Mathematics and Computer Science
AIP Conf. Proc. 1836, 020039-1–020039-6; doi: 10.1063/1.4981979
Published by AIP Publishing. 978-0-7354-1506-5/$30.00

SYSTEM

Information support of the system

Information support of the system includes a global knowledge base (GKB) and a global database (GDB). Global knowledge base includes a set of entities and relationships between them, storing domain knowledge.

Knowledge of the system described by a set of typed entities:

- Recipes - show a set of activities that can be performed on any type of object in order to satisfy a given need.
- Objects
  - Passive - models of natural or man-made objects that change or do not change over time under the influence of external or internal factors, but do not have needs and do not fight for their existence.
  - Active - a model of autonomous units capable of autonomous decision-making. In the models of active agents should take into account the processes of formation and realization of their needs.
- Relationship - the models describe the possible types of bilateral and multilateral relations between the subjects.

System objects have a hierarchical structure and describes the types and the set of properties. Various types of objects are described in the object types directory, and within a given subject area are divided into three types: passive objects, agents and objects - products generators. The hierarchy of objects is caused by properties inheritance from the object class to its subclass, respectively, and on each level there is a possibility to endow the object with additional properties which in turn will also be inherited to lower levels.

Instances of the objects used in the HBB and take part in the simulation.

Recipes are described in the structure by the entities in the aggregate equivalent IDEF0 diagram. The inputs of recipes are certain types of objects, and active object directly executing the recipe may be presented as a mechanism. Exit of the recipe are described in the same way. Assistants-agents and passive objects (labor tools) can be used to enhance the recipe mechanism.

Global database stores a set of instances of entities and relationships between them, describing the current state of the environment.

The relationship with other agents can affect to the choice of the recipe. The collective behavior of agents in the MAS involves cooperation of agents in collective problem solving. In operation the multi-agent system the agent can ask for help from other agents, if not able to solve the problem in front of them on their own. At the same time agents can make plans for joint action, not only relying on its own capabilities, but analyzing the plans and intentions of other team members. Simulation of collective behavior is also necessary in cases where agents share a common limited resource for solutions to their problems. Each agent has to take into account the presence of other agents, and the choice of strategy actions of one agent usually depends on the behavior of others.

A feature of the collective behavior of agents is that their interaction in the process of solving specific problems (or a general) creates a new quality of solving of these tasks. In the models of coordination of the behavior of the agents are using the basic idea: the rejection of finding the best solutions for the benefit of "good". This results in a transition from the strict optimization procedure to search acceptable compromise that implements a particular coordination principle.

Landscape

The main element of the environment is the landscape. Objects, both active and passive are located on it.

The landscape is a set of polygons with unique coordinates. Each specific type of polygon is describes by the name, image and property of permeability, affecting the characteristics of the active object, as well as affecting the ability to move around on it.

To implement this functionality was created three entities. The first entity is called the collection of the worlds described by name and dimension of the landscape. As a landscape is described a set of polygons with unique coordinates, the square of the dimension of the landscape indicates how many of these points, and accordingly, polygons are located on it. Thus, the landscape is a matrix of dimension N * N where N - dimension terrain data referred to in essence.

The second entity called a collection of polygons is describing all possible polygons and their properties that can be used as elements of the landscape.

The third entity called the world map is describing the coordinates of the points of the landscape, their membership of a particular landscape, as well as the polygon located at a point with the specified coordinates.

Links are organized in such a way that a specific landscape belongs to a set of polygons with their unique coordinates.

In the process of simulation used instances of GKB concepts - map objects.

To limit the available recipes agent, entity "recipes of agent" was defined - recipes agent. It is describe the recipes available to agent in the early stages of simulation, and filled in the objects placement process on the map. Data stored in this entity are links such as "knows" in the global knowledge base of recipes. In the process of simulation sets of links to GKB may vary.

To store the history of the agents behavior a set of temporal entities responsible for the storage of the changes taking place with agents in the simulation, was implemented.

As noted, previously stored history is used by an active object in the making decision on how to implement the needs.

Base of history

The main base of the history of the system:

- Log of the location of the object
- Log of execution recipes
- Log of ownership changes
- Log of object properties
- Log of relations

In the object location log, data is stored to change the location of all the objects in the simulation. This entity captures a time interval the host agent in a specific location.

In the recipe log, data is stored about the implementation of the specific recipes active objects. This entity fixes what recipe was successful or unsuccessful implemented a particular active object, as well as the time frame in which the active object is trying to execute the recipe.

The actions performed with objects of ownership and time period in which they occurred are represented in the ownership change log.

Time ranges of objects properties values are represented in the log of objects properties changes.

Intervals established relationships between objects and their role in these relations are stored in the log of relation.

Cycle of the needs satisfaction

In contrast to other approaches, in this work is accepted that the need is a request to perform actions on transfer of the intelligent agent to the desired state or preservation of the current state. Requests are formed based on the internal and external events perceived by agent. In this case, the cycle needs to meet may be represented as follows (Figure 1):

1. The formation of a need (type, parameters (the initial and the desired state));
2. Selection of the priority need (type, parameters);
3. Actualization in the agent knowledge mental base of the set of possible recipes types of request execution and choice the type of executable recipe (including opportunity of the agent, experience, the forecast of feasibility and belief to the feasibility).

The recipe determines the types and standards of the required input resources, implementation tools, execution control, environmental requirements recipe applying, and the types and standards of the recipe applying results. The global knowledge base (GKB) has a library of a typal needs and recipes of their execution. Mental knowledge base of agent defined by reference to the GKB.

4. The recipe can be implemented without prior actions (perhaps by scaling the available resources)? Yes - go to claim 5, No - "freezing execution", the formation of additional needs required types with the necessary parameters and go to step 2.

The positive answer is given when:

• available the means of the agent (of his body);

• the necessary instances of inputs, means of execution, recipe management are in spatial, legal and temporary access;

• there are the necessary external conditions.

5. Starting the selected recipe with parameters that ensure the implementation of need.

6. Waiting of the recipe execution end, or waiting of failure about execution it.

7. The recipe performed? Yes - go to Step 8. No - Recovery of need, the transition to step 2.

8. "Defrost" of the execution of waiting recipes? Yes - go to claim 1, No - The end.

The algorithm of needs implementation takes into account the instinctive character and rationality in selecting ways to implementation of a need. Agent is select the recipes and opportunities from a set of known to it that will allow to rational achieve a state of need satisfaction in his opinion.

In the base of the model of rational choice ways to implement the needs is put proposed by the authors of this work methodology of gradual strengthening of agent capabilities, which assumes that an agent in the search of needs satisfaction ways gradually increasing a set of possible needs satisfaction recipes. At that it uses its own experience and experience of other agents, which interact with, as well as available knowledge of other agents.

The main methods of amplification are: physical development; education and skills; acquisition and development of promising new technologies and means of production; research and development of new types of objects; spatial and infrastructure practices; organizational and structural methods, including the division of labor; economic methods; political and ideological practices; etc.

In view of the above identified the following recursive algorithm for selecting the realization of the need:

1. The choice of the need satisfaction recipe from the area instinct MKB (RI). If you have a recipe with satisfactory properties, the choice of it as a priority. The end.

2. The use of local history (experience) of the application of need satisfaction recipes (RI + RH). If you have a recipe with satisfactory properties, and then the choice of it as a priority. The end.

3. The use of the local knowledge (RI + RH + RK), obtained in the course of training, the available reading books, etc. If you have a recipe with satisfactory properties, the choice of it as a priority. The end.

4. Independent search of the need satisfaction recipes through their senses, performing some actions (to do - hear, see, feel) by trial and error (RI + RH + RK + RS). If you have a recipe with satisfactory properties, the choice of it as a priority. The end.

5. The use someone else's experience, and (to an external memory access) (RI + RH + RK + RS + RE). Possible ways:

5.1. The publication of needs for recipes or request to specific agents and the transition to the standby state. If you have a recipe with satisfactory properties, the choice of it as a priority. The end.

5.2. Web search. If you have a recipe with satisfactory properties, the choice of it as a priority. The end.

5.3. Failure to find of recipe. The end.

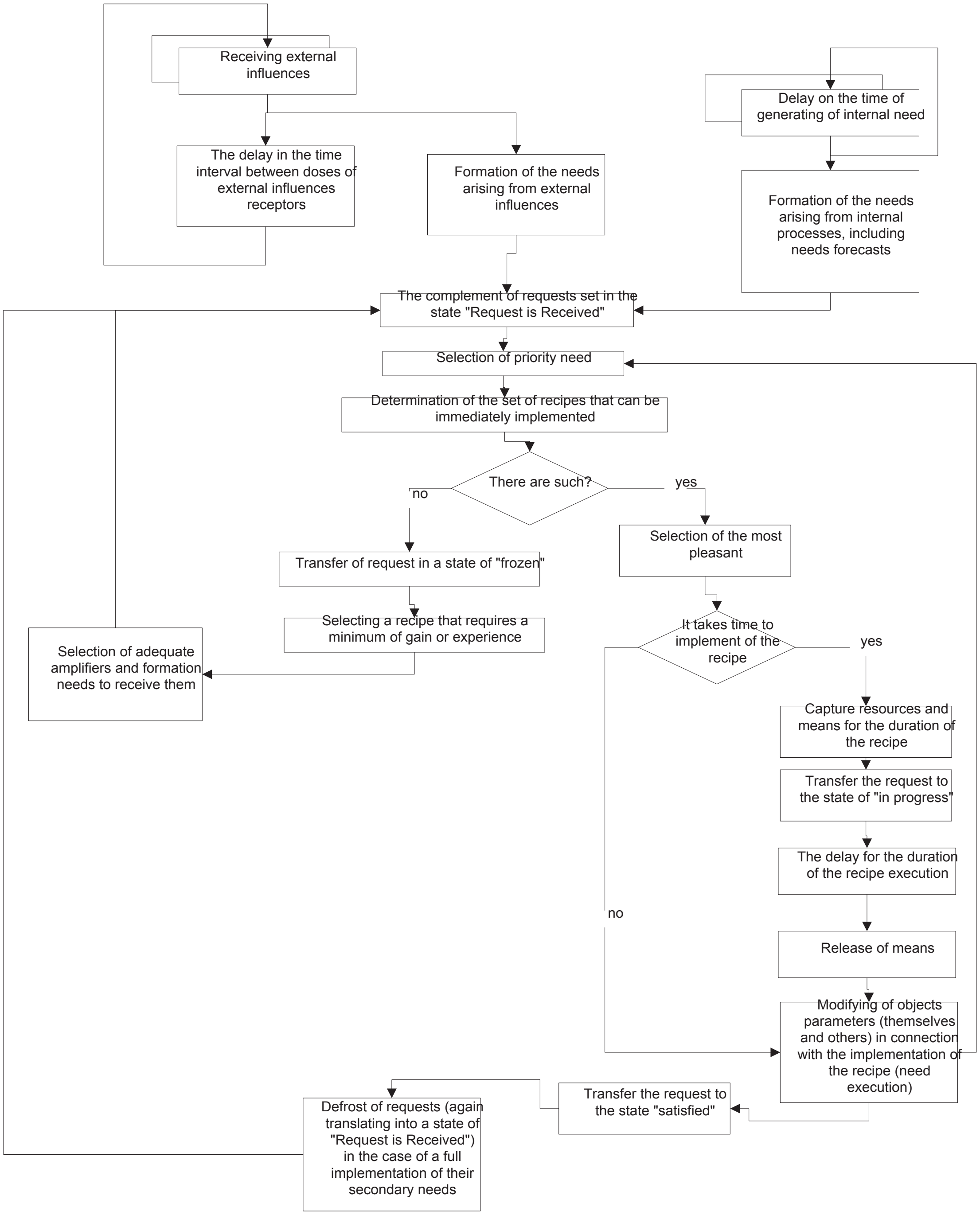
Receiving external influences
The delay in the time interval between doses of external influences receptors
Formation of the needs arising from external influences
Delay on the time of generating of internal need
Formation of the needs arising from internal processes, including needs forecasts
The complement of requests set in the state "Request is Received"
Selection of priority need
Determination of the set of recipes that can be immediately implemented
There are such?
no
yes
Transfer of request in a state of "frozen"
Selection of the most pleasant
Selecting a recipe that requires a minimum of gain or experience
Selection of adequate amplifiers and formation needs to receive them
It takes time to implement of the recipe
yes
no
Capture resources and means for the duration of the recipe
Transfer the request to the state of "in progress"
The delay for the duration of the recipe execution
Release of means
Modifying of objects parameters (themselves and others) in connection with the implementation of the recipe (need execution)
Transfer the request to the state "satisfied"
Defrost of requests (again translating into a state of "Request is Received") in the case of a full implementation of their secondary needs

FIGURE 1. Cycle of the needs satisfaction

It is necessary to note that each of these steps of the algorithm can also be viewed as a need, and accordingly, each of them recursively unfolds under the same scenario.

CONCLUSIONS

This article presents some aspects of the developed multi-agent simulation system. Described landscape used in system, work with global knowledge base (work with recipes in particular), presented the use of the history base of the system. Recipes are described by the equivalent IDEF0 diagrams. Also impact of the collective behavior of the recipe on the choice to the needs satisfaction was affected.

Finally, an algorithm of the needs satisfaction of the agents based on the mechanisms of the needs generating and recipes of the needs satisfaction outlined in the Knowledge Base, was described.

The general description of the model and a detailed comparison of existing architectures with the architecture of the presented model are shown in [6]. The project is also developed in the RePast Simphony environment using Groovy scripting language. In the experiments was examined the dependence of agents' needs of a food according the resources that agents can be able to use to satisfy the needs. The experimental results are confirmed the validity of the proposed approach and the ability to use it to select of the options of territory development control.

REFERENCES

1. Rao A., Georgeff M. P., 1995. BDI-agents: From Theory to Practice. In Proceedings of the first international conference on multiagent systems (ICMAS-95), pp 312–319, SanFrancisco.

2. Hong Jiang, Jose M. Vidal, Michael N. Huhns, 2007. EBDI: An Architecture for Emotional Agents, Proceedings of the Autonomous Agents and Multi-Agent Systems Conference, USA.

3. Wooldridge M. J, 2001. An introduction to multiagent systems, John Wiley & Sons, LTD, Baffins Lane,Chichester, West Sussex, England.

4. Skobelev P.O., 2003. Mul'tiagentnye modeli vzaimodejstvija dlja postroenija setej potrebnostej i vozmozhnostej v otkrytyh sistemah, Avtomatika i telemehanika, Russia, vol. 1, pp 177-185.

5. Klebanov B.I., Antropov T.V., Rjabkina E.M., 2016 BASES OF IMITATION MODEL OF ARTIFICIAL SOCIETY CONSTRUCTION ACCOUNTING OF THE AGENTS' NEEDS RECURSION, 16th International Multidisciplinary Scientific GeoConference SGEM 2016, www.sgem.org, SGEM2016 Conference Proceedings, Book2 Vol. 1, 101-108 pp, Bulgaria.

6. Klebanov B.I., Antropov T.V., Rjabkina E.M, 2015. Principy postroenija rasshirennoj BDI arhitektury agenta v modeli razvitija obshhestva, Teoreticheskie osnovy i metodologija imitacionnogo i kompleksnogo modelirovanija (IMMOD), Russia, pp 116-120.

7. Klebanov B.I., Aksenov K.A., 2015. PRIMENENIE IMITATSIONNOGO MODELIROVANIYA V SISTEME UPRAVLENIYA KACHESTVOM METALLURGICHESKOJ PRODUKTSII. Fundamental'nye issledovaniya, (9-2), 247-251.

8. Aksyonov K.A., Spitsina I.A., Sysoletin E.G., Aksyonova O.P., Smoliy E.F, 2014. Multi-agent approach for the metallurgical enterprise information system development. в CriMiCo 2014 - 2014 24th International Crimean Conference Microwave and Telecommunication Technology, Conference Proceedings, 437-438.