Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 103 (2017) 120 – 125
Procedia Computer Science

XIIth International Symposium «Intelligent Systems», INTELS'16, 5-7 October 2016, Moscow, Russia

Domain theory verification using multi-agent systems

G.M. Novikova*, E.J. Azofeifa

RUDN University, 6 Miklukho-Maklaya str., Moscow 117198, Russia

Abstract

We consider the use of multi-agent systems to verify a domain theory, which describes the object types and their properties, relationships and connections between them, as well as a set of true statements about the functioning laws of the domain objects. We study two approaches to the creation of multi-agent systems capable of verifying the compliance of types and object properties in the real domain, with types and properties defined in the theory. Requirements for the properties and relationship structure of intelligent agents are determined for each approach, as well as a mechanism for creating networks of multi-agent systems, which maximize efficiency and effectiveness in the verification of a given fragment of a domain theory. Simulation results show that the first of these approaches is considerably more efficient in terms of total verification time of a domain theory.

Peer-review under responsibility of the scientific committee of the XIIth International Symposium "Intelligent Systems"

Keywords: multi-agent system; domain theory; verification; intelligent agent.

1. Introduction

Currently, the creation of multi-agent systems (MAS) is one of the strategic topics of research in the field of artificial intelligence[1]. Interest in intelligent systems of this kind is due to the development of engineering, telecommunications and information technologies, which enable to create not only intelligent agents, but also complex dynamic agent systems capable of collecting, processing and transmitting information in real-time that is not only symbolic, but also sonic and figurative. Today, an intelligent agent can be an info-communication or engineering-technical system, as well as a social subject, which, in particular, allows the creation of such complex adaptive dynamical systems like intelligent organizations[2]. In automatic mode, the intelligent agent is not only able to collect information passively, but also to perform an action on an object in order to obtain such information.

A range of tasks commonly solved by MAS has been widely defined, from information gathering and object state recognition in a domain, up to decision-making in control theory over different application fields: medicine, military affairs, construction, and industrial, economic and social systems[2]. In general,

* Corresponding author.
E-mail address: novikova_gm@mail.ru

1877-0509
Peer-review under responsibility of the scientific committee of the XIIth International Symposium "Intelligent Systems"
doi:10.1016/j.procs.2017.01.026

any real domain related to a theory that describes object types and properties, state classes, relationships, interaction laws and object functioning, can be approached using MAS. Typically, a domain theory is formed either based on observations of object behavior, on systems theory of higher order, or is given a priori in the design and development of systems that operate in real time and space. In reality, however, the behavior and properties of objects may contradict or extend beyond the frame of the current theory, specific to the subject area. Thus, a topical task is the constant verification and harmonization of the theory describing a domain and its real state. A mechanism for solving this problem is to create and configure a MAS able, in real-time, to identify inconsistencies and determine incompleteness of a given theory over the domain.

2. Domain theory verification

Let A be a domain theory, which includes the definition of a set of object types and their properties, relationships and connections between them, as well as a set of true statements about the functioning laws of the domain objects. Formally, a simplified theory can be described by a mathematical tool such as predicate calculus[3],

$$S =< T, F, A, R >,$$

where T—the alphabet of the theory (a finite set of basic characters); F—the (countable) set of formulas (also called well-formed formulas - WFF), constructed from elements of the set T using a set of syntax rules; A—the set of formulas referred to as axioms; and R—a finite set of inference rules.

From the point of view of the domain, a part of the alphabet - the set of terms - corresponds to the set of object types, properties, and functional and logical relations between them. Axioms, built using the theory alphabet, establish a correspondence between object type and the set of properties, types of objects and the set of relations, and object type and the set of possible object states.

During the verification process, discrepancy of all the elements of a formal system can be found[4]. In a real domain, the truth value of statements about the properties, relations and states of objects may not hold, as well as the correspondence between objects and relations, objects and states, and objects and properties, as previously defined. Notice that a real domain is typically a dynamic system in which not only the relations, states and properties of the objects change, but also the actual set of properties and relations[5]. However, the construction of a verification mechanism of the dynamic properties described in theory is beyond the scope of this article.

2.1. Verification: objectives and methods

In accordance with the description of a domain theory within a formal system, a few problems arising in the theory of verification can be highlighted:

Objective 1. Let an object within a domain theory be described as a set of properties. Let A_i—the set of properties describing the i-th object, and A—the set of known sets of properties that an object can possess. The verification task consists in finding those objects, whose set of properties and states do not correspond to the subsets $A_i - A$. The verification result consists in either applying actions over the domain in order to bring it in line with the theory, or applying the necessary changes to the theory.

Objective 2. Let S_i—the set of states in which an object can be found, and $V(i) = V(A_i)$—the set of threshold values of the properties that define the state of an object. The verification task consists in determining the real correspondence of state classes and object types. The verification result is either finding new state classes in an object that do not correspond to it according to the theory, or determining an absence of states in which the object should be found, according to the theory.

Objective 3. Let S_{adj}—the set of adjacent states of an object. The verification task consists in determining the presence of transitions between the states of an object. The verification result is either finding new alternative adjacent states, identifying theoretical state transitions that do not occur in the real domain, or finding real state transitions that theoretically do not correspond to the object.