Computers and Mathematics with Applications 64 (2012) 3658–3665

Contents lists available at SciVerse ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Machine learning in agent-based stochastic simulation: Inferential theory and evaluation in transportation logistics

Janusz Wojtusiak [a,*], Tobias Warden [b], Otthein Herzog [b]

[a] *George Mason University, 4400 University Dr., Fairfax, VA 22030, USA*
[b] *University of Bremen, Bremen, Germany*

ARTICLE INFO

Keywords:
Multiagent systems
Multiagent-based simulation
Machine learning
Inferential theory of learning
Autonomous logistics

ABSTRACT

Multiagent-based simulation is an approach to realize stochastic simulation where both the behavior of the modeled multiagent system and dynamic aspects of its environment are implemented with autonomous agents. Such simulation provides an ideal environment for intelligent agents to learn to perform their tasks before being deployed in a real-world environment. The presented research investigates theoretical and practical aspects of learning by autonomous agents within stochastic agent-based simulation. The theoretical work is based on the Inferential Theory of Learning, which describes learning processes from the perspective of a learner's goal as a search through knowledge space. The theory is extended for approximate and probabilistic learning to account for the situations encountered when learning in stochastic environments. Practical aspects are exemplified by two use cases in autonomous logistics: learning predictive models for environment conditions in the future, and learning in the context of evolutionary plan optimization.

1. Introduction

Modeling of complex systems often requires using simulation techniques that approximate real-world systems' behavior. Stochastic simulation is frequently used to model systems whose operation cannot be captured directly by deterministic rules, and thus need to be approximated probabilistically. Further, direct modeling of entire systems is often not possible due to their complexity and need to be distributed. The latter can be realized by multi-agent systems whereby instead of modeling an entire system as a whole, only the behavior of individual agents is explicitly modeled [1,2]. These actions of individual agents can be deterministic or stochastic in nature, depending on the specific problem modeled.

In order to perform their tasks better, intelligent agents learn from their own or other agents' experiences. In many cases the learning process is long and may be costly, thus, in practice it is infeasible in real world-applications. Instead, intelligent agents can be trained within simulation systems that mimic real-world environments, and then deployed to solve real tasks. This resembles how people receive training, i.e., pilots training in flight simulators.

This paper describes agent-based systems as means of performing stochastic simulations, with special focus on providing environments for agents' learning. It describes the agent's learning processes using the Inferential Theory of Learning (ITL), and relates them to the simulation problems. This extension of ITL describes how artificially generated data from simulations can provide the experience necessary for agents' training.

* Corresponding author.
E-mail addresses: jwojtusi@gmu.edu (J. Wojtusiak), warden@tzi.de (T. Warden), herzog@tzi.de (O. Herzog).

0898-1221/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2012.01.079

2. Agent-based simulation

Multiagent-based simulation (MABS) systems adopt the agent programming paradigm for simulation, and facilitates the design and development of complex simulations through the development of simulation actors and simulation environments. MABS has been employed successfully for systems analysis and evaluation in a variety of domains, ranging from biological or social systems to complex business processes such as shop floor logistics or supply chain management. This approach to the simulation of complex systems is based on the micro-level design, where individual decision makers are modeled explicitly as autonomous agents embedded in dynamic environments. In contrast to alternatives, such as equation-based modeling, MABS facilitates the design of complex systems due to task decomposition, a natural mapping from real-world actors or entities to agents, and the focus on modeling of individual behavior [3].

In general, MABS can combine distributed discrete event or time-stepped simulations with decision-making encapsulated in agents as separate and concurrent logical processes [3]. In classical simulation systems, the logical processes involved as well as interaction links have to be known in advance and must not change during the simulation [4]. This is not the case in MABS as each agent may interact with all other agents [5]. Agents may join or leave the simulation during execution, depending on a stochastic simulation model or human intervention. Agents may also learn to improve their abilities.

Thus, the key idea behind agent-based simulation is that complex processes, do not necessarily require global simulation models. Rather, complex systems can be described by their parts, and interaction mechanisms between these parts. All components of the modeled systems are represented by agents, each equipped with a behavior model. Thus, an agent-based system can be described as a triple (A, S, C), in which A is a set of agents, S is the schema that defines environment in which the agents operate, and C is a communication mechanism between the agents. The simulation process can be described as a function of these three components:

$$(A_{t+1}, S_{t+1}, C_{t+1}) = F(A_t, S_t, C_t). \tag{1}$$

This model assumes that all three components change over time. Although the process seems to be Markovian, i.e., the individual agents may act based on the past history of experiences. Agent-based simulation is usually realized as a discrete event simulation, in which each agent is "executed" in each simulation step.

The above definition of an agent-based system does not explicitly describe the system's goal. Rather, each agent in the simulation has its own goal, explicitly or implicitly defined by the agent's behavior rules. This paper concerns two types of agents in A, stochastic agents that mimic real-world environment, and learning agents that improve their skills or extend knowledge base by interacting with stochastic agents within the environment defined by S.

2.1. Distributed decision making

In multiagent-based systems, control of processes is delegated to agents. This is a shift from more traditional simulation methods with centralized control. This applies to both, agents responsible for stochastic simulation processes and learning agents that interact with the simulation. In many application areas, including transportation logistics [6], distributed decision making has been shown to perform as well as centralized systems. Moreover, distributed systems are more robust and better equipped to handle disruptions to the process.

2.2. Learning agents

Through their interaction with stochastic simulation environments, intelligent agents gather experience. That experience, combined with agents' background knowledge and the information exchanged between agents can be used for learning. In multiagent systems, the goal of learning is to improve agents' performance on a specific task.

The most commonly studied form of learning in multiagent systems is *reinforcement learning*, in which the objective is to improve the performance of an agent in solving a specific task [7–9]. In reinforcement learning, agents learn by trial-and-error interaction with their environment in which they maximize the reward by performing tasks better. Methods such as Q-learning [10], Temporal Difference Learning, Adaptive and Dynamic Programming are frequently used. Reinforcement learning addresses the global problem (task) being solved by an agent, thus the learning process is very complex. In other words, the learning space for the agent is very large—only the final outcome is used to score agents' actions. Many approaches have been used to improve reinforcement learning. For example, in [11] authors combine reinforcement learning with rule learning in order to reduce learning space. An overview of agent-based learning systems is available, for instance, in [12].

The presented work focuses on learning parts of the task and using the learned components as input to an agent's decision model, DM. Here, action of i-th agent is determined by the agent's decision model, based on the input $\bar{x}$ that represents perceived state of the environment and agent's internal state. The model DM is considered to be the agent's knowledge on how to solve the decision problem.

$$\mathrm{Action}_i = \mathrm{DM}_i(\bar{x}). \tag{2}$$