Expert Systems with Applications 40 (2013) 3207–3219

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

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

Agent-based game-theoretic model for collaborative web services: Decision making analysis

Babak Khosravifar [a], Jamal Bentahar [b,*], Rabeb Mizouni [c], Hadi Otrok [c], Mahsa Alishahi [b], Philippe Thiran [d]

[a] Faculty of Art and Science, McGill University, Montreal, Canada
[b] Faculty of Engineering & Computer Science, Concordia University, Montreal, Canada
[c] College of Engineering, Khalifa University of Science, Technology, & Research, Abu Dhabi, United Arab Emirates
[d] PReCISE Research Centre, University of Namur & Sirris, Brussels, Belgium

ARTICLE INFO

Keywords:
Agent computing
Web service
Decision making
Payoff

ABSTRACT

In agent and (web) service computing, collaboration takes place when distributed entities have limited knowledge and capabilities, so they cannot perform required tasks without interacting and helping each other. For instance, web services, which are loosely-coupled business applications, are called to cooperate in distributed settings for the sake of efficiency. In this context, agents that abstract and act on behalf of web services could act in cooperative groups that gather a number of agents sharing some common goals. Enabling those agent-based web services to decide about their strategies in terms of joining and acting within groups, inviting other agents to join, and leaving a group to act alone is an open issue that we address in this work. In this paper, we propose a framework where agent-based web services select strategies that maximize their outcomes. These strategies could be categorized into cooperative strategies involving other agents and strategies that highlight the single operative attitude. Although cooperation seems to bring better utility to cooperative agents, we highlight that web services in some environments obtain better outcome while they act individually (i.e., outside the group). This means that the cost of cooperation (in some particular cases) might negatively influence the outcome and obtained utility. As solution, we propose in this paper (1) an agent-based model that formalizes web services decision making where different parameters are considered; and (2) a game-theoretic framework that analyzes the web services strategies allowing them to maximize their acting performance where non-zero-sum games are being used. The paper presents theoretical results, which are also confirmed through extensive simulations.

1. Introduction

A multi-agent system (MAS) is any system that is composed of multiple interacting agents playing various roles and following different acting strategies towards achieving their predefined goals. Each agent is in fact a decision maker that seeks to effectively accomplish its goals. Some of these goals are shared with other agents while others are only defined in the individual agent's desires. Because agents have limited capabilities and knowledge and can observe the environment only partially, they are called to interact with one another. The network of web services is an example of MAS where agents represent and abstract web services and act on behalf of them to make decisions regarding their cooperative attitudes (García-Sáncheza, Valencia-Garcíaa, Martínez-Béjarb, & Fernández-Breis, 2009; Lomuscio, Qu, & Solanki, 2012; Vidonia, García-Sánchezb, Gasparettoa, & Martínez-Béjarb, 2011). A web service with an associated intelligent agent is capable of providing some services in some certain domains (service category). To do this, the agent-based web service, called web service agent, maintains some interactions with the environment and gets involved in group tasks hoping to enhance its productivity (Bentahar, Khosravifar, Serhani, & Alishahi, 2012). In this setting, a typical service consumer is an intelligent agent capable of comparing different service qualities and based on its domain knowledge, attempts to obtain a high quality service.

In this type of agent-service computing (Kun, Manwu, Hong, & Jian, 2010), each agent web service still acts individually, but it is the resulting joint action that produces the outcome (Charif & Sabouret, 2011). Collaboration is therefore a crucial aspect that improves web services's performance, robustness and scalability (Yahyaoui, 2012). The goal of cooperation is to improve the outcomes for the group as a whole while the key aim of agent-based web services is to maximize their individuals' outcome in terms of business opportunities and performance. In this context, performance is defined as the extent to which a web service agent

* Corresponding author. Tel.: +1 514 848 2424; fax: +1 514 848 3171.
E-mail address: bentahar@ciise.concordia.ca (J. Bentahar).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.12.034

is successful in accomplishing its goals. This is achieved when it can effectively use its resources and capabilities to accomplish the required tasks, which allows obtaining higher outcome. Adversely, a web service agent is unsuccessful (i.e., poorly efficient) when it fails to manage its resources due to either high service demand or no demand. To this end, the web service's goal is to maximize its performance with respect to its individual capabilities.

To address web service cooperation, there have been efforts attempting to model and analyze collaborations within *communities* of web services (Khosravifar, Bentahar, Thiran, Moazin, & Guiot, 2009; Rosario, Benveniste, Haar, & Jard, 2007; Ruth & Shengru, 2007). Communities are frameworks that gather functionally-similar web service agents that share a common goal (Khosravifar, Bentahar, Moazin, & Thiran, 2010; Maamar, Subramanian, Thiran, Benslimane, & Bentahar, 2009). In the context of communities, we distinguish web services collaboration from web services composition (Yeung, 2011). By collaboration, we mean that the community aggregates web services capable of interacting with one another to manage allocated tasks, for example by allowing a web service to replace another that is incapable of executing a task. By composition, we mean extension initiated by a web service to satisfy a specific request. The web service agent invokes another web service, waits until its execution completes and uses the results of its execution, which results in a type of composite service. Similar web services providing flight booking, hotel booking, and car booking with different qualities of service, such as http://www.Orbitz.com and http://www.cheapoair.com can be gathered in a community. Another concrete example is a community of web services providing healthcare such as http://www.medcommons.net and http://www.hipaaspace.com.

In all the aforementioned frameworks of communities, the objective is to increase the overall performance where web services shape a distributed system. However, in such frameworks, strategies that web services can follow to achieve this goal are limited to a form of aggregation-based collaboration. More elaborated strategies considering when to collaborate and when to act alone are yet to be analyzed and investigated. Such strategies can help communities and individual web services achieve higher performance in using their resources. *One of the challenging issues to be addressed* is to determine if being involved in a community to cooperate with other agent web services will necessarily end up with higher outcome for an individual web service agent with some limited capabilities. In the affirmative case, identifying the best time for an individual web service agent to join a community to cooperate with others is another issue not resolved yet in the literature.

1.1. Motivations

The aim of this paper is to investigate strategies representing long-term rational behaviors agent-based web services and communities can select to increase their respective performance. We use game-theoretic principles that have been advocated to design cooperative and competitive, self-interested multi-agent systems (Pendharkar, 2012). More precisely, we present a game-theoretic model in which web services can either act alone or cooperate with other web services within a community. Each entity (single web service or community of web services) manages its reputation, market share, capacity, and performance parameters. Using our proposed framework, interactive services are capable of making decisions in uncertain and stochastic MAS environment. In fact, agents follow their own goals and therefore, their acts are not fully predictable. In such models, incentives could deviate decision-making processes that are maintained by agents from particular directions that might be socially optimal for the whole network. Therefore in this paper, a game is defined between a particular single web service and the representative of a typical community (called *master web service*). Each entity looks for better performance following strategies of *joining/leaving* a community, *accepting/refusing* a request to join a community, and *inviting* to join a community from the master. In different scenarios, we investigate the situation that maximizes players' performance.

1.2. Contributions

Overall contributions of the paper are threefold: (1) we provide a distributed network of web services and consumers where the task allocation problem is regulated by a mechanism taking reputation, market share, and performance into account; (2) we propose a game-theoretic analysis investigating the stabilized situation within which entities achieve high performance where non-zero-sum games are used; and (3) we identify thresholds allowing the master web service of a community to identify the optimal number of web services to be part of the community. We also provide experiments that show and uphold the impact of our game-theoretic analysis on the behavior of rational web services.

1.3. Paper organization

The rest of the paper is organized as follows. In Section 2, the components of the model we are considering are introduced. This includes web services, communities, service consumer agents, the feedback system, our reputation mechanism, and three key parameters: market share, capacity, and risk. Section 3 presents our game-theoretic model where the computation of the web services's and communities's performance is investigated. The scenario of the web service being outside the community and the scenario of the web service being part of the community are analyzed using non-zero-sum games. Section 4 introduces our simulation settings and results to confirm the theoretical findings. Section 5 discusses relevant related work. Finally Section 6 concludes the paper.

2. The model architecture

Before introducing our agent-based game-theoretic model, we present in this section its architecture components, which are the core of our framework. The purpose is to introduce the different concepts, which will be used through the paper.

2.1. Web service

In this paper, we consider a web service as a software application associated with a rational agent that is capable of providing a service (Kun et al., 2010) (a web service agent i is denoted by w_i). The agent represents and acts on behalf of the web service and its goal is to maximize its individual income. We define income as the total obtained fee that compensate the provided service. In short-term, the goal of a typical web service agent would be to attract the most possible service consumers and thus, obtain relatively high service fee, not necessarily appropriate for an agent with rational behavior in the long-term. Rational agents are expected to follow goals that are achievable in long-term interactions specifically in dynamic environments. For instance, an acting strategy could yield a relatively high outcome in the short-term, while the side effect of such an action is loosing a network of consumers in the future. Consequently, the active web service agent in this case requires a mechanism that guarantees its survival in the multi-agent environment with competitors, while maximizing its income. In this mechanism, there is a number of internal and