Expert Systems With Applications 45 (2016) 359–372

Contents lists available at ScienceDirect

Expert Systems With Applications

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

A Stackelberg game for distributed formation of business-driven services communities

Omar Abdel Wahab[a], Jamal Bentahar[a,*], Hadi Otrok[a,b], Azzam Mourad[c]

[a] *Concordia Institute for Information Systems Engineering, Concordia University, Montreal Quebec, H3G 2W1,Canada*
[b] *Department of Electrical & Computer Engineering, Khalifa University of Science, Technology & Research, Abu Dhabi, UAE*
[c] *Department of Computer science and Mathematics, Lebanese American University, Beirut, Lebanon*

ARTICLE INFO

Keywords:
Service federation
Stackelberg game theory
Distributed community formation
Cloud computing
Business ecosystem

ABSTRACT

With the advent of cloud computing and the rapid increase in the number of deployed services, the competition between functionally-similar Web services is increasingly governing the markets of services. For example, Amazon and Google are in an intense competition to dominate the market of cloud-based Web services. Such a highly competitive environment motivates and sometimes obliges services to abandon their pure competitive strategies and adopt a cooperative behavior in order to increase their business opportunities and survive in the market. Several approaches have been advanced in the literature to model the cooperation among Web services in a community-based environment. However, the existing approaches suffer from two main drawbacks that limit their effectiveness in the real-world services market. First, they rely on a centralized architecture wherein a *master* entity is responsible for regulating the community formation process, which creates a single point of failure. Second, they ignore the business potential of the services and treat all of those services in the same way, which demotivates the participation of the well-positioned ones in such communities. To tackle these problems, we distinguish in this paper between two types of services: *leaders* and *followers*. Leaders are those services that enjoy high reputation, market share, and capacity of handling requests; whereas followers are those services that cannot compete against the leaders. Thereafter, we model the community formation problem as a virtual trading market between these two types of services and propose a distributed Stackelberg game for this purpose. Promisingly, the proposed model gives guidance to a cooperative model that can be applied in the real markets of Web services in order to achieve higher performance, efficient services compositions, and better resources utilization. The performance of the model is analyzed using a real-life flight booking dataset that includes 2507 services operating on the Web. Simulation results show that the proposed model is able to increase the satisfaction of Web services in terms of gained payoff and reputation and the satisfaction of users in terms of quality of service provided to their requests compared to the existing models, namely the one-stage game theoretical model and a heuristic model.

1. Introduction

The industry push is increasingly shifting toward the execution of distributed business processes that spread over multiple applications. This requires assuring high-levels of interoperability and wider flexibility in managing business processes (Agostino, Michele, & Paola, 2007). The trend is to adopt the service-oriented architecture (SOA) as a strategic paradigm for achieving and automating the business processes, thanks to the wide set of benefits it provides such as: loosely coupling, reusability, location transparency, parallel development, and better scalability (Eric & Greg, 2004; Li, Zhang, & O'Brien, 2011). SOA is constructed by means of a collection of services that communicate with each other. This demands software components to be able to understand the objectives of the process, deduct the steps needed to attain these objectives, and select the adequate services to communicate with Barros et al. (2011). These requirements emphasize the need for a certain degree of autonomy enabling Web services to adapt to the circumstances that they may face during the run-time such as the need to cooperate with other services or the ability to manage a composition request. In this scope, autonomy has been widely investigated as a building block property allowing Web services to cope with the surprising challenges that may arise due to the dynamic, heterogeneous, and complex environment within which they operate (Alferez & Pelechano, 2011; Hamadi & Benatallah, 2003; Paolucci & Sycara, 2003). Agent-based Web services

* Corresponding author. Tel.: +1 514 848 2424; fax: +1 514 848 3171.
E-mail addresses: o_abul@encs.concordia.ca (O.A. Wahab), bentahar@ciise.concordia.ca (J. Bentahar), Hadi.Otrok@kustar.ac.ae (H. Otrok), azzam.mourad@lau.edu.lb (A. Mourad).

http://dx.doi.org/10.1016/j.eswa.2015.09.047
0957-4174/

(García-Sánchez, Valencia-García, Martínez-Béjar, & Fernández-Breis, 2009; Gibbins, Harris, & Shadbolt, 2004; Maamar, Mostefaoui, & Yahyaoui, 2005; Maximilien & Singh, 2003) is a paradigm that implements this vision by allowing Web services to make appropriate decisions in order to adapt to the real-time challenges. Such an agent-based model has shown to be successful in a wide range of application domains including Grid and Cloud resource management (Kang & Sim, 2012), Intelligent Transportation Systems (Wahab, Otrok, & Mourad, 2013, 2014a, 2014b), and humans daily activities management (e.g., personalized home-care treatments) (Isern et al., 2011). In this context, two decision making levels have to be distinguished: (1) the strategic level; and (2) the operational level. The strategic level is about decisions related to the cooperation agreements between providers (companies). The operational level, which is the focus of the paper, deals with real-time decisions to participate in a particular cooperation request based on the current situation of the involved services. Thus, two Web services cannot decide to cooperate unless an agreement between their providers does a priori exist. On the other hand, the existence of such an agreement does not entail that the Web services will be always cooperating. This decision is mainly driven by the real-time parameters of the services.

Motivations. Communities of Web services have been proposed as virtual clusters grouping autonomous Web services offering similar functionalities to provide a joint environment of collaboration and interoperability among different applications (Khosravifar, Bentahar, Moazin, & Thiran, 2010; Maamar, Subramanian, Bentahar, Thiran, & Bensilamane, 2009; Yahyaoui, Maamar, Lim, & Thiran, 2013). The high-level objective of the community is optimizing the responsiveness, quality, and availability of the services as well as facilitating their discovery. The individual objective of each service in the community is to increase its share of assigned requests and augment the chances of participating in composition sequences.

From the business perspective, the concept of community is similar to that of *Business Ecosystem* (Gueguen, 2006; Muegge, 2013) wherein entities (e.g., companies, lobbies, associations, and so on) are motivated and sometimes obliged to form a joint business ecosystem to be able to survive and compete in the market. In this work, we envisage communities of Web services from the business perspective by considering each community as a business ecosystem gathering Web services offering a common functionality and seeking to improve their position in the market. Torrès-Blay (2009) defines a business ecosystem as "*a heterogeneous coalitions of organisations from different industries forming a strategic community of interests or values, structured as a network, around a leader that manages to impose or share its business vision or its technological standard*". In this paper, we adopt a similar definition for Web services community except that community members are not necessarily coming from "different industries", which reflects the fact that Web services within communities share similar functionalities. Based on Torrès-Blay's definition and as is the case in the real-world business markets, two types of Web services can be distinguished: *leaders* and *followers*. Leaders refer to as those services that are well-positioned in the market thanks to their strong resources and parameters.[1] As real-life examples, Google, Amazon, and eBay may represent such leaders. On the other hand, followers are those that have less resources and parameters compared to the leaders. In the rest of this paper, the terms "*leader (Web) services*" and "*follower (Web) services*" will be used to refer to these two kinds of Web services. Additionally, the terms "confederate" and "confederation", that are inspired by the real-world business context, will be used to describe the act when Web services join each other to form up a joint community.

Several approaches (Khosravifar, Alishahi, Bentahar, & Thiran, 2011; Khosravifar et al., 2013; Khosrowshahi-Asl, Bentahar, Otrok, & Mizouni, 2015; Lim, Thiran, Maamar, & Bentahar, 2012; Liu, Li, Huang, Ying, & Xiao, 2012; Maamar et al., 2009) have been proposed to tackle the problem of building Web services communities. The common trend in these approaches is the use of a Service Level Agreement (SLA; Bianco, Lewis, & Merson, 2008) contract to regulate the community formation process, where a certain Web service is statically designated as a master (or manager) for the community to control, among other issues, the membership of the services (i.e., accepting/refusing and inviting/firing Web services in the community). Different from these approaches, our proposal is a fully distributed formation model, where all Web services are autonomous in making their decisions and there is no need for a central static party to control the membership issue. Such a distributed approach complies better with the autonomous nature of the Web services. Moreover, the existing approaches do not consider the business perspective of the community formation process. Practically, in the real markets of services, some Web services may consider themselves strong enough to act alone and prefer thus not to give the other (less strong) Web services the advantage of being structured within the same community. Such services should receive some privileges in order to stimulate their contribution in the community formation process. To tackle this issue, we use in this paper a Stackelberg game model that gives the leader services, that are well-positioned in the market, the advantage of acting first by selecting the set of followers that they are willing to cooperate with and making the appropriate offer based on their own business needs and preferences.

Contributions. The main contribution of this paper is a distributed services community formation model that takes into account the business perspective of the services. To this end, we formulate the community formation problem as a virtual trading market that works as follows. First, a given leader pre-selects, at a time, a set of follower Web services to confederate with based on their advertised parameters (e.g., the top ten Web services in the domain of hotel reservation) and specifies a limited *quota* based on which the final selections will occur. The size of the quota is determined according to the need of the leader (i.e., based on its business requirements and objectives). The leader publishes its offer consisting of its own reputation, market share, and capacity to the pre-selected set. Under the pre-determined parameters, followers compete with each other in a non-cooperative game model to decide about the payment (community membership fee) to be made for the leader in response to his offer. This payment can be represented as credit, token, or other equivalents. The objective of the followers is to convince the leader to consider them when selecting its final quota, while minimizing the community membership fee. Based on the results of the followers' non-cooperative game, the leader adjusts its strategy by selecting the optimal quota of followers that maximizes its utility. In this scenario, all leader and follower Web services are rational; thus seeking foremost to maximize their own utilities. Therefore, the considered problem can be analyzed by means of game theory. Moreover, the model is characterized by a hierarchical structure, where the leader (first player) optimizes its strategy while being aware of the impacts of its decision on the behavior of the followers (second player). This naturally formulates a two-stage game, where leader services play first and follower services play second after observing leaders' strategies.

The main contributions of the paper are highlighted in the following:

1. Proposing a distributed formation model for services communities, where all services are totally autonomous in making their decisions. To the best of our knowledge, this is the first work that considers a fully distributed environment for Web services communities' formation. Existing approaches suppose the existence of the community and community manager (or

[1] Throughout the paper, the term parameters is used to represent the reputation, market share, and capacity of handling requests of the Web service.