A Combinatorial Auction Model for Composite Service Selection Based on Preferences and Constraints

Mahboobeh Moghaddam¹ ² Joseph G. Davis¹ Taso Viglas¹
¹School of IT
University of Sydney
NSW 2006, Australia
mahboobe@it.usyd.edu.au
joseph.davis@sydney.edu.au
taso.viglas@sydney.edu.au

²NICTA
Australian Technology Park
Sydney, Australia

Abstract—We propose a novel market-based approach for dynamic composite service selection based on combinatorial auctions. The combinatorial auction model we developed allows us to incorporate service providers’ and requesters’ preferences in the service selection process. From the providers’ perspective, the combinatorial formulation allows them to express their preferences for offering combinations of services, or bundles. Moreover, the combinatorial model has the potential to lower the overall cost to the service requester as a result of providers offering discounts over service bundles. The proposed model also enables the service requester to express their preferences over the types of bundles by defining constraints over the configuration of the composite service provisioning, and data-cohesiveness of the bundles. We have mapped the problem to an Integer Linear Programming formulation and performed a number of experiments to evaluate the proposed model.

Keywords- web service composition; composite service selection; combinatorial auction; data cohesion

I. INTRODUCTION

Service composition presents the opportunity for rapid development of complex applications by facilitating the logical composition of existing web services. The selection of the most appropriate services to be composed together, known as composite service selection, has become an important research problem in the service-oriented computing realm.

With the growing market for web services on today’s Internet, it is possible to find tens of web services online with similar functionality. These services are differentiated based on their Quality of Service (QoS) offer and price which are the most important criteria for QoS-based service selection.

Existing approaches to QoS-based composite service selection mostly assume that each provider offers a single service. Even if they offer more than a service, the offers are not related; i.e. there is no possibility of offering bundle of services. This is in contrast to the current marketing practice of spending billions of dollars for promoting service bundles [1]. Service consumers are encouraged to buy bundles of services through incentives such as the bundle’s reduced cost, more convenient billing options, or free gifts [1]. From the providers’ perspective, the motivations for bundling include increased competitive power in the market by offering a discount over the bundle’s price, increased consumer loyalty [2], or exploiting the complementarities among services [3].

Furthermore, most service selection approaches allow the service requester to specify their constraints over a specific set of service’s quality attributes, including execution time, availability, reputation or budget. Quality attributes such as maintainability or (provider-) dependability, and preferences over the configuration of the composite service provisioning are largely neglected in the extant literature.

In this paper, we develop a model for composite service selection based on combinatorial auctions. This model views the composite service requester as the auctioneer and the service providers are the bidders who bid to offer services for the composite service. This makes our auction a reverse or procurement type of auction. We have also enabled the service requester to specify preferences over the configuration of composite service provisioning, and the desired level of cohesion for the composite service. As cohesion is known to be correlated with maintainability [4-6], the proposed constraint over cohesion helps the service requester define the required level of this quality attribute.

The model is defined as an Integer Linear Programming (ILP) problem with the cost minimizing objective subject to quality constraints and the newly introduced cohesion and service provisioning constraints. We have performed a number of experiments to evaluate the time complexity of the proposed model as well as the influence of the
parameters involved in the model on the cost of the composite service.

The remainder of this paper is organized as follows. Section 2 outlines the proposed combinatorial auction model in detail. The mathematical formulation of the proposed model is presented in Section 3. Section 4 discusses the results of the experiments that are performed. Related works are presented in section 5. Section 6 provides the conclusion.

II. THE COMBINATORIAL AUCTION MODEL

Auctions as market-based mechanisms, allow for dynamic pricing which is critical for products such as web services that are characterized by dynamic execution environments (in terms of the provider’s available resources), and users with different and changing demands.

In combinatorial auctions, multiple distinct items are auctioned simultaneously and the bidders can bid over a combination of items, or bundles. Bundling enables the bidders to express their preferences over the items more fully, which leads to economic efficiency and greater auction revenue [7]. The possibility for bundling is particularly important when bidders have preferences not just for specific items but for their bundles due to the complementarities or substitutability effects that exist among the items [8]. The dependencies can make the utility of a bundle greater (when items are complements) or smaller (when they can be substitutes) than the sum of the utilities of the individual items.

In the web services domain, services combined to form a composite service are dependent on factors such as sequence of execution time, resources consumed, input/output message or data, and user-specified constraints. These dependencies make it attractive for service providers to offer services as in bundles. Consider a service provider interested in providing services for a set of consecutive services which exchange data. By provisioning for these dependent services and bidding for them in one bundle, the provider can internalize some of the costs of interface compatibility required for data exchange. This can decrease the cost of service provisioning. Consequently, the discount over the bundle’s price can result in the provider’s increased competitiveness in the market for web services.

In our proposed model, we allow the service providers to offer services in the bundles. We also offer the service requesters the opportunity to specify the type of bundles they require in two areas: a) They can specify their requirements for the degree of cohesion in the bundles and composite service which has a significant impact on the maintainability of the composite service, b) they can control the configuration of composite service provisioning which permits them to express a number of complex preferences affected by this configuration.

A. Cohesion in a Composite service

Maintainability is among the key qualities that a software system should support. According to IEEE Standard, maintainability is defined as “the ease with which a software system or component can be modified to correct faults, improve performance or other attributes, or adapt to a changed environment” ([9], p46). A variety of metrics have been proposed in the literature for measuring maintainability [10], including a number of studies which have shown that cohesion is positively linked to maintainability [4-6]. In general, cohesion can be defined as “a measure of the bindings of the elements within a single module” [11]. Based on the level of encapsulation and abstraction, cohesion has been defined more specifically for procedural systems [12], object-oriented paradigm [13, 14], and service-oriented softwares [5, 15]. Cohesion is broadly classified as functional, sequential, communication, temporal, logical and coincidental [12]. Functional bindings create the strongest level of cohesiveness and coincidental bindings create the weakest. As well, higher classes of cohesion include the characteristics of the lower ones.

In defining cohesion for service oriented systems, service is the main design construct to apply encapsulation and abstraction principles. In a composite service where services are provided by different providers, there is another abstraction level which is the bundle.

To the best of our knowledge, no definition of cohesion exists for a composite service considering the bundles as another level of abstraction. We characterize the cohesion for a composite service based on the bundles as the abstraction level and measuring the binding of its elements, i.e. services.

We measure cohesion for a composite service based on direct data dependencies between services, the reason is that In the web service selection problem, we are dealing with web services which are designed to implement (realize) separate and distinct functionalities. This makes the functional cohesion less important in this context. The next level of cohesiveness is sequential which exists when part of the output of one element is the input to another element. We can map this cohesion to the service-to-service message or data dependency which is considered as one of the most important types of dependencies between services [16]. Data dependency exists when (part of) the output of one service is consumed as (part of) the input of another service [17]. Its importance is due to the fact that ultimately at the lowest level, the connection between services is through mapping the input and output messages between partner services’ ports.

Being able to adjust the cohesion level of the bundles, and consequently the composite service is of significant importance for the service requester. The proposed cohesion metric directly influences the composite service maintainability, and also its (provider-) dependability. The service requester might be interested in having more data-cohesive modules or less cohesive ones based on the composite service structure, knowledge about the web service’s market, or user-specific constraints.

For instance, following the “design rule theory” [18], when designing complex systems, it is preferred that the tasks strongly dependent on each other be performed by the same doer. This is called modular design and the main reason behind it is to achieve desirable features such as change manageability, and maintainability. For the same reason, the service requester might prefer the strongly dependent services to be performed by the same service
provider. In other words, they may prefer the bundles with higher cohesion.

However, as the web services’ execution environment is the Internet, with service providers mostly being unreliable, service requester might be interested in minimizing the risk of composite service execution failure by reducing the degree of dependability to the providers. This can be achieved by assigning tasks to different independent providers, which means minimizing the cohesion in the bundles.

To include these requirements in the model, we define the data cohesion factor (DCF). The DCF of a bid is calibrated based on the direct data connections between the services it is bidding for, i.e. if a bundle is bidding for two services exchanging data, its DCF will be 1, and 0 otherwise.

As an example of a privacy concern related to configuration of composite service provisioning, consider this motivating scenario: in the composite service, the collective data provided to two individual services can reveal a person’s true identity despite anonymizing the data. To preserve privacy, the service requester might want to enforce the two services to be procured by different providers. Another example for a security requirement affected by provisioning configuration is when some of the services in the composite service need to support an encryption algorithm. The requester might then need to minimize the number of providers that have access to the encryption key.

More specifically, the service requester might have two types of preferences regarding the configuration of composite service provisioning: a) specific services need to be provided by the “same” provider, or b) be provided by “different” providers. The security concern and the privacy concern discussed before are the examples for these preferences, respectively.

We have mapped the composite service selection to an Integer Linear Programming problem in which the objective function is to minimize the cost for the service requester, subject to quality and data-cohesion constraints and service provisioning preferences.

III. MATHEMATICAL FORMULATION OF THE COMBINATORIAL AUCTION MODEL

We assume that the composite service is defined at a high level, as an abstract business process (BP) which comprises a set of tasks. The service providers bid to procure services for the tasks in the business process. For simplicity, we assumed the BP only includes sequential structures. However, it is possible to extend the model to other structures (parallel, loop, conditional) following techniques such as the one suggested in [19].

Let $B$ be the set of all received bids from all providers, with an arbitrary member denoted as $b_i$ where the total number of all received bids is $N$. Let Task be denoted as the set of all tasks in the business process, with an arbitrary member defined as $t_i$ where the total number of tasks in the BP is $M$. Let also $K$ be the total number of bidding providers for these tasks.

Each bid $b_i$ is defined as $b_i=(T_i, c_i, Q_i)$, where $c_i$ is the cost of providing service(s) for the task(s) in the set $T_i (T_i \subseteq \text{Task})$ and $Q_i$ is the set of offered quality values for those service(s). Each member of $Q_i$ is a tuple, including quality attributes’ values of the tasks in $T_i$. For the current model, we consider two quality attributes in the quality tuple: availability and response time denoted as $v_i$ and $r_i$ respectively; i.e. $Q_i = \{(v(t), r(t)) | t \in T_i\}$. $v_i$ and $r_i$ are defined as functions from the set of tasks to a positive number.

The objective function is to minimize the cost for the service requester, (1). The decision variable is denoted as $z_i$ to be 1 if $b_i$ is a winning bid and 0 otherwise.

Constraint (2) ensures that each task is assigned to no more than one provider. It also implies that any number of non-overlapping bids of the same provider can win the auction simultaneously. In other words, the bidding language is OR [20]. To get the unique assignment, we defined matrix...
The aggregation function of availability is linearized using a logarithm function [19].

\[ \sum_{i=1}^{N} \sum_{j \in X} a_{ij} \times d_{ik} \times z_{ij} \leq |X| + BIGM \times (1 - y_k) \]  

(6-2)

\[ \sum_{k=1}^{N} y_k \geq 1 \]  

(6-3)

Constraint (7) ensures that the tasks in \( Y \) will have different providers provisioning them, where \( Y \) is a subset of \( Task \).

IV. EXPERIMENT

In order to evaluate the proposed service selection model, we performed a number of experiments. In the first experiment, the impact of the number of tasks on the combinatorial auction’s cost is evaluated. In the second experiment, the time required to find the optimum solution is assessed. The impact of the number of tasks in the bundles over the cost of composite service is evaluated in the third experiment. Finally, the influence of cohesion factor on cost is addressed in the forth experiment.

The combinatorial auction model is implemented using AMPL\(^2\), and the related ILP problem is solved with CPLEX 10.0, using a computer with 16 processors, each 1600 MHz and total memory of 24 GB RAM.

The combinatorial bids for the experiments were generated by the CATS suit [22], using arbitrary distribution. CATS is a suit of distribution families for generating combinatorial bids for five real-world domains of combinatorial auctions. The classification of these domains is based on the type of dependencies that exist between the items of a bundle. The arbitrary distribution is used to generate bids for domains when there are arbitrary dependencies between the items under auction.

As the bids generated by CATS only include bundles’ items and price, we added quality attributes to the bids using QWS Dataset [23] which includes real web services’ quality data. Overall, 2950 problem instances were solved for the following experiments.

A. The impact of the number of tasks

Two parameters are included in the experiment design, summarized in Table I. For each factor combination, 30 independent problem instances were generated (overall 450 problem instances were solved). Cost is averaged across them. Fig. 2 shows the total cost of the winning bids as a function of the number of bids attending in the auction, for business processes with 10, 20 or 30 tasks.

### TABLE I.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Level</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Num of bids</td>
<td>100, 150, ..., 300</td>
<td>5</td>
</tr>
<tr>
<td>Num of tasks in the BP</td>
<td>10, 20, 30</td>
<td>3</td>
</tr>
</tbody>
</table>

\(^2\) A Mathematical Programming Language (http://www.ampl.com/)
To be able to compare the cost of the auctions with different number of tasks involved, we calculate the relative cost of the winning bids, that is the ratio of the cost of the winning bids to the cost of the first data point, for 100 bids. Fig. 2 shows that having more bids attending the auction decrease the cost of composite service. However, the rate of the decrease is much faster when the number of tasks under the auction is lower, as shown in Fig. 3. As we see in Figure 3, when we increase the number of bids from 100 to 200 we can expect savings of over 10% in the total cost, while 300 bids would lead to an expected savings of over 20% in the total cost.

Fig. 2 shows that having more bids attending the auction decrease the cost of composite service. However, the rate of the decrease is much faster when the number of tasks under the auction is lower, as shown in Fig. 3. As we see in Figure 3, when we increase the number of bids from 100 to 200 we can expect savings of over 10% in the total cost, while 300 bids would lead to an expected savings of over 20% in the total cost.

**Figure 2.** Total cost of the winning bids as a function of the number of available bids for BP with 10, 20 and 30 tasks.

**Figure 3.** Relative cost of auctions with different number of tasks.

### B. Running Time

In a similar setting to the previous experiment, we evaluated the running time of the service selection auction, Fig 4. The running time includes the CPU time of both AMPL (model generation) and CPLEX solve time. It is averaged across the 30 independent problem instances for each factor combination. The diagram clearly demonstrates the exponential time complexity of the combinatorial formulation. However, in the worst case with 300 bids over 30 tasks, the maximum time is around 20 seconds which is quite acceptable.

**Figure 4.** Running time of the composite service selection auction.

### C. The impact of the number of services in a bundle

In the CATS program, the number of items in a bid is calculated this way: starting with 1 item in the bundle, repeatedly increasing the size of the bundle until \( \text{rand}(0, 1) \) exceeds a threshold \( \alpha \). To evaluate how the number of services in a bundle affects the cost of the auction, an experiment with two parameters is designed, Table II. The higher \( \alpha \) is, more crowded the offered bundles will be. Our experiments use the following set of parameters with \#tasks under auction=20.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Level</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Num of bids</td>
<td>100,150, ..., 300</td>
<td>5</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>0.25, 0.5, 0.75</td>
<td>3</td>
</tr>
</tbody>
</table>

As illustrated in Fig. 5, the cost of the composite service increases with having more crowded bundles. Fig. 6 shows that with the least crowded bundles (\( \alpha=0.25 \)), increasing the number of bids from 100 to 300 decreases the cost by over 30%. While this decrease for bundles with \( \alpha=0.75 \) is around 15%.

**Figure 5.** Impact of the number of services in the bundles on cost with non sub-additive pricing function for bundles.
Figure 6. Relative cost of auctions with different number of services in the bundles with non sub-additive pricing function.

The main reason is in the pricing function provided for the CATS suit, which is by default non sub-additive; i.e. no discount is considered for the bundle price. Therefore, we developed a sub-additive pricing function to consider the impact of having more crowded bundles when providers consider a discount over the bundle price. The discounted pricing function for the bundle, $P_d^B$, is defined as $P_d^B = P^B - 0.03(\beta \cdot P^B)$ [24], where $P^B$ is the additive price of the bundle (sum of the prices of the services in the bundle), and the discount rate $\beta$ is derived randomly from $[0,1]$ for each service provider.

As demonstrated in Fig. 7, when providers presents discount over the bundles, having more crowded bundles results in lower cost for the composite service as expected. The effect of $\alpha$ on cost is more pronounced for higher values of the parameter $\alpha$ that controls the size of the bundles.

Figure 7. Cost of the composite service selection auction with sub-additive pricing function for bundles.

D. The impact of the cohesion factor on cost

To evaluate the effect of the cohesion factor on the cost of composite service, we considered two cases:

a) The service requester is interested in high cohesion for the composite service: LC, cohesion lower bound, is at least 50% or 75% of the $MAXC$ (maximum possible cohesion of the composite service),

b) The requester is interested in low cohesion: UC, cohesion upper bound, is at most 25% or 50% of $MAXC$.

We ran each case with the non sub-additive pricing function (Fig. 8) as well as the discounted pricing function (Fig. 9). Totally 1200 problem instances were solved ($\#tasks=20$, $\#providers=50$, $\alpha=0.75$).

TABLE III. EXPERIMENT3 DESIGN PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Level</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Num of bids</td>
<td>100, 150, ..., 300</td>
<td>5</td>
</tr>
<tr>
<td>$LC=0$</td>
<td>0.25, 0.5</td>
<td>2</td>
</tr>
<tr>
<td>$UC=1$</td>
<td>0.5, 0.75</td>
<td>2</td>
</tr>
</tbody>
</table>

Both Fig. 8 and 9 show that for each case, when the cohesion constraint is tighter, the cost is higher. At the same time, achieving a high cohesion (LC at least 50% or 75% of $MAXC$) is much more expensive than achieving a low cohesion in the composite service, independent of the pricing function. This implies that setting maintainability as a non-functional requirement increases the cost of the composite service.

Figure 8. Impact of the cohesion on cost (non sub-additive pricing function).

Figure 9. Impact of the cohesion on cost (sub-additive pricing function).

Figure 10. Success rate of the service selection auction for different cohesion constraints with sub-additive pricing function.
Fig. 10 explains this finding from the point of the number of successful auctions for each case. It illustrates that success rate for auctions with low cohesion preference (LC=0.75) is much lower than that of other cases. It also shows that success rate increases with having more bids in the auction.

To get more detail about the influence of cohesion on the cost, we repeated the experiment increasing the levels of cohesion in the experiment (0.2,0.4,0.6,0.8) for a fixed number of bids (300) and tasks (10) with 50 independent data files (overall 400 problem instances were solved).

The results are shown in Fig. 11. This diagram repeats the previous finding that tighter constraints on both lower and upper bounds increase the cost of the composite service. When seeking higher cohesion, the cost dramatically increase at LC=0.6. The jump happens at UC=0.6 when the objective is low cohesion. These findings can help the service requester to set the appropriate level of cohesion considering the tradeoff in terms of the cost of the composite service.

![Figure 11. Cost as a function of cohesion for #tasks=10, #bids=300](image)

V. RELATED WORK

The two main approaches to service selection for a composite service are optimization-based and negotiation-based service selection [25]. In optimization-based approaches, the service selection problem is mapped to an optimization problem, e.g. integer linear programming [19, 26], genetic algorithm [27], constraint satisfaction [28], constraint optimization [29], and stochastic programming [30]. The objective function is to maximize user satisfaction from the execution of the composite service. This objective is generally translated into maximizing a number of quality attributes, such as availability, security, reliability, while minimizing some others such as price, response time, and delay.

In negotiation-based approaches [31-33], automated negotiators (software programs such as software agents) negotiate on behalf of the service providers and the service requester. Negotiation-based service selection is also referred to as Service Level Agreement (SLA) negotiation where the service provider and the requester negotiate over SLA terms such as QoS attributes, rewards, penalties, and deliverables, in order to come up with a formal SLA at the end of the process [34].

One important distinction between the two service selection approaches is in their underlying assumption about the QoS profile of a web service. The optimization-based approaches typically assume a predetermined, not-customizable QoS profile, while negotiation-based approaches consider the profile to be flexible and negotiable. The fixed profile in the former approach is not very realistic in light of the relatively dynamic environments that characterize the selection and composition of web services. In the negotiation-based approach, a complex decision model is required to build a fully automated negotiator. Application of such negotiators in real world settings is somewhat impractical, at least for the near future. Moreover, the dynamic aspects of negotiation approaches complicate the problem of finding globally optimum solutions.

There is very limited research on the application of dynamic market mechanisms such as auctions for service selection. The combinatorial procurement auction discussed in [35, 36] is an optimization-based service selection technique with the objective function as maximizing the composite service’s quality subject to a budget constraint [35], or minimizing the cost subject to quality and interface matching constraints [36]. The main innovation is that the service provider is able to offer services for more than one task in the BP.

However, the formulation of the problem assumes one quality offer for the bundle of services. Although one price offer for the bundle is reasonable, it is not always possible to define one aggregated quality for the services in the bundle, especially if services are not directly dependent in the structure of the composite service. Our model support individual quality offers for the service bundles as well as the aggregation. Moreover, we have provided the service requester the possibility to benefit from the combinatorial formulation, by enabling them to specify their constraints over the provisioning configuration and data-cohesiveness of the composite service.

A multidimensional procurement auction is proposed in [37] for trading composite services. The objective is to maximize the joint utility of the service requester and providers. There are some restrictions in their mechanism: first, the proposed auction is not combinatorial, although they can form coalitions and offer services jointly. Secondly, each service provider needs to prepare a bid for each combination of her services with possible precedent services from other providers. This is not trivial, especially in the domain of web services and Internet where it is difficult for a provider to obtain enough information about others easily. Finally, the service requester cannot define any constraint over required quality or the budget for the composite service.

VI. CONCLUSION

In this paper, we proposed a combinatorial procurement auction model for composite service selection. The combinatorial formulation benefits both service requester and service provider by allowing them express more complex set of constraints and preferences about the composite service.

We mapped the proposed model to an Integer Linear Problem with objective function as minimizing the cost for the service requester, subject to quality and cohesion...
constraints, and preferences of service requester over provisioning of the composite service. We performed a number of experiments to evaluate the proposed model in terms of the running time and the impact of experiment design parameters on the achieved cost of the optimum solution.

ACKNOWLEDGMENT

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

REFERENCES