The selection of web services in a composition based QoS

Khadija MAHJOUBI
Laboratory TIM, FSBM, Hassan II University of Casablanca, Morocco
khadija.mahjoubii@gmail.com

Hassan LABRIJI
Laboratory TIM, FSBM, Hassan II University of Casablanca, Morocco
labriji@yahoo.fr

Abstract — When an application is web-based, its component tasks can be run using web services. In particular, for the same task, one can discover several web services capable of executing it, one would then face a problem of selection of web services in order to choose the combination of services that offers the best quality of service. For this we give a classification of approaches to our problem.

Keywords- *Service Oriented Architecture; Service Selection; Quality of Service*

I. INTRODUCTION

According to the Service Oriented Architecture (SOA) paradigm, composite applications are specified as abstract processes consisting of a set of tasks. For each task, one can discover several web services that have the same functionality but different in their non-functional settings (quality of service parameters).

The QoS-aware web service composition QoS-based service composition selection aims to find the best combination of web services that meet the user's requirements (QoS constraints). This problem is modeled as an optimization problem, it is associated with the variant of multidimensional multiple choice backpack problem (MMKP: multi-dimension multichoice knapsack problem) [1]. This class of problems belongs to the class of combinatorial problems that are identified as NP-difficult problems [2]. The exact solutions of this problem are exponentially complex, so we can no longer guarantee real-time requirements.

We begin by formalizing the problem of selection then we give a classification of the approaches to our problematic.

II. PROBLEM OF THE SELECTION OF WEB SERVICES

In general, user requests (composed of input concepts and output concepts) are not satisfied by a single web service but by a service composition. The services constituting these different compositions are distinguished from each other by their non-functional properties, such as the Quality of Service criteria (reputation, reliability, execution time, etc.). Quality of Service (QoS) -based selection consists of choosing among the discovered web services of each task, those that best meet the requirements of the user (based on non-functional QoS requirements). The QoS-aware web service composition-based QoS web service selection depends on the specification adopted when defining the QoS criteria and QoS profile of the web service. This problem is an instance of combinatorial problems. It is a multidimensional version of the knapsack problem known to be NP-hard [3]. Therefore, any exact solution to this problem requires exponential execution time, and therefore we cannot guarantee the real-time requirements of users.

III. FORMALIZATION OF THE PROBLEM

More formally, we model the problem as follows. Let:

- CA = {S1, ..., Sn}, an abstract composition representing the user's request, ie the n classes of services Si to consume.
- cg = {cg1, .., cgn}, a set of global constraints defined by the user.
- C = {si1, ..., snj} a concrete composition, ie we replace each class If by a concrete service sij 2 Si
  q (sij) = {q1, .., qm} vector of the qualities of the service sij.
- Q (C) = {Q1 (C), .., Qm (C)} vector of the qualities of the service of the composition C.
- U (C) the objective function (also called fitness) to minimize or maximize. She may be:
  - multi-objective: in this case, U (C) = {U1..Uk} where Uk represents the kth function to optimize.
  - mono-objective: in this case, the m QoS attributes are aggregated into a single value.
- The selection problem is then:

- Global: if we look for all the compositions that will satisfy the constraints global ie Qk _ cgk, 8k 2 {1..n}.

- Local: if we search all the compositions without taking into consideration the global constraints cg to optimize, we seek to optimize the QoS parameters in each abstract class independently of the others. If we assume that the number of candidates per class is L, then the overall number of possible compositions is Ln.

IV. State of the Art

Several works have been proposed to solve the problem QoS-aware service composition. In the literature, several classifications of approaches are found according to whether the function to be optimized is monoobjective, multiobjective or hybrid [4], depending on whether the selection strategy used is local or global [3] or according to the approach used. In the latter case we distinguish 04 large classes [5, 6]:

1. The exact methods (non-heuristic).
2. Heuristic methods (approximate).
3. Methods based on meta-heuristics (approximate).
4. Pareto dominance-based approaches.

In what follows we present in detail the two exact and heuristic approaches by relying on several papers of the literature of the problem of selection of composite services [5-7].

A. Exact approaches

Exact methods solve the optimization problem optimally.

They use the techniques of constraint programming or dynamic programming, integer Linear Programming Portion (ILP) or Mixed Integer Programming (MIP). These approaches give optimal results but they have an exponential execution time. Zeng et al. [8], [9] focus on dynamic selection based on quality of service. Authors use global planning to find the best web service compositions. They use Mixed Integer Programming (MIP) [10] techniques to find the optimal mix of services.

In the same vein, Ardagna et al. [11], [12] extend the linear programming model to include local constraints. In this model, global constraints are expressed on the entire composition, and by the end user, while local constraints can be specified by the composition designer (at the class level). Alrifai et al. [13] proposed a two-step solution:

First, they use full and mixed programming (MIP) to find the optimal decomposition of global QoS constraints into local constraints. Second, they use a distributed local selection to find the best web services that satisfy these local constraints

. In [14], the authors proposed a model that treats selection as a constraint optimization problem (Constraint Optimization Problem). They relaxed the problem by a weighting of constraints, the goal is to find a solution maximizing a function of weighted constraints.

The main idea in their proposal is that instead of having all the constraints to satisfy, some constraints are flexible and optional. Gao et al. [15] propose the entire programming taking into account the conflict between services (which belong to different classes). Conflicts mean that the use of class X service i is not compatible with class Y service j.

Experimentation shows that the addition of conflicts causes a 13% increase in the execution time of the MIP-based approach. Xu et al. [16] propose a method for selecting QoS-based web services by considering a variation of QoS values as a function of the time interval (the fluctuations are given by the service providers). For this the user introduces the weights and the range of interest that can encompass multiple intervals with different QoS values, but they do not handle global constraints. For this, they use dynamic programming that cuts the composition in a series of pairs of services (02 consecutive classes), the algorithm looks for the concrete pair that minimizes the objective function. [16] Propose a method for finding the solution for a service composition by using a dependency graph and 0-1 LIP. Liu et al. [17] propose a convexhull method and apply the multi criteria decision making method (MCDM) to merge multiple resources for global and local constraints.

The author in [18] confirms that the methods based on the entire programming are effective if the service number L is small, on the other hand if L exceeds a certain limit then the execution time is no longer reasonable and therefore these methods do not scale.

B. Heuristic approaches (approximate)

Heuristic approaches use rules and empirical functions adapted to specific optimization problems. They try to find a good solution of the problem in an acceptable time. The solution obtained is not the optimal solution but an approximate solution [19]. The heuristic algorithms take advantage of the peculiarities of the problem, unlike exact algorithms that take a long time to obtain the optimal solution; the heuristic algorithms obtain almost optimal solutions in a reasonable time [5].

Several works are inspired by heuristic algorithms. [20] proposes one of the first algorithms for solving the backpack problem (and its MMKP-multi-dimensional multiple choice knapsack problem- version). The author proposes a heuristic named UHE for its resolution; UHE uses a measure called the consumption of aggregated resources, to update an element of each group at each round of selection. The authors in [21] modified the heuristic by creating M-UHE, they propose a pretreatment step to find a feasible solution and a postprocessing step to improve the total value of the solution, for that they make improvements (upgrades) by updating an element that increases the total utility, then they degrade the solution with one or more hits (downgrades), ie they select an element that decreases the value of the total utility. [22] propose another heuristic, CUHE, and evaluate its performance and optimality against several heuristics, including the M-UHE algorithm. The results show that C-UHE is better compared to M-UHE in terms of execution time.

However, the experiments also show that M-UHE is the best in terms of optimality, while the optimality of C-HEU decreases as the number of candidates per class increases. The experiments also show that the C-UHE algorithm works better in the case where the objective to be maximized (for example, the utility value of the service composition) is not proportional to the resource requirements (ie, the QoS values of web services). And therefore it cannot be applied for the selection of quality-based composite services (since the utility depends on the quality of service).

A modified version of the M-UHE algorithm, called WS-UHE, is designed by [23]. The authors propose two models for the service selection problem:

(1) a combinatorial model and (2) a graph model. A heuristic algorithm is introduced for each model: the WS-HEU algorithm for the combinatorial model and the MCSP-K for the graph-oriented model. The time complexity of WSUHE is polynomial, while the complexity of MCSP-K is exponential. In spite of the significant improvement of these algorithms compared to exact solutions, the two algorithms do not pass to the scale anymore (ie if the number of Web services grows in a dramatic way then the execution time is no more real time). Moreover, the WS-UHE algorithm is not adapted to the distributed nature of web services. This is because WS-HEU applies enhancements to a service composition whose QoS values come from multiple facilitators (Each facilitator maintains a given class).

Klein et al. [24] use the hill climbing algorithm to reduce computation time complexity and compare their method with that using LIP. Qi et al. [25] presented a local optimization and enumeration method to find local candidates and then combine them to find the optimal solution. Diana et al. [26] proposed a heuristic search model for QoS-based service composition selection.

Li et al. [27] discussed an efficient and reliable approach for selecting the optimal composition of services based on confidence parameters.

The authors in [28] propose a global selection of service composition by adopting 4 control flowstructures: parallelism, sequence, conditional choices, and loops, the request is formalized in BPEL, it has 5 QoS criteria. The algorithm is called qssac, it can give a result close to optimal. To reduce execution time, they group similar services in terms of QoS using an algorithm called optics (which is less sensitive to noise and has few initialization problems, optics is based on density) , we get M classes (and their representatives) for each task, then the algorithm lists all possible compositions for 2 or K tasks of the BPEL document, and sorts them using an objective function. Other heuristics have been proposed by [29]: H1 RELAX IP, H2 SWAP, and H3 ANNEAL to find the optimal solution and improve the efficiency in the problem of selection of web services. Luo et al. [30] proposed a Heuristic HCE algorithm for the composition of web services taking into account the global constraints of the user.

In [31] the authors consider the QoS as non-deterministic data, to select the quasi-optimal compositions the approach adopts the probability of satisfying the global constraints as objective function to maximize, the authors divide the global constraints into local constraints avoiding outlier values. The division uses the minimal QoS of each abstract class during the computation of local constraints.

Several other works have proposed several heuristic algorithms [32], [33], [34] and [35] to reduce the complexity of time related to global and local constraints in the selection of optimal composition

CONCLUSION

In this work, we presented the problem of selecting services based on QoS. We also highlighted, a state of art illustrating research work on this issue. Specifically, we presented the works that belong to both classes, namely the exact methods and methods heuristics.

In general, we can conclude that metaheuristics are better adapted to this problem than others, since the exact methods no longer pass to scale. More, heuristic approaches are not generalizable for all kinds of objective functions. On the other hand, Pareto-based optimization is no longer effective when the number of QoS criteria is large.

Therefore the best class of algorithms that manages all these compromises (given number of QoS attributes, polynomial complexity, quasi-optimality of solutions, adaptability) will be the class of metaheuristics

REFERENCES

[1] Yu, T., Zhang, Y., and Lin, K.-J. (2007). Efficient algorithms for web services selection with end-to-end qos constraints. ACM Transactions on the Web (TWEB), 1(1) :6.

[2] Parra-Hernandez, R. and Dimopoulos, N. J. (2005). A new heuristic for solving the multichoice multidimensional knapsack problem. IEEE Transactions on Systems, Man, and CyberneticsPart A :Systems and Humans, 35(5) :708–717.

[3] Alrifai, M., Risse, T., and Nejdl, W. (2012). A hybrid approach for efficient web service composition with end-to-end qos constraints. ACM Transactions on the Web (TWEB), 6(2) :7.

[4] Hadjila, F. (2014). Composition et interopération des services web sémantiques. PhD thesis.

[5] Jatoth, C., Gangadharan, G., and Buyya, R. (2015). Computational intelligence based qos-aware web service composition : A systematic literature review. IEEE Transactions on Services Computing.

[6] Khanouche, M. E., Amirat, Y., Chibani, A., Kerkar, M., and Yachir, A. (2016). Energy-centered and qos-aware services selection for internet of things. IEEE Transactions on Automation Science and Engineering, 13(3) :1256–1269.

[7] KAMAL, S., IBRAHIM, R., and GHANI, I. (2014). Review on service selection schemes based on user preferences. Journal of Theoretical & Applied Information Technology, 70(2).

[8] Jatoth, C., Gangadharan, G., and Buyya, R. (2015). Computational intelligence based qos-aware web service composition : A systematic literature review. IEEE Transactions on Services Computing.

[9] Zeng, L., Benatallah, B., Dumas, M., Kalagnanam, J., and Sheng, Q. Z. (2003). Quality driven web services composition. In Proceedings of the 12th international conference on World Wide Web, pages 411–421. ACM.

[10] Zeng, L., Benatallah, B., Ngu, A. H., Dumas, M., Kalagnanam, J., and Chang, H. (2004). Qos-aware middleware for web services composition. IEEE Transactions on software engineering, 30(5) :311–327.

[11] Nemhauser, G. L. and Wolsey, L. A. Integer programming and combinatorial optimization. Wiley, Chichester. GL Nemhauser, MWP Savelsbergh, GS Sigismondi (1992). Constraint Classification for Mixed Integer Programming Formulations. COAL Bulletin, 20 :8–12.

[12] Ardagna, D. and Pernici, B. (2005). Global and local qos constraints guarantee in web service selection. In Web Services, 2005. ICWS 2005. Proceedings. 2005 IEEE International Conference on. IEEE.

[13] Ardagna, D. and Pernici, B. (2007). Adaptive service composition in flexible processes. IEEE Transactions on software engineering, 33(6).

[14] Alrifai, M. and Risse, T. (2009). Combining global optimization with local selection for efficient qos-aware service composition. In Proceedings of the 18th international conference on World wide web, pages 881–890. ACM.

[15] Rosenberg, F., Celikovic, P., Michlmayr, A., Leitner, P., and Dustdar, S. (2009). An end-to-end approach for qosaware service composition.In Enterprise Distributed Object Computing Conference, 2009. EDOC'09. IEEE International, pages 151–160. IEEE.

[16] Gao, A., Yang, D., Tang, S., and Zhang, M. (2006). Qosdriven web service composition with inter service conflicts. Frontiers of WWW Research and Development-APWeb 2006, pages 121–132.

[17] Xu, L., Shi, L., Wang, R., and Jennings, B. (2011). A multiple criteria service composition selection algorithm supporting time-sensitive rules. In Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on, pages 718–721. IEEE.

[18] Gabrel, V., Manouvrier, M., and Murat, C. (2014). Optimal and automatic transactional web service composition with dependency graph and 0-1 linear programming. In International Conference on Service-Oriented Computing, pages 108–122. Springer.

[19] Liu, D., Shao, Z., Yu, C., and Fan, G. (2009). A heuristic qos-aware service selection approach to web service composition. In Computer and Information Science, 2009. ICIS 2009. Eighth IEEE/ACIS International Conference on, pages 1184–1189. IEEE.

[20] Maros, I. (2012). Computational techniques of the simplex method, volume 61. Springer Science & Business Media.

[21] Yang, X.-S. (2009). Harmony search as a metaheuristic algorithm. In Music-inspired harmony search algorithm, pages 1–14. Springer.

[22] Khan, M. S. (1998). Quality adaptation in a multisession multimedia system : Model, algorithms and architecture. PhD thesis, University of Victoria.

[23] Akbar, M., Manning, E., Shoja, G., and Khan, S. (2001). Heuristic solutions for the multiple-choice multi-dimension knapsack problem. Computational science-ICCS 2001, pages 659–668.

[24] Akbar, M. M., Rahman, M. S., Kaykobad, M., Manning, E. G., and Shoja, G. C. (2006). Solving the multidimensional multiple-choice knapsack problem by constructing convex hulls. Computers & operations research, 33(5) :1259–1273.

[25] [23] Yu, T., Zhang, Y., and Lin, K.-J. (2007). Efficient algorithms for web services selection with end-to-end qos constraints. ACM Transactions on theWeb (TWEB), 1(1) :6.

[26] [24] Klein, A., Ishikawa, F., and Honiden, S. (2011). Efficient heuristic approach with improved time complexity for qos-aware service composition. InWeb Services (ICWS), 2011 IEEE International Conference on, pages 436–443. IEEE.

[27] [25] Qi, L., Tang, Y., Dou, W., and Chen, J. (2010). Combining local optimization and enumeration for qos-aware web service composition. In Web Services (ICWS), 2010 IEEE International Conference on, pages 34–41. IEEE.

[28] [26] Comes, D., Baraki, H., Reichle, R., Zapf, M., and Geihs, K. (2010). Heuristic approaches for qos-based service selection. In International Conference on Service-Oriented Computing, pages 441–455. Springer.

[29] [27] Li, J., Zheng, X.-L., Chen, S.-T., Song, W.-W., and Chen, D.-r. (2014). An efficient and reliable approach for quality-ofservice-aware service com composition. Information Sciences, 269 :238–254.

[30] [28] Xia, Y., Chen, P., Bao, L., Wang, M., and Yang, J. (2011). A qos-aware web service selection algorithm based on clustering. In Web Services (ICWS), 2011 IEEE International Conference on, pages 428–435. IEEE.

[31] Berbner, R., Spahn, M., Repp, N., Heckmann, O., and Steinmetz, R. (2006). Heuristics for qos-aware web service composition. InWeb Services,2006. ICWS'06. International Conference on, pages 72–82. IEEE.
[32] Luo, Y.-s., Qi, Y., Hou, D., Shen, L.-f., Chen, Y., and Zhong, X. (2011). A novel heuristic algorithm for qos-aware end-to-end service composition. Computer Communications, 34(9) :1137–1144.
[33] Kim, M., Oh, B., Jung, J., and Lee, K.-H. (2016). Outlierrobust web service selection based on a probabilistic qos model. International Journal of Web and Grid Services, 12(2) :162–181.
[34] Luo, Y.-s., Qi, Y., Shen, L.-f., Hou, D., Sapa, C., and Chen, Y. (2008). An improved heuristic for qos-aware service composition framework. In High Performance Computing and Communications, 2008. HPCC'08. 10th IEEE International Conference on, pages 360–367. IEEE.
[35] Li, J., Zhao, Y., Liu, M., Sun, H., and Ma, D. (2010). An adaptive heuristic approach for distributed qos-based service composition. In Computers and Communications (ISCC), 2010 IEEE Symposium on, pages 687–694. IEEE.
[36] Li, Y.-Q. and Wen, T. (2012). An approach of qosguaranteed web service composition based on a win-win strategy. In Web Services (ICWS), 2012 IEEE 19th International Conference on, pages 628–630. IEEE.
[37] Li, M., Zhu, D., Deng, T., Sun, H., Guo, H., and Liu, X. (2013). Gos : a global optimal selection strategies for qosaware web services composition. Service Oriented Computing and Applications, 7(3) :181–197.