ACIT'2017
The International Arab Conference on Information Technology
Yassmine Hammamet, Tunisia
December 22-24, 2017

A comparative analysis of IoT service composition approaches

Idir AOUDIA, Saber BENHARZALLAH, Laid KAHLOUL and Okba KAZAR LINFI Laboratory

Biskra University 07000 Biskra, Algeria

***Abstract*—The Internet of things is the integration of information space and physical space, becoming more and more popular in several places. A number of approaches to IoT's service composition have been proposed. In this paper, we will present a review of existing approaches for IoT's service composition; it describes and compares them among each other with respect to some key requirements. This paper also presents some basic concepts to introduce IoT challenges with a comparison between traditional Web service composition and IoT service composition. This paper represents a support for researchers to focus on their efforts and to deliver lasting solutions in this field.**

***Keywords*— Internet of Things; service composition; adaptability; context.**

I. Introduction

The Internet of Things (IoT) is an emerging technology which can alter the industry, environment, social and medical field. IoT is a fusion between information space and physical space, known as the interconnection of embedded computing, sensors, mobile devices, or other uniquely identifiable objects, which are able to interact with each other and cooperate with the surrounding environment to reach common goals, by exploiting the existing Internet infrastructure.

The emergence of IoT has engendered a complexity in IoT„s management due to the heterogeneity since the involvement of more smart devices. In order to enhance the homogeneity a common adopted way based on the encapsulation of heterogeneous devices by web services [1]. The concept of service has grown strongly on the Web where Web services are the basic components of any treatment in a service-oriented architecture (SOA). Their principles based on the description, publication, research and invocation. They provide modularity, scalability, composability and reusability. The services composition consists in combining the functionality of multiple services within a single business process in order to answer complex demands with a single service couldn"t satisfy [2].

A. Challenges of service composition in IoT

IoT has many challenges, heterogeneity and the rapid increasing number of devices are one of them, the encapsulation of heterogeneous devices as homogeneous web services, uses mainly the traditional triangular SOA operational model that register the devices in a centralized Universal Description, Discovery, and Integration (UDDI) registry, the number of devices registered increases exponentially leading to decrement of service management. As a consequence service management scheme should be designed from centralized to distributed, and thus brings scalability and robustness [1,2].

The cost of service composition in IoT needs to be considered, because IoT„s services are mainly deployed in resource-constrained devices with limited energy storage and processing capability and that"s comparing to internet services which are deployed in computers with rich resource and powerful function [4].

A monitoring mechanism should be used to ensure fault tolerance and to monitor the IoT system. Because the information of IoT"s service is instable (its state switches between availability/unavailability) comparing to internet services [4]. It is most ideal to allocate a resource to each monitoring device atomic. However, this can lead to a high cost [5]. The motivations for the adaptation of service are summarized in two points: ensure continuity of services to the user and improve the user interaction with the environment. It is suitable for IoT to introduce a QoS strategy to choose and compose sensory information service, and that"s because IoT information service composition aims to provide user with sensory information. Contrary to internet service composition who aims to solve programmatic issue, focusing on the demonstration of feasibility and logicality [4]. An IoT system connects the physical world in cyberspace, thus one of the challenges of the IoT system is the management of trust system to resist malicious attacks in order to survive in hostile environments [6].

Among the above challenges, the following three aspects are considered the most difficult:

1) *Finding the most optimal composition.*

2) *Protecting and monitoring the devices with restricted monitoring resources in dynamic environment.*

3) *Managing dynamic change of the environment and resources.*

B. The application of standard Web services on IoT's services

Recently, SOA becomes the mainstream of development of information and integration, it is widely used in various combinations of business services in many work of national and international researchers in the world. SOA approach opts for the use of accessible, open and described software components, offering atomic services [7]. The architecture thus produced is then decoupled, meaning that the different elements exchange information while they stay very little dependent on each other and only related by the contract stipulating the way of communication [8]. The services composition requires that services must be described in a standard way in a directory, and offer rich, accurate and subject exchange formats (or encapsulation of exchange) submitted to strict rules but easily scalable, to allow the exploitation of available resources. As far as this approach is meaningful in the construction of a service which will be offered to very numerous users, as in IoT relationships evolve, and the organization is brought to be rethought. The IoT‟s applications involve personal elements which have to answer individual problems. The intervention of an architect is then required for the construction of a solid aggregation of services aimed the widest population of users as possible. Each IoT‟s application may be used only by its designer and his family, because it‟s adapted more specifically to their use, and/or some of their objects. In IoT, the circuit designer/consumer is reduced, and sometimes these roles are played by the same person [9].

In IoT and highly constrained environments, it seems appropriate to disqualify SOAP[1], including its heaviness to the service it is supposed to make. Most of the current operations can be performed through HTTP, more simply, lighter and concisely. Brevity and lightness of exchanges are exactly what IoT requires, and that's what RESTful [10] approach offers.

The choreography and the orchestration of services are two different approaches that we wonder about their recoveries; if they express the same rules and satisfy the same needs. A study [11] was interested in the use of choreography under SOAP and REST, and compares their strengths and weaknesses facing the workings of the choreography. REST provides a weak coupling and clarity of the definition of the desired operations that appear in the message, while SOAP is used to describe the complexity of certain transactions and provides monitoring of exchanges, particularly interesting to debug the cinematic dialogue. The approach choreography is adapted to IoT; it allows reducing the use of network resources involved in IoT. The choreography respects the energy constraints, because the direct interactions between nodes require less network then data feedback to a central point. In addition, the processing capabilities added to objects (admittedly limited, but present) allow the implementation of the algorithms describing the task to be performed on the object itself [9].

Practice shows that the direct application of standard Web services on IoT‟s services do not fit, and will bring new problems; because IoT‟s services are different from traditional services and their environment is dynamic. The biggest difference between traditional and IoT‟s services is that traditional service is a virtual entity but IoT‟s services are directly related to the physical world [12], providing services that satisfies the user and that constitute one of colossal problematic of IoT.

The remainder of this paper is organized as follows: in section II, we introduce some basic concepts which have a relation with our work. In section III, we present different approaches, already, proposed for service composition in the IoT. After the presentation of the existing approaches, section III proposes a comparative study of these approaches based on specific criteria. Section IV presents a discussion and in section V, we review the related work. Finally, we conclude the paper in section VI.

[1] Simple Object Access Protocol (https://www.w3.org/TR/2000/NOTE-SOAP-20000508/)

II. Basic Concept

A. Services

A service is a mechanism that is capable of providing one or more functionalities. A service can be used in compliance with provider-defined restrictions and rules and through an interface [13]. Services for smart environments are generally considered to be extensions of existing Web services (for example, messaging, location services, device management, multimedia applications...) while repeating the basic principles. Two categories of services can be distinguished: the user services and software services. The context-aware services know a big development nowadays, for example in safety, control of privacy and the adaptation of content. There are four major families of services: content services, monitoring services, communication services and e-commerce services [14].

B. RESTful

The popular style RESTful [10] is not a protocol, but rather an architectural approach, which aims to facilitate the programming of services-oriented applications using HTTP. REST puts the accent on the roles of client and server, the identification of available resources through a single address, and exchanges do not require states. So for tracking all actions and their contextualization, the exchanges must be self-sufficient and not depend on prior actions or weigh on future reactions. All framing information to/from the IoT's devices goes in the Uniform Resource Identifier (URI) [15], what makes possible for users or upper-layer applications to access devices via HTTP / HTTPS protocol a unified and flexible manner [3].

C. The orchestration and choreography

The orchestration of services is the conception of an algorithm under the control of a central point, responsible for the chain of service calls and management of returned results. Instead, the choreography is an approach in which each service interacts with others without being placed under the responsibility of a main node. The choreography focuses more on the relationships between different services, meaning, the causes, dependencies and exclusions. The choreography does not describe internal effects (while orchestrating focuses on the internal reasoning of the main node) but rather the reactions of each of the services submitted to exchanged stimuli [9].

III. Existing Approaches for Service Composition in IoT

This section presents some popular proposed approaches for service composition in IoT and then a comparative study of these approaches.

A. Approaches

In this part, we are going to present several approaches already proposed [3-6, 12, 20, 22, 23-24, 26, 28] for service composition in the IoT.

In [3], a distributed social network based approach for IoT device management and service composition has been proposed. The authors encapsulated IoT devices into Web services using RESTful style; they modeled the relationships between IoT services by social networks, and classified it into three dimensions which are location, type and correlation. Schemes of service registration, selection and management are carefully designed for each of these three dimensions, and thus the social network are divided into three separate sub networks for efficient management and accelerated service search. The authors proposed a flexible and scalable mechanism for IoT service composition; taking the social relationships between IoT services into account, it allows for the automatic collaboration of heterogeneous IoT devices in order to meet user requirements.

In [4], a QoS method for service composition has been proposed, the authors used a Multi-attribute decision-making (MAMD) [19] to calculate QoS performance and evaluate each individual service. According to the QoS performance and function value of service, several services are chosen to be in the composition. The authors divide service composition into user demand analysis, service searching and matching, service selection and finally service composition, and an improved Genetic Algorithm is used to find the optimal service composition solutions.

In [5], a novel Petri net-based [16]–[18] composition algorithm (*FindTOptimal*) has been proposed to find the most optimal path, which uses a comprehensive performance function named *(rtc)* to evaluate the cost-effectiveness. To manage dynamic change of the environments, the authors proposed a monitoring algorithm (*FbasedMonitor*) to monitor the IoT system in a cheapest way.

In [6], an adaptive trust-based social approach for the service composition has been proposed, which aims to solve the design and validation of an adaptive and survivable trust management protocol for SOA-based social IoT systems [21]. A user performs trust evaluation based on its past direct satisfaction experiences and trust feedbacks from other users sharing similar social interests. The authors divide social relationships into three lists and each user has at least one designated high-end device (i.e., smart phone and laptop) storing these lists in the user‟s profile. Other devices of the same user have the privilege to access the profile. By delegating the storage and computation of social networks to a high-end device for each user, many low-end devices (i.e., sensors) are able to share and utilize the same social information to maximize its performance. The authors developed an adaptive filtering technique to find the best way to combine direct trust and indirect trust feedback dynamically, allowing each node to adaptively select its best trust parameter to minimize convergence time and trust bias.

In [12], a context-based method for service composition has been proposed, where the authors used OWL to build the ontology of context in the environment of the IoT. In order to reduce the scope of context information, the number of common concepts and attributes in different sub domain, the authors divide service discovery field into different sub areas. In service composition, process of the service selection was divided into two sub-processes. First, the authors use computational context to select the appropriate services. Then, according to quality of service from user expectations, they choose the best services for service composition to meet the user‟s needs. Thus, the authors used the contextual information to optimize the service composition in a particular context, to find the most compatible with the needs of service users while satisfying the contextual constraints.

In [20], a middleware approach for the service composition of logistics services in IoT has been proposed. A middleware which consists of resource agents and task agents is used to designate at runtime candidate component services to participate in the composition of a logistics transaction. Each component service is represented by a resource agent, which is responsible for maintaining the QoS information of the corresponding service. The task agents represent composite service‟s instances and are responsible for finding component services which satisfy the user‟s end-to-end QoS requirements. The authors use just a few monitoring resources to monitor and ensure the operations and guarantee the robustness of the logistics system.

In [22], a probabilistic approach has been proposed to formally describe and analyze the reliability and cost-related properties of the service composition in IoT. The authors developed an approach to model the reliability and cost of service composition on the basis of the MDP[2] with cost structure. So, the probabilistic model checking can be applied to verify and analyze the quality properties of the service composition. By using this approach, the success probability of the composite service can be calculated with the current IoT settings of the devices. And furthermore, the cost value for each candidate composite service can be obtained. The developer of IoT services can use the results to make the decision about the service selection and the results can alert them to decide if they need to deploy alternative candidate service in case of the failure of the selected service when the success probability of the selected service is not very high.

In [23], a dynamic service composition approach has been proposed, which is based on the online generation and rescheduling of optimal plans for services orchestration by using a heuristic planning technique. The proposed selection mechanism is based on an optimization technique that detect situations where some composition requirements are not satisfied or some services become unavailable or when failures/exceptions happen. It has the ability to operate despite the changes that occur in the behavior of the selected services and reduces consequently the need for human intervention in the reconfiguration of the composition.

In [24], a lightweight asynchronous RESTful Web services composition approach for service composition in IoT, which is based on the BPEL extention [25]. The authors divided the

[2] Markov Decision Process

architecture into six layers, and added asynchronous interaction to save time and improve performance.

In [26], a lightweight REST service composition framework for IoT based on functional programming with monads[3] has been proposed. To manage dynamic needs and the large number of resources, the authors treated resources as distributed objects with local states [27] that can be accessed by uniform interface through hypertext, such as the HTTP 1.1 GET, PUT, POST, and DELETE methods. Since the output of these methods depends on the resource states, they cannot be modeled as pure functions whose output only depends on the input. For this reason, the authors use monads to access the REST resources to be composed.

In [28], a framework for IoT services composition in IoT has been proposed. The authors take into account the reputation of the service provider and the reliability of the reputation provider in order to identify the best candidates to the creation of a composed service, to cope with a given task. A provider reputation is computed locally, so every provider may compute a different reputation based on his/her previous personal experiences with different providers (asymmetry). This approach is computationally cost and do not manage connection failure, not very suitable for big environment like an IoT service composition scenario.

B. A comparative study between approaches

In this section, we present a comparative study “TABLE I” between the different approaches already presented. Our comparison is based on the following criteria:

1) *Adaptivity:*

   The composition must ne adaptive, to manage dynamic changes of the environment and resources, the composition may use monitoring, fault tolerance system or some reasoning methods.

2) *Autonomy and independence:*

   The composition‟s entities must Autonomous and be described independently without assuming the existence or absence of the others.

3) *Decentralization and distribution:*

   Service composition system in IoT must be decentralized and distributed to manage the growing number of devices.

4) *Protection and trust management:*

   One of the challenges of the IoT‟s service composition system is the management of trust system to resist malicious attacks in order to survive in hostile environments, because an IoT system connects physical world in cyberspace [6].

5) *The optimization of the composition:*

   The optimization uses some constraints to improve behavior of the composition and its components, like response time, real time and resources constraints (energy and processing capacity).

[3] A monad is a structure for manipulating pure functional languages with imperative traits.

TABLE I. COMPARATIVE STUDY BETWEEN APPROACHES ALREADY PROPOSED.

	Adaptivity	**Autonomy and independence**	**Decentralization and Distribution**	**Protection and trust management**	**The optimization of the composition**		
					Resource constraint	**Response time and real time constraint**	**Approach used**
[3]	-	+	+	-	-	-	*A social network based approach:* -Social network. -Location, type and correlation. -Searching processes are implemented in a parallel way.
[4]	-	+	- (Use a service repository)	-	+	+	*Improved genetic algorithm:* - Calculation of QoS performance based on Multi-attribute decision-making (MAMD). - Price, response time, reliability, reputation and geographic location.
[5]	+ Monitoring system (FbasedMonitor)	+	+	-	+	+	*Petri net approach based on RTC function:* -Reliability, Response time and cost. - Cost-effectively
[6]	+ (Adaptive filtering technique)	+	+	+	+ (high-end device for each user)	-	*Adaptive & survivable trust evaluation system:* -A friend list, a location list and a list with devices (services) directly interacted with. -Storing in a high-end device.
[12]	+ Reasoning: Context modeling with OWL	+	+	-	+	+	*Method based on context:* -User context (QoS). -Physical environment. -Computing environment.
[20]	+ Monitoring system (decentralized coordinating mechanism)	+	+	-	+	+	*A middleware approach:* -Resource agent: maintaining the QoS information. -The task agent.
[22]	+ Monitoring system (Alternative candidate in case of the failure)	+	- (Orchestration)	-	+	-	*Approach to model the reliability and cost:* -Based on MDP.
[23]	+ Monitoring and fault tolerance	+	- (Orchestration)	-	- -Extremely demanding - Intensive Computing	-	*Optimization technique that can detect situations where some composition requirements are not satisfied or some services become unavailable or when failures/exceptions happen.* -Heuristic planning technique.
[24]	-	+	+	+	-	-	*Extending BPEL approach and asynchronous interaction :* -Save time and improves performance.
[26]	- (Error processing and event monitoring for future work)	+	+	-	+	-	*Using a collection of nomads (functional programming) arranged to access a set of REST resources to be composed.*
[28]	- (Monitoring system for future work)	+	- (Decentralize the reputation computation for future work)	-	- -Extremely demanding	-	*An approach based on the reputation of the service provider and the reliability of the reputation provider* -The approach does not focus on mechanisms for service discovery or service specification languages.

(+) Satisfy

(-) Unsatisfied

IV. DISCUSSION

"TABLE I" shows that some Approaches adopt an adaptive composition to manage dynamic changes of the environment and resources, like in [5], [20], [22] and [23] the authors implement a monitoring and a fault tolerance system, But [12] used a reasoning approach, and [6] a Adaptive filtering technique to manage it. "TABLE I" shows also that all approaches satisfy the autonomy and independence criteria and the most of them are distributed and Decentralization aside from [4], [22], [23] and [28].

The second part of "TABLE I", we can note that the most of approaches satisfy resource constraint challenge, except [3], [23] and [28]. Concerning response time and real time constraint, only [5], [4] and [20] satisfy that. Protection and trust management are the major drawbacks of all the approaches, only [6] and [24] satisfy these two criteria. Concerning the optimization of service composition, approaches like [12], [5], [4] and [20] implement methods based on QoS optimization. Also there is approaches like [6], [22] and [28] which are based on the reputation, reliability and trust. Finally the searching processes like in [3] or [6] are implemented in a parallel way by using some social approach.

Petri net approach [5], *middleware approach* [20] and the *approach based on context* [12] (which uses reasoning method OWL), are the best approaches adapted to the environment to IoT environment, however all this approaches don"t satisfy protection and trust management criteria. *Adaptive & survivable trust management approach* [6], demonstrated that the application is able to approach the ideal, however the author didn"t focus on the others aspects managed by the three approaches mentioned above.

V. RELATED WORK

In the best of our knowledge, no review has been produced in the service composition for Internet of things field. The closest works that have been done are on the side of the composition of Web service.

In [29], a comparative study of existing approaches for web service composition has been described, the authors compare them among each other with respect to some key requirements. Also in this paper, the authors provide an overview of service composition methods and approaches, and they discuss the advantages and disadvantages of each investigated paper.

In [30], an overview and comparison of recent progress in web service composition has been provided. The authors classify these approaches on three categories (Workflow-based, XML-based, and Ontology-based). In each category, they give the introduction and comparison of selected approaches based on some benchmarks (like QoS, scalability, and correctness).

In [31], a study of different Web service composition techniques based on currently existing composition platforms and frameworks has been done. The authors compare these techniques based upon the merits and demerits. They discuss what makes Web service composition so special and derive challenges for business community.

In [32], an overview of the recent progress made in Web Service discovery approaches has been provided. Furthermore, the authors performed an analysis over these approaches and highlighted some of their merits as well as shortcomings. After introducing a taxonomy which categorizes Web Service discovery systems from different points of view, they presented the advantages and disadvantages of each group.

In [33], a study of different Web Service selection techniques which are available in market has been done. The authors compare these Web service selection methodologies using different parameters (QoS and Reliability).

In [34], an overview of state of the art research in semantic Web composition approaches has been discussed. The authors compared and categorized such approaches into two categories; semantic Web composition approaches with Quality of Services (QoS) support and semantic Web composition approaches without QoS support. Six evaluation criteria were proposed for the purpose of systematically comparing the two approach categories to semantic Web services composition; scalability, non-determinism, dynamic aspect, adaptability, domain independence, correctness, semantic capability and QoS awareness. In each category, they give the introduction and comparison of selected approaches. The objective of the paper was to identify the best approach that can be used for semantic Web services composition.

In [35], an articulated framework for analyzing and comparing Web service composition approaches has been proposed. The authors proposed a taxonomy that consists of the dimensions (language, knowledge reuse, automation, tool support, execution platform, target user) that characterize and compare service composition approaches. They provided a systematic analysis of the most representative service composition approaches by evaluating and classifying them against the proposed taxonomy.

VI. CONCLUSION

In this paper, we have described a comparative study of recent approaches for service composition for IoT, with respect to some key requirements based on the most important challenges of IoT: The management of heterogeneity (autonomy/independence, decentralized, the optimization of the composition, protection and trust management for the management of the growing number of devices), monitoring and fault tolerance and reasoning for the management of dynamic change of the environment and resources, resource constraint (energy and processing capacity), response time and real time constraint. However, we cannot claim that this classification is very exhaustive.

We have noticed that the most of approaches don"t satisfy all our criteria and some fundamental aspects are commonly ignored or poorly discussed, such as response time, real time constraint, monitoring and fault tolerance, we, finally, argued protection and trust management are the major drawbacks of all the approaches. The lack of available implementations is an issue, since these aspects are very important. It raises opportunities for the improvement of the existing service composition approaches strategies for IoT, as well as for future research in the area. We believe that service composition mechanisms for IoT would benefit from deeper research into the aforementioned themes.

REFERENCES

[1] L.-J. Zhang, J. Zhang, and H. Cai, "Services Computing," *Springer Berlin Heidelb.*, 2007.

[2] D. Berardi, D. Calvanese, G. D. Giacomo, M. Lenzerini, and M. Mecella, "A foundational vision of e-services," *C. Bussler, D. Fensel, M. E. Orlowska, J. Yang, Ed. Vol. 3095 Lect. Notes Comput. Sci.*, pp. 28–40, 2003.

[3] G. Chen, J. Huang, B. Cheng, and J. Chen, "A Social Network Based Approach for IoT Device Management and Service Composition," *2015 IEEE World Congr. Serv.*, pp. 1–8, 2015.

[4] Z. Yang and D. Li, "IoT information service composition driven by user requirement," *Proc. - 17th IEEE Int. Conf. Comput. Sci. Eng. CSE 2014, Jointly with 13th IEEE Int. Conf. Ubiquitous Comput. Commun. IUCC 2014, 13th Int. Symp. Pervasive Syst.* , pp. 1509–1513, 2015.

[5] R. Yang, B. Li, and C. Cheng, "A Petri Net-Based Approach to Service Composition and Monitoring in the IOT," *2014 Asia-Pacific Serv. Comput. Conf.*, pp. 16–22, 2014.

[6] I. R. Chen, J. Guo, and F. Bao, "Trust management for service composition in SOA-based IoT systems," *IEEE Wirel. Commun. Netw. Conf. WCNC*, vol. 4, no. 4, pp. 3444–3449, 2014.

[7] M. P. Papazoglou and W.-J. Van Den Heuvel, "Service oriented architectures : approaches, technologies and research issues," *VLDB J. 16 (2007), no. 3*, pp. 389–41, 2007.

[8] M. H. Valipour, B. Amirzafari, K. N. Maleki, and N. Daneshpour, "A brief survey of software architecture concepts and service oriented architecture," *Comput. Sci. Inf. Technol. 2009. ICCSIT 2009. 2nd IEEE Int. Conf. on, IEEE, 2009*, pp. 34–38, 2009.

[9] A. Karmouch, G. Roussel, O. Festor, C. Nicaud, J. Bonnin, and Y. Ghamri-doudane, "Architecture and application protocols for services choreography in the Internet of Things," 2013.

[10] R. T. Fielding and R. N. Taylor, "Principled design of the modern web architecture," *ACM Trans. Internet Technol*, vol. 2, pp. 115–150, 2002.

[11] M. zur Muehlen, J. V. Nickerson, and K.D, "Swenson, Developing web services choreography standards-the case of rest vs. soap," *Decis. Support Syst. 40*, vol. 1, pp. 9–29, 2005.

[12] K. Li and L. Jiang, "The research of web services composition based on context in Internet of Things," *2012 IEEE Int. Conf. Comput. Sci. Autom. Eng.*, vol. 1, pp. 160–163, 2012.

[13] A. Jula, E. Sundararajan, and Z. Othman, "Cloud computing service composition: A systematic literature review," *Expert Syst. Appl.*, vol. 41, no. 8, pp. 3809–3824, 2014.

[14] Y. BENAZZOUZ, "Context discovery for the automatic adaptation of services in ambient intelligence," 2011.

[15] L. Richardson and S. Ruby, "RESTful web services," *O'Reilly Media, Inc*, 2008.

[16] W. The Petri Nets, "The Petri Nets World." [Online]. Available: http://www.informatik.uni-hamburg.de/TGI/PetriNets/index.php. [Accessed: 05-May-2017].

[17] L. Li and W. Chou, "Design and Describe REST API without Violating REST: A Petri Net Based Approach," *ICWS*, pp. 508–515, 2011.

[18] Y. Du, X. Li, and P. Xiong, "A Petri Net Approach to Mediation-Aided Composition of Web Services," *InIEEE Trans. Autom. Sci. Eng.*, vol. 9, no 2, pp. 429–435, 2012.

[19] J. Xu and W. Wu, "Multiple Attribute Decision Making Theory and Methods," *Tsinghua Univ. Press*, pp. 37–116, 2006.

[20] R. Yang, B. Li, and C. Cheng, "Adaptable service composition for intelligent logistics: A middleware approach," *Proc. - 2014 Int. Conf. Cloud Comput. Big Data, CCBD 2014*, pp. 75–82, 2015.

[21] et al D. Guinard, "Interacting with the SOA-Based Internet of Things: Discovery, Query, Selection, and On-Demand Provisioning of Web Services," *IEEE Trans. Serv. Comput. vol. 3, no. 3*, vol. 3, no. 3, pp. 223–235, 2010.

[22] L. Li, Z. Jin, G. Li, L. Zheng, and Q. Wei, "Modeling and analyzing the reliability and cost of service composition in the IoT: A probabilistic approach," *Proc. - 2012 IEEE 19th Int. Conf. Web Serv. ICWS 2012*, pp. 584–591, 2012.

[23] a Kouicem, a Chibani, a Tari, Y. Amirat, and Z. Tari, "Dynamic services selection approach for the composition of complex services in the web of objects," *Internet Things (WF-IoT), 2014 IEEE World Forum*, pp. 298–303, 2014.

[24] L. Zhang, S. Yu, X. Ding, and X. Wang, "Research on IOT RESTful web service asynchronous composition based on BPEL," *Proc. - 2014 6th Int. Conf. Intell. Human-Machine Syst. Cybern. IHMSC 2014*, vol. 1, pp. 62–65, 2014.

[25] S. et S. collectivement BPEL est développé par IBM, BEA, Microsoft, "BPEL." .

[26] L. Li, T. Tang, and W. Chou, "An XML Based Monadic Framework for REST Service Compositions," *2015 IEEE Int. Conf. Web Serv.*, pp. 487–494, 2015.

[27] H. Abelson, G. J. Sussman, and J. Sussman, "Structure and Interpretation of Computer Programs," *Second Ed. MIT Press*, 1996.

[28] L. Bossi, S. Braghin, and A. Trombetta, "Multidimensional reputation network for service composition in the internet of things," *Proc. - 2014 IEEE Int. Conf. Serv. Comput. SCC 2014*, pp. 685–692, 2014.

[29] N. Hesami Rostami, E. Kheirkhah, and M. Jalali, "Web Services Composition Methods and Techniques: A Review," *Int. J. Comput. Sci. Eng. Inf. Technol.*, vol. 3, no. 6, pp. 15–29, 2013.

[30] S. G. H. Tabatabaei and S. Ibrahim, "A Review of Web Service Composition Approaches," *66*, vol. 6, no. 0, 2008.

[31] K. N. Shah, S. Santoki, H. Ghetia, and K. Ra, "A Review on Web Service Composition Techniques," *Int. J. Eng. Res. Appl.*, vol. 3, no. 3, pp. 929–934, 2013.

[32] S. Pakari, E. Kheirkhah, and M. Jalali, "Web Service Discovery Methods and Techniques : a Review," *Int. J. Comput. Sci. Eng. Inf. Technol.*, vol. 4, no. 1, pp. 1–14, 2014.

[33] K. N. Shah, S. Santoki, H. Ghetia, and K. Ra, "A Review on Web Service Selection for Web Service Composition," *Int. J. Eng. Res. Appl.*, vol. 3, no. 3, pp. 929–934, 2013.

[34] M. Abbas, E. Sabeil, and A. Abdul Manaf, *Comparative Evaluation of Semantic Web Service Composition Approaches*, vol. 180, no. PART 2. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[35] A. L. Lemos, F. Daniel, and B. Benatallah, "Web Service Composition: A Survey of Techniques and Tools," *ACM Comput. Surv.*, vol. 48, no. 3, pp. 1–41, 2015.