Challenges of Service Composition in IoT

Subarna Dhar[1] Sushmita[2] Ashish Kumar[3] Tushar Gupta[4] Prof. Swati Nikam[5]
[1,2,3,4]Student [5]Professor
[1,2,3,4,5]DIT, Pimpri, India

Abstract— The Internet of Things refers to the ever-growing network of physical objects that connect to each and every daily life objects to internet, and the communication that occurs between these objects and other Internet-enabled devices and systems. Service composition is defined such that it describes all steps of the proposed composition process like: requirements definition, requirements decomposition or aggregation using domain ontology, composite service structure construction, service discovery, and structure and service plan optimization. The behavioural service composition problem arises when no available service can achieve target behaviour. This paper discusses about different existing methodologies in which Service Composition is implemented. Service composition is emerging as a valuable approach towards the automatic synthesis of many applications. In addition, various challenges in the latter are focused.

Key words: Service composition, Internet of Things, Service Oriented Architecture

I. Introduction

Internet of Things represents a general concept for the ability of network devices to sense and collect data from the world around us, and then share that data across the Internet where it can be processed and utilized for various interesting purposes.

IOT impacts every business. Mobile and the Internet of Things will change the types of devices that connect into a company's systems. These newly connected devices will produce new types of data. The Internet of Things will help a business gain efficiency, harness intelligence from a wide range of equipment, improve operations and increase customer satisfaction. IOT will also have a profound impact on people's lives. It will improve public safety, transportation and healthcare with better information and faster communications of this information.

In real world applications of IOT, a single module is responsible for performing variety of services or tasks. Combining these services is an important aspect when considering an IOT device. Thus, Service Composition is responsible for achieving this goal.

Service Composition is a type of process to combine multiple services from different types and combining them for a purpose. To qualify as a composition, at least two participating and existing services plus one composition initiator need to be present. Otherwise, the service interaction only represents a point-to-point exchange. Service compositions can be classified into two types, primitive and complex variations of services. In early service-oriented solutions, simple logic was generally implemented via point-to-point exchanges or primitive compositions. As the surrounding technology developed, complex compositions became more common and was widely used. Much of the service-orientation design paradigm revolves around preparing services for effective participation in numerous complex compositions. So much so that the Service binding principle is dedicated to ensure that existing services should be repeated from that new service should be composed.

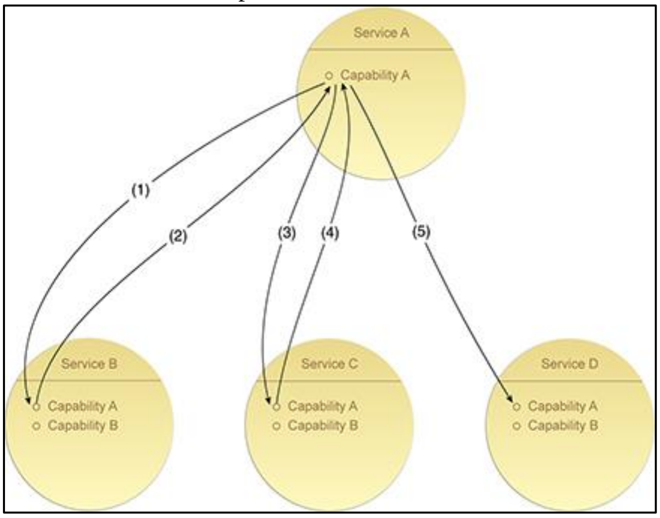

Fig. 1: Service Composition

II. Examples of Service Composition:

A. Smart room

The concept of IOT is used in rooms to make them smart enough to sense the environment and make appropriate decisions to achieve user oriented goals. Service composition is achieved in smart room environment by combining various services like automatic switching of lights on and off whenever the user enters the room, regulation of room temperature according to the environment etc.

B. Smart irrigation

Smart irrigation is a concept in IOT to ease the activities performed in agriculture sector by humans.

Activities include smart sensing of the moisture content in the soil and based on that irrigating the fields from time to time. Also regulation of temperature accordingly to get the best quality crops after harvesting.

C. Smart transportation

IOT plays a vital role in improving the transportation system. Smart traffic signal management which helps in managing the traffic in a very efficient way by switching the traffic lights according to the number of vehicles present on the signal.

D. Smart watch

Smart watches upgrades the traditional watches by providing additional features like calling, monitoring heartbeat, tracking of location, camera and voice recognition etc. Combining all these services into a single watch is the best example of service composition in IOT.

III. RELATED WORK

Matthias Thoma [1] introduced a conceptual view on IOT-services, which allows easy integration of IOT entities into the SOA world and into service science. A definition of IOT services and its classification based on the relationship to a physical entity and their lifecycle. They aimed to merge ideas from the Internet of Services (IOS) and the enterprise IT world for describing and provisioning IOT-services. They propose a simple but comprehensive classification of services along two dimensions: Relationship with the Entity and based on the life-cycle. One of the main challenges in the integration of IOT and Enterprise IT systems is that, in the enterprise world many stakeholders with different roles, different skillsets and different backgrounds participate in creating a complex system.

In [2] Yang Zhang establish a distributed event-based IOT service platform to support IOT service creation and allow for the hiding of service access complexity. They proposed a platform to support the scalable creation of event-driven IOT services based on IOT resource models, called the EDSOA platform. Further, they designed a service-oriented publish/subscribe middleware, based on which established an event-driven service bus to connect together wide-area heterogeneous IOT resources. They implemented EDSOA platform, and have deployed on it a coal mine monitor-control system (CMCS) application which is scalable and reliable.

The formal situation event pattern with event selection and consumption strategy is defined by Bo Cheng in [3]. An automaton-based situational event detection algorithm is proposed. Also, an Enhanced Event-Condition-Action (E-ECA) is used to coordinate the IOT services effectively. The event driven SOA-based services coordination approach is more suitable for IOT application scenarios that are highly delay sensitive and demand a strict real-time response. The real-time data distribution service for largescale IOT application is not optimised. There is not any uniform resource access framework for different physical sensory devices. It does not provide an easy and open API for users.

Pablo Rodriguez-Mier [4] computed a web service dependency graph. A reduction on no. of services was performed by eliminating unused services and combining equivalent services. Lastly, A* search was applied over the reduced graph which finds an optimal service composition with minimal no. of services and execution path. The construction of a non-redundant service dependency graph at the first stage by removing unused services and combining the equivalent ones. Other approaches use simple filtering techniques that do not remove all data redundancy. The use of the A* algorithm backwards, handling multiple services in each step in order to maximize the execution in parallel of the web services. The use of dynamic optimization during the search, reduces the number of possible paths to explore by combining equivalent combination of services.

In [5], the algorithm chooses the best service according to the quality of service. We know that the quality of service is judged by its some non-functional factors, including:

– Running cost
– Runtime
– Success ratio
– Usability
– Trustworthiness
– Degree of security
– Degree of semantic correlation.

According to weight of each factory, we will get the best service. The new algorithm can not only ensure the service quality but also reduce service composite time. Therefore, the new algorithm is more suitable for real and complex situation.

In [6] a new approach to service selection for Service Composition based on QOS and under the user's constraints. So in this approach, the QOS measures are considered based on the user's constraints and priorities. Gravitational Search Algorithm (GSA) is a new optimization algorithm based on the law of gravity proposed. Gravitational and inertia masses are simply calculated by the fitness evaluation. A heavier mass means a more efficient agent. This means that better agents have higher attractions and walk more slowly. Assuming the equality of the gravitational and inertia mass, the values of masses is calculated using the map of fitness. Optimization algorithm has many merits, for example rapid convergence speed, less memory use, considering a lot of special parameters such as the distance between solutions, etc. This algorithm is very much similar to the PSO algorithm (birds population), comparing the result and efficiency of these two algorithms for solving web service composition.

Hui Zhang in [7] proposed an algorithm which eliminates the services with low QOS firstly and then reduces the problem of services composition plan selection for global QOS guarantee to multi-dimension multi-choice 0-1 knapsack problem which is solved by the heuristic method for improving the efficiency of composition. The local optimization approach performs optimal service selection for each individual task in a composite service without considering QOS constraints spanning multiple tasks and without necessarily leading to optimal overall QOS. The global planning approach on the other hand considers QOS constraints and preferences assigned to a composite service as a whole rather than to individual tasks, and compute optimal plans for composite service executions.

Jiuyun Xu in [8], Heuristic immune algorithm with an efficient encoding and mutation method. The algorithm involves two steps: An immune selection operation, which is maintaining antibody population diversity and a clonal selection. The use of a vaccine during the evolution provides heuristic information that accelerates the convergence. The use of heuristic information speeds up the convergence of evolution, especially when the initial population of antibodies is not good.

In [9], Zhenqiu Huang proposed a forward filtering algorithm to find out the optimal composition results efficiently they use backward search (It adopts a forward filtering algorithm to reduce the number of service candidates, along with a modified dynamic programming approach to compute optimal values). An efficient pruning algorithm is proposed and implemented in this paper, which execute a forward pruning before the backward search. In the forward procedure, a filtering algorithm is utilized to reduce the search space and the modified dynamic programming is adopted to compute the optimal QOS values in the algorithm, there are two types of pruning: the first one is removing the services that cannot be enabled and the second type is removing the active services that can provide some concepts but not with the optimal value. The forward-backward algorithm, are more expensive in terms of memory and compute time. Doing a search with a hidden Markov model is about 10 times slower than using a simple Markov model--for larger HMMs (needed for longer target sequences).

Jian Zhou Feng proposed a fuzzy multi objective genetic algorithm (FMOGA) based on fuzzy QOS attributes and fuzzy weights to achieve QOS-based cloud service composition in [10]. The fuzzy QOS total goals are calculated based on weighted-sum approach. The fuzzy weights are obtained through fuzzy AHP method (fuzzy systems are universal approximators of nonlinear functions as neural networks, theoretically we can improve their approximation accuracy on training data to an arbitrarily specified level by increasing their complexity). Theoretical analysis such as statistical learning theory seems to be required. The need efficient tricks for the handling of large data sets by evolutionary algorithms (e.g., stratification. Parallel algorithms seem to be a promising research direction.

A GA–based approach for QOS–aware service composition by Gerardo Canfora in [11]. They determine a set of concrete services to be bound to abstract services contained in an orchestration to meet a set of constraints and to optimize a fitness criterion on QOS attributes. The proposed approach is not for some large–scale service–oriented system. Multi–objective fitness functions will also be considered as an alternative to single–objective fitness functions where factors are aggregated using a weighted sum.

Ming Zhu in [12] discusses the problems of service composition in overlay network environment. RTLB algorithm adopts the service load and node load as the decision making measures and forwards requests to suitable nodes. However, it is complicated in terms of implementation. Prediction algorithm requires existing data and has long processing time.

Yu Chau in [13] proposed a service composition algorithm as a mechanism to provide transparent support to user tasks in service oriented computing environments. The graph based representation of services are combined together into an aggregation that maintains the collective information about all the services available. It provide transparent support in a number of application areas. The techniques to incorporate support for nonlinear composition is not proposed.

Wolf-Tilo Balke in [14] propose the E2Mon algorithm that monitors the execution chain of Web services and gracefully recovers from failures of individual services and network-specific or device-specific alarm. They presented E2Mon, a novel monitoring scheme for the execution of Web service compositions that dynamically adapts to the execution environment. It dynamically responds to a wide variety of events, including service failures, network changes, the discovery of new service implementations, and device-specific alarms, like low battery warnings.

In [15] an on-the-fly algorithm was proposed. Service can achieve a target behaviour in less time. Avoids the full computation on the product of services. Find a solution by visiting only the pertinent (relevant) portion of the state space. On the-fly behaviour can be paired with a heuristic to speed up the synthesis. Visits states as needed, which allows it to deal efficiently with systems containing a large number of complex services. It is self-contained and can be easily incorporated in any other model. Finding out the suitable relevant solution for a particular part of system is not easy. Once the part of the services are selected to do the service composition it might be the case that the algorithm does not work properly in that part. While its worst-case complexity is also exponential in the number of services (this is a lower bound) we argue that in the average case it is much better.

A web service composition algorithm based on the planning graph model was proposed in [16]. Many automatic Web service composition algorithms based on AI planning techniques have been proposed. In this paper, author present an efficient syntactic Web service composition algorithm based on a simplified planning graph. The planning graph, another AI planning technique, provides a unique search space. Author put their efforts into removing the redundant Web services contained in the planning graph. Author approach can find a solution in polynomial time, but with possible redundant Web services. It finds a solution in polynomial time.

Author's approach provides an efficient solution to the Web service composition problem. It is not an effective strategy and solution is not economical. It takes more time to find a solution when we first run Planning Graph Service Composition Algorithm.

Author's design a Cross-modified Artificial Bee Colony Algorithm (CMABC) for service composition in networks, which is an ABC algorithm based method with the assistance of a cross strategy in [17]. In this paper, based on artificial bee colony algorithm (ABC), author pay attention to service selection and propose a cross-modified ABC algorithm (CMABC) to solve the optimization of IOT service instantiation. Aiming at provide user a satisfied service task as quickly as possible, author build a service model and use CMABC to accomplish its instantiation. They proposed to achieve the optimal solution services in an acceptable time and high accuracy. They Exhibited faster convergence speed and better convergence accuracy than some other intelligent optimization algorithms. CMABC has the most beneficial to web service composition instantiation. CMABC is better than some improved intelligent algorithms at accuracy, stability, convergence rate, and time consumption. The algorithm is not perfect. The sequential optimization of task nodes is not considered.

In [18] SOA layer is built on top of the network layer to manage the data and information. The current architecture of IOT does not offer consensus decision making. A cluster based approach is used to calculate the consensus locally which is then combined to reach a global consensus. K-means clustering algorithm has been selected for this particular implantation.

Steps for implementation:

1) Identifying services for implementation.
2) Creating a service composition.

3) Clustering of services.
4) Cluster controlled decision making.

SOA based architecture along with decision making is an effective methodology to deploy IOT solutions when information availability is either inadequate or loaded at the various IOT edge nodes. It provides a consensus for efficient decision making and consistent means of combining information. The Efficiency of clustering algorithm for larger sets of data is a matter of concern.

IV. Research Challenges

The above survey take into account different characteristics of the services providing more sophisticated service composition using different representation mechanisms and degree of automatic composition. The goal of the service composition is combining the services in such a way that they can interact successfully with each other and result in providing a complete and working composite service. During composition both functional and non-functional service characteristics should be considered in order to provide a complete solution.

There has been a significant research on service composition and its approach based on QoS. The concept of QoS is widely applicable in service composition. The work for this approach focuses on the performance of tasks and services. The criteria for ranking are non-functional based on QoS metrics.

Artificial Intelligence methodologies are also suitable for modelling the inter-workings of complex services. Although the ideas expressed within the various Artificial Intelligence methodologies are difficult to apply in real world environments. They do not have a way of expressing important metadata related to the service components.

Semantic methodologies examine the service compatibility and consider both functional and non-functional requirements and allow reasoning on which component is best to use in each situation. They can easily find alternates to existing services and replace them if needed. They are designed to allow description of services in a machine readable way which is difficult to be used by humans.

The various middleware solutions aim to combine the above paradigms. They use AI methods to model the composed service flow while they take advantage of semantic description to allow interoperability between services providing human understandable forms of the composition of services that allow the developer to modify his solutions in code level. They do not consider the requirement for building a service model prior to the actual service development in software.

All the above approaches ignore the business side of providing composite services. In practice, operators will have to set up and manage a variety of business agreements, sometime quite complex, that will enable them to access and exploit component services offered by 3rd parties. Due to which, automation in service composition is a much more complicated issue than the ability to provide a service composition algorithm with a satisfying performance.

V. Conclusion

We have come across different challenges in service oriented architecture of IOT and studied different algorithms related to the service orientation in IOT.

References

[1] Matthias Thoma, Sonja Meyer, Klaus Sperner, Stefan Meissner, Torsten Braun, “On IoT-services: Survey, Classification and Enterprise Integration“, IEEE International Conference on Green Computing and Communications, Conference on Internet of Things, 2012, pp. 257-260.
[2] Yang Zhang, Jun-Liang Chen and Bo Cheng,”Integrating Events into SOA for IoT Services”, IEEE Communications Magazine, September 2017, pp. 180-186.
[3] Bo Cheng, Ming Wang, Shuai Zhao, Zhongyi Zhai, Da Zhu, and Junliang Chen, “Situation-Aware Dynamic Service Coordination in an IoT Environment”, IEEE/ACM TRANSACTIONS ON NETWORKING, 2017, pp. 1-14.
[4] Pablo Rodriguez-Mier, Manuel Mucientes, Manuel Lama, “Automatic web service composition with a heuristic-based search algorithm”, IEEE International Conference on Web Services, 2011, pp. 81-88.
[5] Hai Van, Wang Zhijian, Lu Guiming, “A Novel Semantic Web Service Composition Algorithm Based on QoS Ontology”, International Conference on Computer and Communication Technologies in Agriculture Engineering, 2010, pp. 166-168.
[6] B. Zibanezhad, K. Zamanifar, N. Nematbakhsh, F. Mardukhi, “An Approach for Web Services Composition Based on QoS and Gravitational Search Algorithm”, IEEE, 2009, pp. 340-344.
[7] Wei-hua AI, Yun-xian Huang, Hui Zhang, Ning Zhou, “Web Services Composition and Optimizing Algorithm Based on QoS”, IEEE, 2008, pp. 1-4.
[8] Jiuyun Xu, Stephan Reiff-Marganiec, “Towards Heuristic Web Services Composition Using Immune Algorithm”, IEEE International Conference on Web Services, 2008, pp. 238-245.
[9] Zhenqiu Huang, Wei Jiang, Songlin Hu, Zhiyong Liu, “Effective Pruning Algorithm for QoS-Aware Service Composition”, IEEE Conference on Commerce and Enterprise Computing, 2009, pp. 519-522.
[10] Jianzhou Feng, Lingfu Kong, ”A Fuzzy Multi-Objective Genetic Algorithm for QoS-based Cloud Service Composition”, 11th International Conference on Semantics, Knowledge and Grids, 2005, pp. 202-206.
[11] Gerardo Canfora, Massimiliano Di Penta, Raffaele Esposito, Maria Luisa Villani, “An Approach for QoS-aware Service Composition based on Genetic Algorithms”, ACM, June 2005, pp. 1-7.
[12] Cairong Yan, Ming Zhu, Youqun Shi, “A Response Time based Load Balancing Algorithm for Service Composition”, IEEE, 2008, pp. 13-16.
[13] Yu chao, Yuan Meng-ting, “Service Composition Algorithm in SOC”, IEEE, 2010, pp. 1-4.
[14] Wolf-Tilo Balke and Jorg Diederich, “A Quality- and Cost-based Selection Model for Multimedia Service

Composition in Mobile Environments", IEEE, 2004, pp. 1-8.
[15] Hikmat Farhat, Guillaume Feuillade, "On-the-Fly Algorithm for the Service Composition Problem", IEEE, 2015, pp. 1-6.
[16] Xianrong Zheng, Yuhong Yan, "An Efficient Syntactic Web Service Composition Algorithm Based on the Planning Graph Model", IEEE International Conference on Web Services, 2008, pp. 691-699.
[17] Lei Huo, Zhiliang Wang, "Service Composition Instantiation Based on Cross-Modified Artificial Bee Colony Algorithm", China Communications, October 2016, pp. 233-244.
[18] Sandhya Balasubramaniam, R. Jagannath, "A Service Oriented IoT Using Cluster Controlled Decision Making", IEEE International Advance Computing Conference, 2015, pp. 558-563.