Journal of Engineering Sciences
Vol11,Issue2,FEB/2020
ISSN NO:0377-9254

A COMPOSITE SERVICE COMPOSITION IN MOBILE AD HOC NETWORKS BASED ON NOVEL NODE MODEL

V. CHANDRASEKHAR[1], Dr. P. VEERESH[2]

1. Asst. Professor, Department of Computer Science and Engineering, Samskruti College Of Engineering & Technology, Ghatkesar, Hyderabad, chandrasekhar_bly@yahoo.co.in

2. Professor, Department of Computer Science and Engineering, SJCET, YEMMIGANUR

Abstract: Composite service provision in mobile ad hoc networks encounters great challenges and its success rate is not satisfactory because the nodes' locations are dynamic and the nodes maybe unavailable at any time. Composite service is built through the service composition. In mobile ad hoc networks, the factors influencing the success rate of service composition are mainly the number of nodes and the time spent for the composition. The node's failure probability is proportional to the length of time the node exist in the networks. In order to improve the success rate of service composition, we take several measures. First, we split the service requirement into several segments and cluster the nodes, so that the nodes' waiting time for service composition can be reduced. Second, we propose a new node model of "one node contains multiple services" in mobile ad hoc networks. Using this type of nodes model, the number of nodes required for service composition can be reduced. These means can increase the success rate of service composition.

Key words: service provision; service composition; mobile ad hoc networks; cluster; requirement splitting

I. INTRODUCTION

Currently, the mobile devices, e.g., smart-phones and PDAs are becoming ever-increasingly powerful, and bring us more and more convenience. With the growing popularity of these powerful smart mobile devices, their connection with each other are increasingly important. Mobile devices can constitute a Mobile Ad Hoc Networks (MANET) through blue-tooth or Wi-Fi. MANET is a multi-hop, self-organized wireless network, and it can be composed of mobile devices, such as mobile phones, PDAs. These mobile devices act as nodes. In the MANET, if a node is in the transmission range of other nodes, these nodes can communicate with each other. Services are provided by these nodes, and service composition can be conducted by the cooperation of them. By employing the Service-Oriented Architecture (SOA) technology, the services provided by heterogeneous platforms can be seamlessly integrated or composed.

Different from the wired networks, the topology of mobile ad hoc networks is time-varying [3]. The mobile terminal device (node) is also different from the terminal in the wired networks. Some physical properties of the mobile devices may subject to change, such as speed, direction, and location. Furthermore, nodes may fail at any time, so the services which the nodes provided are unavailable. These features bring great challenges to the services composition in the MANET.

For example, in a subway, mobile devices carried by passengers form a mobile ad hoc network. A passenger proposes a requirement of service composition including 10 services $s0$, $s1$, $s2$, ... $s9$, which exist in the passengers' mobile devices. Here, the *requirement of service composition* means which services are required by a passenger to complete a task.

The requirement is sponsored by the passenger who wants to do the task. Assuming that the services are executed according to their sequence from $s0$, $s1$, $s2$, ... to $s9$, and the services $s0$, $s1$, ... $s6$ have been executed successfully. When the service $s7$ is being executed, the subway stops. Some passengers get off the subway, and the nodes which provide the services $s8$ and $s9$ leave the network. If the substitutes of services $s8$ and $s9$ cannot be found in the network, the service composition may fail.

Specifically, we improve the success rate of service composition by taking the following measures:

Firstly, we split the service requirement into several segments and cluster the nodes, so the nodes' waiting time for service composition can be reduced. Secondly, we propose a new node model of "one node contains multiple services" in mobile ad hoc networks. Using this type of node model, the number of nodes required for service composition can be reduced. By these means, the success rate of service composition can be increased.

Generally, service provision includes service publishing, service discovery, service selection, service composition, and so on. In this paper, service provision mainly refers to service composition, also contains a part of service selection.

II SERVICE COMPOSITION

2.1 A novel node model

In order to reduce the failure rate of service composition, we first propose a new type of node model has several services are contained in a single node. Using this type of node model, the number of nodes involved in a service composition may be reduced. As the decrease in the number of nodes, the success rate of the service composition may increase.

2.2 The Process of service composition

When a requirement of service composition is proposed, it will experience the process of requirement splitting, nodes grouping and nodes clustering, etc. The process is depicted in Figure 1.

The Figure 2 denotes the splitting of requirement and the clustering of nodes. After a requirement is proposed, the long requirement can be divided into multiple service segments according to the thinking of some services that can be executed in parallel. Each segment contains several services. Nodes providing these services constitute a node group, so every service in a segment is contained in a group. Since there are several services which can be executed concurrently in the requirement, the requirement has several splitting methods.

The algorithm 1 describes how to split the requirement of service composition into segments. After the splitting of requirement and the grouping of nodes, the services in a segment of requirement are contained in a node group. That is, a group of nodes complete a service segment. In the group, there may be several nodes providing the same service and there semay be a node providing multiple services. If a service provided by a node is a component of a composite service, i.e., the composite service takes this service as a part of it, we say that the node participates in the composite service. Because each node has the probability of failure, the more nodes participate in the composite service, the greater the failure probability of the composite service will be. In the process of service composition, nodes will communicate with each other. The more nodes participate in the composition, the more the traffic of networks may be. Ensuring a service segment can be executed in a group, reducing the number of nodes in the group is able to improve the success rate of service composition. In a node group, we can select the necessary nodes to form a cluster. The services of a segment contained in the group are also contained in the cluster. The method of selecting nodes from a group to constitute a cluster is described in the second algorithm.

After the forming of the cluster, storing the result of the service segment in this cluster to a special node can improve the success rate of service composition. The special node is the cluster head. Additionally, in a cluster, cluster heads manage each node and are responsible for the allocation of resources within the cluster. Outside the cluster, the cluster heads communicate with each other and collaborate to complete the service composition. The node of cluster head is different from the general nodes, and it need to be chosen. The selection method for the cluster head is described in the third algorithm.

2.3 Requirement splitting and node grouping for service composition

We assume that to complete a task requires m services $s0$, $s1$, ..., $sm-1$, and the serial numbers of the services are continuous. That is to say, the requirement of service composition S contains $s0$, $s1$, ..., $sm-1$, i.e., S is:

$$S=\{s0, s1, s2, ..., sn, [sn+1], sn+2, ..., sl, [sl+1], sl+2,..., sm-1\}$$

"[*]" means the service can be executed concurrently with other services, then the segments of S can be expressed as Figure 3:

In Figure 3, S is divided into K segments, and the first segment consists of $s0$, $s1$, ..., sn. Accordingly, in the networks, the starting node containing the starting service $s0$ broadcasts its location in the networks firstly. This node looks for the node containing the successor service $s1$ according to the breadth-first strategy, and the node containing the service $s1$ looks for the successor service $s2$, and so on. The successor services' number must be sequentially continuous. If all the services contained in a segment can be found in the networks, the nodes providing these services constitute a node group. The nodes containing services *s*0 to *sn* compose a group. If *n*<*m*−1, the node in this group which contains the last service *sn* informs the node which contains service *sn*+1. Service *sn*+1 will be the next starting service in the second segment of the composite service. The second starting node containing the service *sn*+1 will look for successor service *sn*+2. Assuming the max numbered service in the second segment is *su*, if *u*<*m*−1, the nodes repeat the above steps, until they have found the last service *sm*-1 in the networks. Similarly, services from *sn*+1 to *su* belong to the second segment, and nodes containing these services constitute the second node group.

Algorithm: Requirement splitting and node grouping for service composition

//*k*: the number of node group

//*Tk*: the set of nodes which contains services

set Tk = *NULL k* = 0

For (i=1; i<=m-1; i++)

{

If service *si* is not the starting service of a segment

{

To seek service *si* in *Tk*

If *si* is in *Tk*

Continue

Else

{

Seek in the networks

Add the node containing *si* to

}

}

Else

{

k + + //staring a new node group

Add the node containing service to *Tk* Continue

}

}

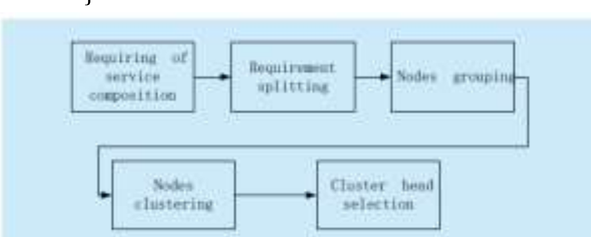

Fig.1 The Process of Service composition

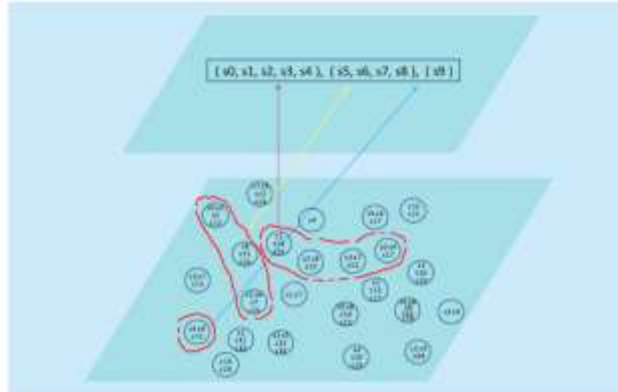

Fig.2 The requirement splitting for service composition and clustering of the nodes.

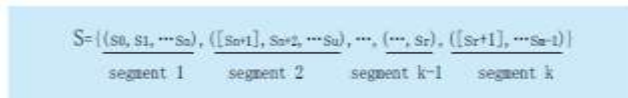

Fig.3 Requirement splitting

2.4 The algorithm to select the cluster members from the group

Consecutively numbered *n* services *s*1, *s*2, ..., *sn* form a segment of a composite service *S*. Let *S*1 {*s*1, *s*2, ..., *sn*}, *S*={*S*1, *S*2, ..., *SW*}. These *n* services *s*1, *s*2, ..., *sn* are contained in *L* nodes *N*1, *N*2, ..., *NL*. *L* is not necessarily equal to *n*. Nodes *N*1, *N*2, ..., *NL* constitute a group *C*1 . Let *T*1 {*N*1, *N*2, ..., *NL*}. In *C*1, there may be several nodes providing the same service. For example, as shown in Figure 4, the nodes *NC*, *ND*, *NF* can all provide the service *s*3. Our objective is to find out the optimal nodes and select these nodes to constitute a cluster.

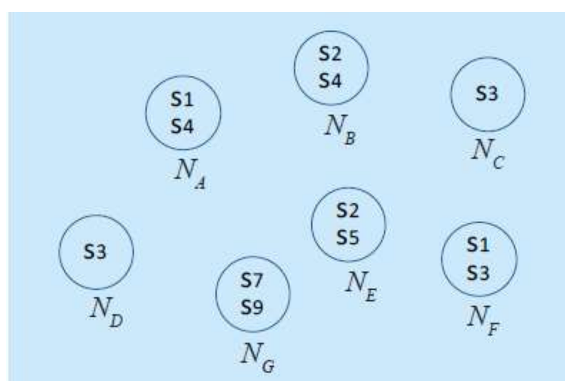

Fig.4 *An example*

The algorithm to select appropriate nodes in a group to constitute a cluster is described as follows:

(1) Let *R*1=*S*1. Here *S*1 is the set of the services contained in a service segment. We can identify the correspondence relationship between

nodes and services by two Tables. Listing out the corresponding relationship from services to nodes, we can see which nodes can provide the service. Additionally, listing out the corresponding relationship from nodes to services, we can see which services are provided by a node.

(2) For some services, if they are provided only by one node, this node is selected as a cluster member. Delete these services from $R1$.

(3) If the node in (2) also contains other service(s) which belong to $R1$, then the service(s) must be provided by this node. Delete the service(s) from $R1$.

(4) If $R1 \neq \emptyset$. Do (5) and (6)

(5) According to the order of the service number, we can select an appropriate node which provides the service. From the Tables listed in (1), we can see that if the node contains the largest amount of services which belong to $R1$, then the node is selected as the cluster member. Delete these services form $R1$.

(6) If several nodes have the same amount of useful services, we select the node as the cluster member according to their F value (algorithm 3). Repeat the above steps (5) and (6). Delete these services from $R1$.

(7) If $R1 = \emptyset$, the operation is complete.

Taking Figure 4 as an example, we explain the process of selecting the nodes as the cluster members according to the above algorithm. We assume $S1$ $\{s1, s2, s3, s4, s5\}$.

Step (1) $R1$ $S1$, then $R1$ $\{s1, s2, s3, s4, s5\}$. We list out the relationship between the nodes and the services in the Table I and Table II. From Table I, we can see which nodes can provide the service. Table II denotes which services are provided by a node.

Step (2) Because the service $s5$ is provided only by the node *NE*, then the node *NE* is selected as a cluster member. We delete $S5$ from $R1$, and now $R1$ $\{s1, s2, s3, s4\}$.

Step (3) Node *NE* also contains service $s2$, $s2 \in R1$. Then the service $s2$ which $S1$ required is provided by node *NE*. We delete $s2$ from $R1$, and now $R1$ $\{s1, s3, s4\}$.

Step (4) $R1 \neq \emptyset$. We study the node in Table I and Table II. The service $s1$ is provided by the node *NA* and . The node *NA* contains $s4$, and $s4 \in R1$. The node *NF* contains $s3$, and $s3 \in R1$. Then the node *NA* and *NF* can both be selected as the member of the cluster. For *NA* and *NF*, we can determine which node is more appropriate and select it as the cluster member according to the third algorithm.

Table I *The services and the nodes which provide the services*

$s1$: *NA*, *NF*
$s2$: *NB*, *NE*
$s3$: *NC*, *ND*, *NF*
$s4$: *NA*, *NB*
$s5$: *NE*
$s7$: *NG*
$s9$: *NG*

Table II *The nodes and the services which a node can provide*

NA : $s1$, $s4$
NB : $s2$, $s4$
NC : $s3$
ND : $s3$
NE : $s2$, $s5$
NF : $s1$, $s3$
NG : $s7$, $s9$

Step (5) In the step (4), if *NA* is selected as the cluster member, then $R1=\{s3\}$. Now, we can select the node *NC* or *ND* as the cluster member because both of them can provide $s3$, then $R1 = \emptyset$. We must use the third algorithm to select the cluster member from *NC* and *ND*. The operation of selecting the nodes as the members of the cluster is over. The sequence for selecting the node to the cluster is *NE*→*NA*→*NC*(or *ND*).

The composition sequence of the service S 1 i s *NA*(s 1) →*NE*(s 2) →*NC*(o r *ND*)($s3$)→*NA*($s4$)→*NE*($s5$). Step (6) In the step (4), if we select the *NF* as the cluster member, then $R1=\{s4\}$. Now, we can select the node *NA* or *NB* as the cluster member because both of them can provide $s4$, then $R1 = \emptyset$. Similarly, we must use the third algorithm to select the cluster member from *NA* and *NB* . The operation of selecting the nodes as the member of the cluster is over. The sequence for adding the node to the cluster is *NE*→*NF*→*NA*(or *NB*).

The composition sequence of the service $S1$ is *NF*($s1$)→*NE*($s2$)→*NF*($s3$)→*NA*(or *NB*)($s4$)→*NE*($s5$).

2.5 The algorithm to calculate the *F* value of a node A cluster executes the services contained in a segment of the composite services. The cluster head is responsible for managing the nodes and saving the results of the service composition segment in the cluster.

After executing all the services of the segment in the cluster, the cluster head exchanges information with other cluster heads. The results of the execution of the services in the cluster are saved by the cluster head. In the cluster, other member nodes' failure has nothing to do with the result after these nodes completing the services they provide. This method can greatly improve the success rate of the service composition. The algorithm to calculate the node's *F* value is described in following steps. According to the *F* value, we can select the cluster head.

Step 1. Calculate the connectivity *CNi* between the nodes The nodes in a cluster are connected to each other. But the connection strength between nodes are not the same, which can be represented by the correlation matrix. The value of the connection strength between two nodes depends on the signal strength of them. Signal strength can be divided into five grades. The values of the connectivity strength between two nodes are equal, i.e. $Rij = Rji$. For example, (2) Here *CNi* denotes the connectivity strength between node *i* and other nodes in the cluster.

Step 2. Calculate the processing ability *DNi* of a node Processing ability of a node represents the time that the node required to complete a service. In the mobile ad hoc networks the node may be unavailable at any time, therefore determining the time a node needed to execute a service is very important. The processing ability of a node depends on its hardware conditions. We classify the processing ability of a node into four levels (refer to Table III). For example, if the processing ability of a node is rank 2, then we set the $DNi = 0.75$.

Step 3. Calculate the number *ENi* of the useful services in a node For a node, we can calculate its useful services by comparing the services it provided with the services contained in the composite service.

Step 4. Calculate the node's *F* value After getting the result from the above steps, they also require further processing, such as, the normalization, and to get the new *CNi*, *DNi* and *ENi*. The node's *F* value can be calculated by multiplying these results and appropriate values of weight which are corresponding to each result. In the cluster, the node which has the highest value will be selected as the cluster head.

III CONCLUSIONS

This paper presents a novel model of node which contains multiple services. Based on the premise that some service can be executed at the same time, we propose a requirement splitting method for service composition and a node clustering method. In this paper, the node's moving speed is 0~1.5m/s. From these figures, we can see that our methods can increase the success rate of the composite service. In the future, we will increase

IV REFERENCES

[1] ZHANG Qing, CHEN Huiqiong, YU Yijun, *et al*, Dynamically Self-Organized Service Composition in Wireless Ad Hoc Networks[J]. Technologies for E-services, 2005: 95-106.

[2] JIANG Shanshan, XUE Yuan, DOUGLAS C. SCHMIDT. Minimum Disruption Service Composition and Recovery in Mobile Ad Hoc Networks[J], Computer Networks 2009: 1649–1665.

[3] R. RAMANATHAN, J. REDI. A Brief Overview of Ad Hoc Networks: Challenges and Directions[J]. IEEE Communications Magazine, 2002, 40 (5).

[4] SUN Qibo WANG Wenbin, ZOU Hua, *et al*, A Service Selection Approach Based on Availability-Aware in Mobile Ad Hoc Networks[J], China Communications, 2011,(1): 87-94.

[5] YEON-SEOK KIM, YOO-SEOK SHIM, KYONG-HO LEE. A Cluster-Based Web Service Discovery in MANET Environments[J], Mobile Information Systems, 2011: 299-315.

[6] WANG Jianping. Exploiting Mobility Prediction for Dependable Service Composition in Wireless Mobile Ad Hoc Networks[J], IEEE Transactions on Service Computing, 2011, 4(1): 44-55.

[7] YOO-SEOK SHIM, YEON-SEOK KIM, KYONG-HO LEE. A Mobility-based Clustering and Discovery of Web Services in Mobile Ad-hoc Networks[C], Proceedings of ICWS'09, 2009: 374-380.

[8] KUN YANG, ALEX GALIS, HSIAO-HWA CHEN. QoS-Aware Service Selection Algorithms for PervasiveService Composition in Mobile Wireless Environments[J], Mobile Networks Appl, 2010, 15: 488-501

[9] I. PAPAPANAGIOTOU, D. TOUMPAKARIS, J. LEE, *et al*. A Survey on Next Generation Mobile Wi-MAX Networks: Objectives, Features and Technical Challenges[J], IEEE Commun. Surveys Tuts, 2009, 11(4): 3-18.