

Web Service Composition Framework using Petrinet and Web Service Data Cache in MANET

M. Deivamani, S.R. Murugaiyan, V. Ravisankar, P. Victor Paul,
R. Baskaran, P. Dhavachelvan

M. Deivamani*, R. Baskaran

Department of Computer Science and Engineering,
Anna University, Chennai, India.
m.deivamani@gmail.com, baaski@annauniv.edu
*Corresponding author:m.deivamani@gmail.com

S.R. Murugaiyan

Department of Computer Science and Engineering,
M.S.University, Tamil Nadu, India.
murugaiyansr@gmail.com

V. Ravisankar

Department of Computer Science,
Bharathiar University, Tamilnadu, India.
v_ravisankar@yahoo.com

P. Victor Paul

Department of Information Technology,
Sri Manakula Vinayagar Engineering College, Puducherry, India.
victorpaul@gmail.com

P. Dhavachelvan

Department of Computer Science,
Pondicherry University, Puducherry, India.
dhavachelvan@gmail.com

Abstract: A Mobile Ad Hoc Network (MANET) is characterized by multi-hop wireless links and frequent node mobility. Every neighboring node in the MANET is likely to have similar task and interests, several nodes might need to access the similar web service at different times. So, by caching the repeatedly accessed web service data within MANET, it is possible to reduce the cost of accessing the same service details from the UDDI and also from the external providers. *Composition of web services leads to a better alternative as, at times a candidate web service may not completely serve the need of the customer.* An effective Data Cache Mechanism (DCM) has been proposed in [6] using the Distributed Spanning Tree (DST) as a communication structure in Mobile network to improve scalability and lessen network overload. As an enhancement, Ant Colony Optimization (ACO) technique has been applied on DST to cope with the fragile nature of the MANET and to improve the network fault tolerance [1]. In these perspectives, an efficient Web Service Cache Mechanism (WSCM) can be modeled to improve the performance of the web service operations in MANET. In this paper, a fine grained theoretical model has been formulated to assess the various performance factors such as Cooperative Cache and Mobility Hand-off. In addition to these, the performance improvement of WSCM using DST and ACO optimized DST techniques in MANET has been proved experimentally using Precision and Data Reliability of the system using appropriate simulation.

Keywords: Data Cache, Web Service, Distributed Spanning Tree, Ant Colony Optimization, MANET, OMNeT++

1 Introduction

A Mobile Ad-Hoc Network (MANET) is an autonomous collection of mobile nodes that communicate over relatively bandwidth constrained wireless links. Since the nodes are mobile, the network topology may change rapidly and unpredictably over time. The network is decentralized; where all network activity including discovering the topology and delivering messages must be executed by the nodes itself. i.e., routing functionality will be incorporated into mobile nodes [18]. The nodes in MANET would probably work for tasks of similar goal (common interest). So, most of the nodes would try to access the same web service data at different time through their corresponding Access Point (AP). The Access Points may be located at the boundaries of the MANET, where reaching them could be costly in terms of delay, power consumption, and bandwidth utilization. Moreover, the access points would be connected to a highly overloaded resource (e.g., a satellite), or an external network that is susceptible to intrusion plays a vital role in response time, security and availability of the system. For such reasons, it is recommended to cache the frequently accessed service information within the nodes in the MANET and the search application should check for the availability of the required service data within the network before requesting the external service registry [2].

Caching of service refers to the technique of caching the service method invocation information (WSDL) from the registry or service response from the corresponding service providers. The cached responses of service methods can be utilized only if the future requests use similar arguments as that of cached responses. Caching the WSDL information saves significant time and resources because subsequent service requests of similar methods will not be required to download WSDL files in a repetitive fashion. At this juncture, a service item refers to the cached WSDL information description or/and the cached web response of the corresponding service. So, the MANET applications should check for the existence of the desired service item within the network before attempting to request the external service source [2, 6]. This scenario can reduce the overload of the system for accessing external source for same service and also avoid the possible intrusion threats.

A set of proxy nodes are introduced in the MANET to provide information about the mobile nodes in the network and the services invoked by them. The proxy nodes are configured with a domain ontology and petri net modelling. The domain ontology enhance the selection of appropriate web service (by using semantics), from the service registry or peer agent nodes. The petri net modelling aims to provide composite value added service to the service requester. The service data caching within the MANET can be discussed in two methods based on the diversity of the cached service information. In most of the works, the decision to cache a service is done locally in the proxies [3, 14, 16, 17], that is, without taking into account the all the peers within the network. In such case, there may situation happens that multiple copies of the same data can be cached in the proxy nodes. This redundancy of same service data cache could reduce the possibility to cache different data that are also of interest, which affect the overall performance of the cache system. On other hand, nodes are made to decide the caching data co-operatively among the proxies [13, 15, 18], which can improve the cache diversity and also the overall data cache performance of the system. Various opportunities and challenges, like load balancing and mobility, which arise on caching web data within mobile networks, are theoretically discussed in [12].

Lan Wang [3] proposed Clustering in large-scale MANET as a means of achieving scalability through a hierarchical approach in which every node in the cluster is one hop away from every other node, that is, each cluster is a diameter-1 graph. But static cached data item manager may easily become the traffic bottleneck and single point failure of the cluster [4]. Hassan Artail et al. [2] proposed the Minimum Distance Packet Forwarding technique for search applications

within MANET that are based on the concept of selecting the nearest node from the designated nodes. The cache techniques in the studied works endure problems such as large hop count, message density and single point failure because of not following some efficient communication structure within the MANET. To cope with the problem stated, Distributed Spanning tree (DST) has been used as a communication structure in MANET for effective data cache technique proposed in our previous work [5]. DST is a recent and formally proved communication structure in MANET to lessen node isolation problem, to reduce the number of hops required to reach the nodes and makes the network scalable [7, 8, 10].

Another important performance factor in MANET is finding and maintaining routes since node mobility causes topology change which need to be observed for effective communication. In [1], Ant Colony Optimization has been used to deal with the fragile nature of the MANET which dynamically identifies the optimal path between the nodes in the DST on-demand. It is also justified that applying ACO on DST, enhance the effective routing of message (at low cost) in the MANET which in turn reduces the number of message hops required for communication to achieve excellent efficiency in DCM applications. Though, effective WSCM in MANET using DST and ACO techniques has been formally proposed in [1, 6], experimentation analyses of the work has been performed for very few performance factors such as hit ratio and message passes. Thus, in this paper, it is intended to conduct an extensive analysis on several other critical performance factors such as Cooperative Cache model, Mobility Hand-off, Precision and Data Reliability.

2 Background information needed

In this section, the discussion on innovative techniques proposed in [1, 6] which are necessary to understand the performance assessments performed in the following sections of the paper.

2.1 Distributed Spanning Tree (DST)

Distributed Spanning Tree (DST) [10] is the interconnection formation we follow as in [5, 11, 21] which, improve the routing and reduce the number of message passes required for any communication in MANET. DST systematizes MANET into a hierarchy of groups of nodes. The DST is an overlay structure designed to be scalable [11]. It supports the growth from a small number of nodes to a large one. A comprehensive algorithm for formulation of DST in MANET has been proposed and exemplified in [9]. Consequently, the MANET can be logically converted into DSTs and each DST should have its root node, named as the Head Node (HN) and the possible Leaf Nodes (LNs). Every HN will hold the complete details regarding its LNs and vice versa. These HNs are to be generated dynamically and should hold the service cache details, which is to be accessed by their corresponding LNs and indeed by other HNs also. *In addition to the cache details, domain ontology and Petrinet formalism is included in the HNs to deliver prominent service compositions.*

The DST formulated MANET can be represented as G_m in equation(1),

$$G_m = \left\{ \begin{array}{l} DST_1 = (HN_1, LN_{11}, LN_{12}, \dots, LN_{1(j_1-1)}) \\ DST_2 = (HN_2, LN_{21}, LN_{22}, \dots, LN_{2(j_2-1)}) \\ \cdot \\ \cdot \\ \cdot \\ DST_i = (HN_i, LN_{i1}, LN_{i2}, \dots, LN_{i(j_i-1)}) \end{array} \right\} \quad (1)$$

Where,

- DST_v is the Distributed Spanning Tree and 'i' is the total number of DSTs formed in the network and $0 < v \leq i$
- HN_v is the Head Node (HN) and 'i' is the total number of HNs in the peer network equal to the number of DSTs and $0 < v \leq i$
- LN refers to the Leaf Node(s). In LN_{vz} , refers to the corresponding HN_v and $0 < z \leq j_v - 1$, where ' $j_v - 1$ ' is the total number of LN_s in the corresponding DST.

Ant colony optimization for DST

Ant colony optimization (ACO) [19,21] is one of the most recent techniques for approximate optimization. The inspiring source of ACO algorithms are real ant colonies. More specifically, ACO is inspired by the ants' foraging behavior. By applying the ACO over the formulated DST [1], we can obtain the optimal path in terms of reduced number of message passes among the nodes in the network. ACO is also capable to reform a new optimal path in case of any problem with the current optimal path. In this paper the Ant Colony Optimization Algorithm has been modified and proposed for finding an optimal path in DST of the MANET. By optimizing every DST and connection among all the other DSTs through their HNs, it can be argued that the entire network is optimized with ACO technique for improved efficiency.

The Complexity of ACO technique depends on the method it is implemented in the MANET. In DST structure, computational complexity for ACO technique can be calculated as,

$$O(N_{DST} * N_{LN}) + O((N_{DST})^2) \quad (2)$$

Where,

- N_{DST} is the number of DSTs or the number of HNs formed in the network
- N_{LN} is the number of LN under a HN (theoretically taken same number of LNs under every HN)

2.2 Web Service Data Cache Mechanism (WSCM) with DST and ACO techniques

An efficient WSCM system in MANET with DST and ACO techniques has been formulated in [1,6] with necessary algorithms. The projected system has the capability to cope with fragile and dynamic topology changing MANET environment and the system structure can be viewed as a four layered as shown in Figure 1.

MANET Network Layer - is a network level layer consists of wireless and highly dynamic topology network.

DST Formulation Layer - is simple and converts the graph structure MP2P network into a collection DSTs. This DST structure provides the features that are necessary for a dynamic network like reduced size of routing table, minimizes routing overhead, easy network management, reduced message pass, load balancing and fault tolerance. This layer offers the dynamic node insertion into the network and exit from the network in both normal and abnormal manner. This layer also makes the system highly scalable.

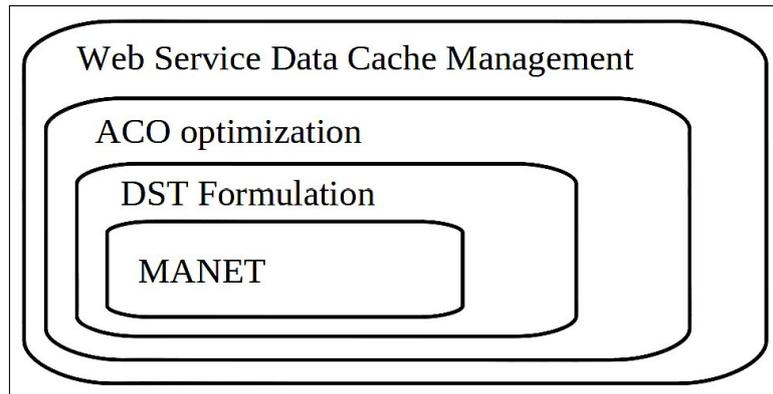


Figure 1: The layered architecture of WSCM in ACO optimized DST MANET

ACO optimization Layer - provides the system to manage with the dynamic nature of the MANET. This layer works in simple, effective and on demand way which makes the system to operate on a fragile system with asymmetric links and constantly changing topology.

Web Service Data Cache Management and Service Composition - is an application level layer with specified protocol for an effective web service information cache management system in MANET. *Using the Petrinet formalism, the web services are composed together to enhance the service quality.* Thus, using this four layer system structure, WSCM application can be efficiently performed in the highly fragile MANET environment using DST and ACO techniques.

3 Experimental analyses

In this section, an extensive analysis of DCM system using DST and ACO techniques in MANET has been performed based on critical performance factors such as Cooperative Cache model, Mobility and Hand-off, Availability, Routing technique, Cache Replacement method, Precision and Data Integrity of the system.

3.1 Simulation Setup

A MANET environment with 30 nodes and 70 service items has been simulated using OM-NeT++ tool, which is an object-oriented modular discrete event network simulator. Table 1 show the partial view of HNs and its neighboring nodes, and hop distance between them. Simulation parameters are followed as similar to [1]. In the simulated network, the HNs formed are *node05, node11, node17, node23 & node29* and other nodes act as LN to any one of these HNs. For our simulated, equation (1) can be rewritten as,

$$\left\{ \begin{array}{l} DST_1 \\ DST_2 \\ DST_3 \\ DST_4 \\ DST_5 \end{array} \right\} = \left\{ \begin{array}{cccccc} 06 & 03 & 02 & 05 & 17 & - & - \\ 12 & 01 & 08 & 14 & 20 & 25 & - \\ 18 & 04 & 07 & 19 & 10 & 22 & 27 \\ 24 & 14 & 13 & 16 & 23 & 28 & - \\ 30 & 29 & 09 & 15 & 21 & 26 & - \end{array} \right\} \quad (3)$$

Where,

- The node number are to be preceded by the term *node*, for example the DST_1 should be interpreted are $DST_1 = (node06, node03, \dots, node17)$, in which the first node *node06* is the HN and all other nodes are LNs of DST_1 .

To simulate the web service cache mechanism, we created 50 different service items and stored in the *service registry* which are accessed by the *proxy nodes* in the MANET through the access point. When a node requires any service, it will send the request to its corresponding HN (proxy node). *The HNs upon receiving the requests, extracts the keywords and match with the cache entries to identify the service has already executed from that node with in a cut-off time. The cut-off time is a time span a service information in a cache will remain valid, after which the WSDL information will be deleted from the cache (leading to cache miss for the next attempt). This deletion ensures to devour the up to date information in the cache. The HNs advertises its presents in the network for the LNs to identify and request for a service.*

S.No	HN	Nearest HNs and distance in Hop(s)			
		HN 1	Hop Distance from HN	HN 2	Hop Distance from HN
1	node06	node18	2	node30	4
2	node12	node06	4	node17	2
3	node18	node30	3	node12	2
4	node24	node17	4	node18	3
5	node30	node12	2	node06	5

Table 1: HNs formulated and its hop distance from the nearest two HNs in the simulated MANET

During the very first time access of any service, the LN saves the copy of the accessed service item and intimates its corresponding HN to save the type of the service item and the details of LN which holds it. When any node request for the web service of similar nature, the requesting node is served with the service item by the LN which holds the cache, identified through its HN. *A keyword extracted from the request may or may not match with the cache entries. If it matches, the WSDL information is provided to the corresponding requesting LN using standard message passing. If the keyword does not match with the cache table, the domain ontology is used at the first level to find related services (rather than exact keyword matching) from the cache. At the next stage, if related services also give a cache miss, the peer HNs or the external service registry is contacted for availing the service configurations.* Upon reaching all the HNs, it can be found that the service does not exist within the network and obtained from the external UDDI/service providers.

A service request initiated by a LN can be fulfilled by an atomic service or a composite service. At times, atomic services may not be available for a given service request, where several compatible services are identified, composed and executed. To identify the compatible services for composing a value added service, petrinet formalism is used in the MNs. Petri Net is used to mainly identify the reachability of the compatible web services within the domain.

DEFINITION 1: The Best and Worst case analysis for accessing the cached service item based on the message hops can be modeled as follows.

Best Case: The total number of message hops required to access the cached data item is minimum, when service item is being cached in any LN which is under the HN of the requesting node.

This can be expressed as,

$$n(access_messagehops) + n(dictionary_search) \leq 4N \quad (4)$$

where,

- $n(access_messagehops)$ is the total number of message hops required to access the cached service item.
- $n(dictionary_search)$ is the total time to search the semantic of the keyword in the search.
- N is the total number of message hops between HN and its LN (consider equal for every LNs).

Worst Case: The total number of message hops required to access the cached service item is maximum, when the HN of the node that holds the requested service item is at distance 'k' from the HN of requester node, where 'k' is the total number of HNs in the MANET. This can be expressed as,

$$n(access_messagehops) + n(dictionary_search) \geq (kXM) + 4N \quad (5)$$

where,

- $n(access_messagehops)$ is the total number of message passes required to access the cached item
- $n(dictionary_search)$ is the total time to search the semantic of the keyword in the search
- M is the total number of message pass between two HNs (consider equal for between every HNs).
- N is the total number of message pass between HN and its LN (consider equal for every LNs).

An extensive analyses to model the various performance factors such as Cooperative Web Service Cache model, Mobility and Hand-off, Availability, Routing technique, Cache Replacement method, Precision and Consistency for the WSCM in DST and ACO optimized DST MANET have been performed in the following sections.

3.2 Cooperative cache model

To improve the service information accessibility, mobile nodes should cache different service item that of their neighbor nodes [1]. Every LN should cooperatively cache the different service to avoid the replicated caching of same service within the network. Caching same service on different LNs may reduce the access delay but on considering size of the cache memory in LNs it will block caching more frequently accessing services (different).

DEFINITION 2: Let LN_i is the mobile node which cache the service item. To show that each LN caches different service,

$$LN_1(S_{id}) \cap LN_2(S_{id}) \cap LN_3(S_{id}) \dots \cap LN_k(S_{id}) = \phi \quad (6)$$

$$LN_m(S_{id1}) \cap LN_m(S_{id2}) \cap LN_m(S_{id3}) \dots \cap LN_m(S_{idn}) = \phi \quad (7)$$

where,

- k be the number of LN which cache data item in MANET.
- n be the number of service items cached in m^{th} LN and $0 < m < k$.
- S_{id} is web service index entry in LN_TABLE of a LN.

Eq. 6 refers that no same service is cached by the different LNs and Eq. 7 refers to no same service is within a LN. Thus, there is no repetition in the web service index entry in LN_TABLE of every LN. So, every service is cached only once in the MANET.

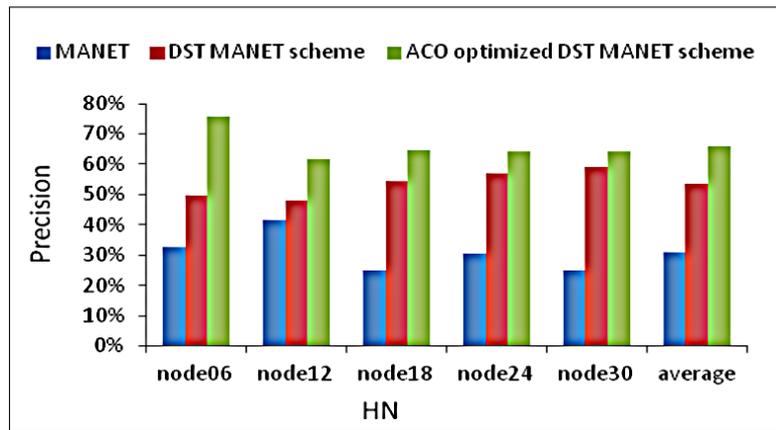


Figure 2: Comparison on Precision performance of three different schemes

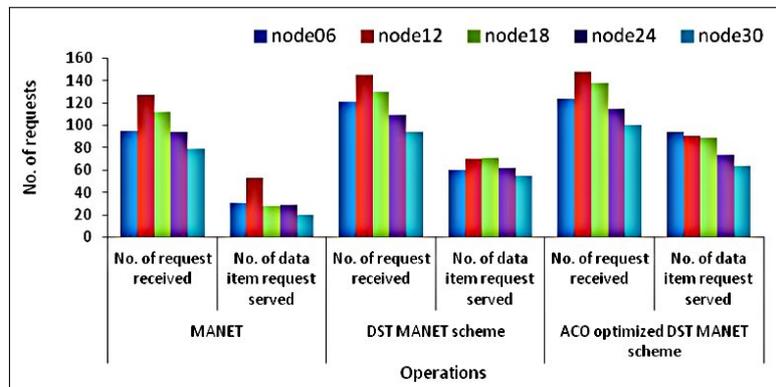


Figure 3: Comparison on service request received and served using three different schemes

3.3 Mobility and hand-off model

Due to the fragile nature, mobile nodes are set to move free in MANET and node tracking task added complex. This tracking task can be accomplished in two scenarios, exit or remove from the MANET [5] and switching among the DSTs within the MANET.

SCENARIO 1: A node can exit/remove from the network dynamically in the following fashion: The node that wants to get remove from the network should send an inform message to its HN, so that the corresponding HN removes the node details and its cached web service identity from its LNs list, *which obviously removes from the spanning tree.*

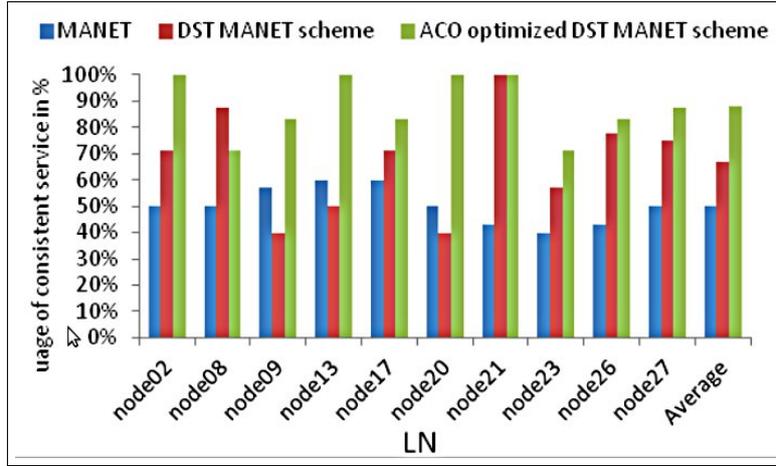


Figure 4: Comparison on % of utilization of consistent service item performance of three different schemes

SCENARIO 2: Switching of node from one DST to another is configured automatically by the HN by passing some messages with the node. Any LN can voluntarily hand-off itself from any HN take HN_1 , if the condition satisfies that number of hops between LN and HN_1 exceeds the number of LNs under HN_2 . And the LN can join to HN_2 in which number of hops between LN and HN_2 will be less than number of LNs under HN_2 . This hand-off should be intimated to both HN_1 and HN_2 . *It is must that every LN should be under any HN. If a request arises in the mean time between handover, the HN_1 will transmit a “Binding Warning” and intimate LN is now under HN_2 .*

S.No	LN	MANET				MANET with DST scheme				MANET with ACO optimized DST scheme			
		No. of data items cached	No. of data item request served	No. of request received	Precision (%)	No. of data items cached	No. of data item request served	No. of request received	Precision (%)	No. of data items cached	No. of data item request served	No. of request received	Precision (%)
1	node06	11	31	95	32.62	11	60	121	49.68	15	94	124	75.63
2	node12	13	53	127	41.77	20	70	145	47.98	11	91	148	61.80
3	node18	10	28	112	24.81	15	71	130	54.52	15	89	138	64.61
4	node24	17	29	94	30.62	12	62	109	57.13	14	74	115	64.39
5	node30	10	20	79	24.78	12	55	94	58.94	15	64	100	64.20
Aggregate Performance		60	160	507	30.9	70	318	599	53.7	70	413	625	66.1

Table 2: Comparison on precision for HNs involved in serving the nodes in all three different scenarios

DEFINITION 3: Let v_i be a mobile node in MANET which is under HN_i and HN_j be another HN in the MANET. Then, v_i can decide to hand-off from HN_i and to join under HN_j , if it satisfies the rule of Eq. 9.

If,

$$HN_i, HN_j \in \{HN_1, HN_2, HN_3, \dots, HN_n\} \quad (8)$$

then,

$$nhop(v_i, HN_i) > n(LN_i) \ \& \ \& \ nhop(v_i, HN_j) > n(LN_i) \quad (9)$$

where,

- n be the total number of HNs in MANET.
- $nhops(v, HN)$ be the number of hops required to reach node v from HN.
- $n(LN_i)$ be the total number of LNs under the Head Node HN_i .

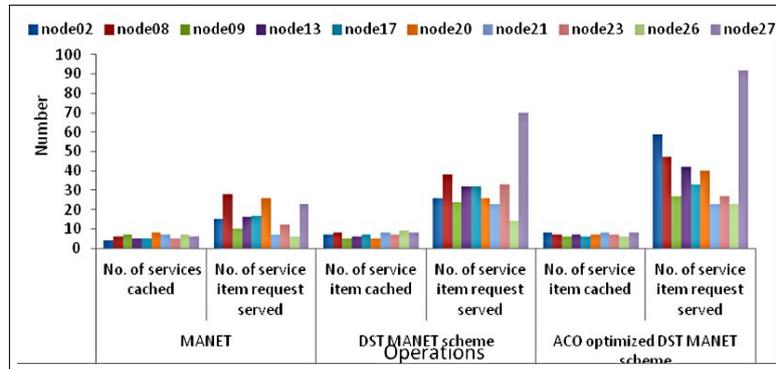


Figure 5. Comparison on data item cached and served using three different schemes.

Thus, any LN whose distance from existing HN is lesser than another HN can perform handoff from existing HN and join with new one. *The Mobility and Handoff model works under the assumption that the HNs will always be in access range within the MANET. This assumption is made as LNs under the MN should not be disconnected from the spanning tree and thereby to the network.*

3.4 Precision

The precision refers to the ratio of total number of service request received to the total number of requested service found in the MANET. The different performance data observed from the simulation in first 100 seconds are tabulated in Table 2; which contain HNs created, No. of services cached by LNs of each HN and No. of service item requests served by each HN, *either at an atomic service or a composition. Composition of web services is carried using the Petri net modelling by the following phases – identifying the similar web services using domain ontology, classification as compatible and non-compatible, execution of compatible web services pertaining to the goal of the service request.* To explain in clear manner, consider the first entry in Table 2, the mobile node, node06 is an HN and the total number of data items cached in its LNs, the total number of data item request served using the cache and the total number of request received for service item are 11, 31 and 95 respectively. Thus Precision for HN node06 is 32.62%.

Table 2 shows that precision percentage of HNs in the MANET can be improved using DST and ACO techniques. The maximum precision value recorded for MANET, MANET with DST scheme and MANET with ACO optimized DST scheme is 41.7%, 58.94 and 75.63% respectively which is illustrated in Figure 2. This is because the DST and ACO techniques reduce the number of message hops required for any operation, such as cache request and cache reply, by discovering an optimal path between the nodes in the MANET. So operations performed faster and more requests are served from nodes local cache within the stipulated time period.

Figure 3 illustrates the comparison on service request received and served using three different schemes. From Figures 3 and 4, it can be observed that in ACO optimized DST scheme outperforms other two schemes. Thus, precision performance of the system can be improved from DST scheme from 30.9% to 53.7% and which is further improved to 66.1% using ACO optimization technique.

S.No	LN	MANET				MANET with DST scheme				MANET with ACO optimized DST scheme			
		No. of service items cached	No. of service item request served	No. of service items deleted (expired) before use	Utilization % of consistent service item	No. of service items cached	No. of service item request served	No. of service items deleted (expired) before use	Utilization % of consistent service item	No. of Service items cached	No. of service item request served	No. of service items deleted (expired) before use	Utilization % of consistent service item
1	node02	4	15	2	50.0	7	26	2	71.4	8	59	0	100.0
2	node08	6	28	3	50.0	8	28	1	87.5	7	47	2	71.4
3	node09	7	10	3	57.1	5	24	3	40.0	6	27	1	83.3
4	node13	5	16	2	60.0	6	32	3	50.0	7	42	0	100.0
5	node18	5	17	2	60.0	7	32	2	71.4	6	33	1	83.3
6	node20	8	26	4	50.9	5	26	3	40.0	7	40	0	100.0
7	node21	7	7	4	42.9	8	23	0	100.0	8	23	0	100.0
8	node24	5	12	3	40.0	7	33	3	57.1	7	27	2	71.4
9	node26	7	6	4	42.9	9	14	2	77.8	6	23	1	83.3
10	node27	6	23	3	50.0	8	70	2	75.0	8	92	1	87.5
Aggregate Performance		60	160	30	50.3	70	318	21	67.0	70	413	8	88.0

Table 3: Comparison on % of utilization of consistent service item by LNs in all three different scenarios

3.5 Service reliability

Service reliability or consistency refers to the correctness of the cached service at the time of access within the MANET. Though consistency technique followed is Time-Based it is not required that every mobile should be synchronized in clock. Every HN which stores the service item type also store metadata about the service item S_i which contain the time at which S_i is being cached. This information is used to check the service item validity. ‘T’ is the constant time value in the MANET which can be varied based on service item being updated in outside network. The factor used to measure the consistency of the system is percentage of consistent service item usage.

Table 3 shows the comparison on % of utilization of consistent service item by LNs in all three different scenarios for first 100 seconds of simulation run. From these statistics, it can be observed that utilization percentage of consistent service item is much improved DST MANET and ACO optimized DST MANET schemes. The same is illustrated in the Figure 4 and 5, which confirms the improved performance of the DST and ACO optimized DST schemes over MANET scheme.

Table 4 shows the utilization increases more using petrinets and deliver compositions. As per the assumption, the mobile agents tender similar service request, a service composition held in the cache will be a suitable candidate for most of the request, rather being a single atomic service. Without loss of generality, we can say that the service served ratio to the atomic service with respect to the composite service will be less. And it is trivial that if an atomic service can satisfy a customer request, a composite service (which includes compatible atomic services) will also satisfy the request in a more efficiently.

3.6 Discussion

To Summarize, the DST structure offers the capabilities that are necessary for a dynamic network like reduced size of routing table, minimizes routing overhead, easy network management, reduced message hops, load balancing and fault tolerance. ACO optimized DST structure provides the system to manage with the highly fragile nature of the MANET and to find the optimal route between HNs and HN & its LNs on demand fashion. This layered approach works in simple, effective and on demand way which makes the system to operate on a fragile environment with asymmetric links and constantly changing topology. Thus, the performance of service cache technique for *Web Service Composition* in MANET has been analysed for Cooperative Cache, Mobility Hand-off, Precision and Data Reliability method. An extensive experimenta-

S.No	LN	MANET with DST and ACO schemes				MANET with DST , ACO and Petri Net			
		No. of service items cached	No.of service item request served	No. of service items deleted (expired) before use	Utilization % of consistent service item	No.of service items cached	No. of service item request served	No.of service items deleted (expired) before use	Utilization % of consistent service item
1	node02	8	59	0	100.0	8	63	0	100.0
2	node08	7	47	2	71.4	7	58	1	85.7
3	node09	6	27	1	83.3	6	41	0	100.0
4	node13	7	42	0	100.0	7	48	1	85.7
5	node18	6	33	1	83.3	6	33	0	100.0
6	node20	7	40	0	100.0	7	42	0	100.0
7	node21	8	23	0	100.0	8	29	0	100.0
8	node24	7	27	2	71.4	7	35	2	71.4
9	node26	6	23	1	83.3	6	27	1	83.3
10	node27	8	92	1	87.5	8	95	0	100.0
Aggregate Performance		70	413	8	88.0	70	471	5	92.6

Table 4: Comparison on % of utilization of consistent service item Vs a composition by LNs

tion has been performed on precision and service reliability of the system which confirms that the ACO optimized DST scheme improves the efficiency of service cache technique in MANET environment.

4 Conclusion

The work presented in this paper described modeling and assessing the various performance factors for the Service Cache Mechanism *for composing web services* using DST and ACO techniques in MANET proposed in our previous works. A comprehensive theoretical model has been developed for the performance factors such Cooperative Cache, Mobility Hand-off, Precision and Data Reliability methods. In addition to these, the performance improvement of Web Service Cache Mechanism (WSCM) has been proved experimentally for improved based on Precision and Service Integrity factors using three different schemes such as MANET, DST MANET and ACO optimized DST MANET. The simulation results shows that the precision performance of the WSCM system is improved using the DST scheme from 30.6% to 53.6% and which is further improved to 68.1% using ACO optimization scheme. And the service reliability performance is enhanced from 50.3% to 67.0% and to 88.0% using DST and ACO optimization schemes respectively.

Acknowledgements

This work is a part of the Research Project sponsored under the Major Project Scheme, UGC, India, Reference No: F. 41-639/July 2012 (SR). The authors would like to express their thanks for the financial support offered by the Sponsored Agency.

Bibliography

- [1] R. Baskaran, P. Victor Paul and P. Dhavachelvan (2012), Ant Colony Optimization for Data Cache Technique in MANET, *Int. Conf. on Advances in Computing (ICADC 2012), Advances in Intelligent and Soft Computing series, India., Springer, 873-878.*

-
- [2] Hassan Artail and Khaleel Mershad (2009); MDPF: Minimum Distance Packet Forwarding for search applications in mobile ad hoc networks , *IEEE Transactions on Mobile Computing*, 8(10): 1412 - 142.
- [3] Lan Wang and Stephan Olariu (2005); Cluster Maintenance in Mobile Ad-hoc Networks, *Springer Science + Business Media*, 8: 111-118.
- [4] P. Krishna, M. Chatterjee, N. Vaidya and D. Pradhan(1997); A cluster-based approach for routing in ad-hoc networks, *ACM SIGCOMM Computer Communication*, 27(2): 49-64.
- [5] P. Victor Paul, N. Saravanan, S.K.V. Jayakumar, P. Dhavachelvan, R. Baskaran (2008); QoS Enhancements for Global Replication Management in Peer to Peer networks, *Future Generation Computer Systems*, Elsevier, 28(3):573-582.
- [6] P. Victor Paul, D. Rajaguru, N. Saravanan, R. Baskaran and P. Dhavachelvan (2013); Efficient service cache management in mobile P2P networks, *Future Generation Computer Systems*, Elsevier, 29(6): 1505-1521.
- [7] Sylvain Dahan (2005); Distributed Spanning Tree Algorithms for Large Scale Traversals, *11th International Conference on Parallel and Distributed Systems (ICPADS'05)*, DOI: 10.1109/ICPADS.2005.131 , 1: 453-459.
- [8] P. Victor Paul, T. Vengattaraman, P. Dhavachelvan (2010); Improving efficiency of Peer Network Applications by formulating Distributed Spanning Tree, *Third International Conference on Emerging Trends in Engineering & Technology (ICETET-2010)*, IEEE, India, 813-818.
- [9] R. Baskaran, P. Victor Paul and P. Dhavachelvan (2012); Algorithm and Direction for Analysis of Global Replica Management in P2P Network, *IEEE International Conference on Recent Trends in Information Technology (ICRTIT)*, May 2012, Chennai, 211 - 216.
- [10] Sylvain Dahan, Jean-Marc Nicod and Laurent Philippe (2005); The Distributed Spanning Tree: A Scalable Interconnection Topology for Efficient and Equitable Traversal, *International Symposium on Cluster Computing and the Grid*, 2005 IEEE.
- [11] Sylvain Dahan (2005); Distributed Spanning Tree Algorithms for Large Scale Traversals, *11th International Conference on Parallel and Distributed Systems (ICPADS'05)*, DOI: 10.1109/CCGRID.2005.1558561, 1: 243-250.
- [12] R. Friedman (2002); Caching Web Services in Mobile Ad-Hoc Networks: Opportunities and Challenges, *Proc. Second ACM Intl Workshop Principles of Mobile Computing*, 90-96.
- [13] J. Zhao, P. Zhang, G. Cao, and C. Das (2010); Cooperative caching in wireless p2p networks: Design, implementation, and evaluation, *Parallel and Distributed Systems, IEEE Transactions on*, 21(2):229-241.
- [14] S. Lim, W. Lee, G. Cao, and C. Das (2006); A Novel Caching Scheme for Internet Based Mobile Ad Hoc Networks Performance, *Ad Hoc Networks*, 4(2):225-239.
- [15] N. Chand, R. C. Joshi, and M. Misra (2006); Efficient cooperative caching in ad hoc networks. COMSWARE, DOI:10.1109/COMSWA.2006.1665190, 1-8.
- [16] Fan Ye, Qing Li, and EnHong Chen, Adaptive caching with heterogeneous devices in mobile peer to peer network, *ACM Symposium on Applied Computing (SAC '08)*. ACM, New York, USA, 1897-1901.

- [17] Guohong Cao, Liangzhong Yin, Chita R. Das (2004); Cooperative Cache- Based Data Access in Ad Hoc Networks, Pennsylvania State University, *IEEE Computer Society*, February 2004, 32-39.
- [18] Hassan Artail and Khaleel Mershad (2009); MDPF: Minimum Distance Packet Forwarding for search applications in mobile ad hoc networks, *IEEE Transactions on Mobile Computing*, 8(10): 1412 - 142.
- [19] Dorigo, M., Maniezzo, V., Colorni A. (1991); *The ant system: An autocatalytic optimizing Process. Tech.Rep. 91-016 Revised*, Politecnico di Milano, Italy.
- [20] A. Colorni, M. Dorigo, V. Maniezzo (1991); Distributed optimization by ant colonies, *Proc. of ECAL91 European Conference on Artificial Life*, Elsevier Publishing, Amsterdam, The Netherlands, 134-142.
- [21] P. Victor Paul, T. Vengattaraman, P. Dhavachelvan and R. Baskaran (2010); Improved Data Cache Scheme using Distributed Spanning Tree in Mobile Adhoc Network, *The International Journal of Computer Science and Communication (IJCSC)*, 1(2): 329-332.