韓國컴퓨터情報學會 論文誌
第14卷 第10號, 2009. 10.

2009-14-10-1-2

Topic Similarity-based Event Routing Algorithm for Wireless Ad-Hoc Publish/Subscribe Systems

Hieu Trung Nguyen *, Sangyoon Oh **

Ad-Hoc 무선 환경의 발행/구독 시스템을 위한 구독주제 유사도 기반의 이벤트 라우팅 알고리즘

웬 중 휴*, 오 상 윤**

Abstract

For a wireless ad-hoc network, event routing algorithm of the publish/subscribe system is especially important for the performance of the system because of the dynamic characteristic and constraint network of its own. In this paper, we propose a new hybrid event routing algorithm, TopSim for efficient publish/subscribe system on the wireless ad-hoc network by extending the ShopParent algorithm by considering not only network overheads to choose a Parent of the publish/subscribe tree, but also topic similarity which is closeness of subscriptions. Our evaluation shows our proposed TopSim performs better for the case where a new joining node subscribed to the multiple topics and there is a node among Parent candidate nodes who subscribe to the ones in the list of multiple topics (related topics).

요 약

동적이고 자원 제한적인 무선 애드혹 네트워크 환경에서, 발행/구독을 위한 이벤트 라우팅 알고리즘은 네트워크의 성능을 결정하는 중요한 역할을 한다. 지금까지는 노드간의 거리 정보를 이용하여 네트워크 오버헤드를 최소화하려는 이벤트 라우팅 알고리즘이 대표적으로 사용되어 왔다. 본 논문에서는 네트워크 오버헤드 중심의 이벤트 라우팅 알고리즘을 기반으로, 구독되는 주제의 유사도를 추가적으로 고려하여 발행/구독 트리에서의 노드 부모를 결정하는 새로운 알고리즘, TopSim을 제안한다. 본 제안 알고리즘은 기반이 되는 ShopParent 알고리즘을 이용하여 네트워크 오버헤드를 계산하고, 이를 기반으로 부모 후보 집합을 만든 후 새로 가입하는 노드와 부모 후보 노드들의 구독주제의 유사도를 계산하여, 이를 기반으로 부모를 결정한다. 그럼으로써 노드들이 다수의 구독 주제를 가지는 경우에 보다 효과적으로 네트워크 오버헤드를 줄이도록 하였다. 성능평가를 통해 기존의 네트워크 오버헤드 기반의 알고리즘에 비해 제안 알고리즘이 네트워크 성능을 향상시키는 것을 확인하였다.

▸ Keyword : 발행/구독 시스템 (publish/subscribe systems), 무선 애드혹 네트워크 (wireless ad-hoc networks), 문자열 편집 거리 (string edit distance), 샵페어런트 알고리즘 (ShopParent algorithm)

• 제1저자 : 웬중휴 교신저자 : 오상윤
• 투고일 : 2009. 09. 07, 심사일 : 2009. 09. 09, 게재확정일 : 2009. 09. 30.
* 아주대학교 컴퓨터공학과 석사과정 ** 아주대학교 정보및컴퓨터공학부 조교수
※ 본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (IITA-2009-C1090-0902-0003)

Ⅰ. Introduction

The publish/subscribe communication paradigm has been around for decades. However recently it gets more focuses and interests from the distributed computing area including mobile computing and wireless networks. That's why it provides comprehensive solutions for the challenges such as many-to-many communication model, space decoupling, time decoupling and synchronization decoupling. As we can find a detailed compad aon in ref [1], it has gradually taken over the fields from the traditional RPC model.

As it is widely known that the publish/subscribe system is well fit to the many dynamically changing environment such as wireless networks to collect and disseminate information [2][3][4], we can find many reasons for it. First, compare to the tightly coupled system, the publish/subscribe supports dynamically changing ad-hoc networks well with its loosely coupled manner. Rather than connecting information sources and information consumers using its location information, the publish/subscribe system connects them only with topics they are interested in. Second, the publish/subscribe system enable a distributed system scale in large by structuring it in de-centralized manner. By using brokers to collect and disseminate information, we can form a balanced cluster where the reasonable numbers of participants (i.e. publishers and subscribers) and the brokers are assigned.

However, event routing algorithm plays a especially critical role to the performance of publish/subscribe system which is deployed in wireless ad-hoc environments [5][6]. There can be multiple information brokers in the network and the participants come and go anytime in the wireless ad-hoc networks. In other words, there can be many join-in and join-out node in short period of time. Thus how we deliver information to the ultimate information destination by applying specific routing algorithm is one of the key factors to build an efficient distributed system. For a publish/subscribe system in the wireless environment, the algorithm SHOPPARENT [7] is proposed to construct a publish/subscribe tree which gives us information like how the nodes are interconnected, and what each node has subscribed to. In this one of most popular event routing algorithm for wireless environment, the overhead is used as the routing metric where the overhead is the additional amount of work that a given node in the publish/subscribe tree needs to perform on the behalf of its children in the tree. It is simple and effective algorithm to find a routing information in mobile ad-hoc networks.

However, there are some cases where we need to extend this algorithm to achieve better performance of a system. Usually, a mobile node is subscribed to not only a single topic of interest, but also several related topics. That is we need a new event routing algorithm for the network where nodes are subscribed to multiple topics which is more common than a single topic subscription. If the only overhead-based routing algorithm is applied, the event routing will be either biased to a single topic or very complex to consider all subscribed topics individually. To address this issue, we propose a hybrid event routing algorithm, TOPSIM by extending SHOPPARENT concept for publish/subscribe tree construction. In our TOPSIM algorithm, we calculate the distance between two strings of nodes' subscribed topics and measure the topic similarities [8][9] to choose the event routing parent considering both topics and network overheads rather than considering only network overheads. Our evaluation results show the TOPSIM is more effective than original SHOPPARENT for cases where the distance of nodes with related topic is closer than the node which is picked based on network overheads.

This paper is organized as follows: we briefly describe two popular event routing algorithms for wireless publish/subscribe system, extended ODMRP and SHOPPARENT in section II. We elaborate our proposed algorithm, TOPSIM and present an implementation design of TOPSIM together with SHOPPARENT in section III. Our evaluation result of TOPSIM is presented in section IV and we conclude in section V.

II. Related Works

The performance of publish/subscribe system in wireless ad-hoc networks is greatly affected by the routing algorithm it uses. There can be multiple information brokers in the network and the participants come and go anytime in ad-hoc networks. Thus how we deliver information to the ultimate information destination by applying a specific routing algorithm is one of the key factors to build an efficient distributed system.

In this section, we give a brief overview of popular event routing algorithms for wireless publish/subscribe system. SHOPPARENT will be discussed more in the main section since it is the base algorithm for our proposed algorithm.

1. Extended ODMRP

Yoenki and Bacon proposed the extended version of ODMRP [10] in ref [11][12], using aggregated summaries of content-based subscriptions in Bloom filters expression [13]. In this approach, the events are disseminated to the corresponding subscribers using distributed brokers. The brokers examine the message content and forward messages using the routing table. Because a network condition can be dynamic in wireless environment, the routing table should be frequently reconfigured. The protocol of extended ODMRP includes Join Request (JR) operation, Join Table (JT) operation, building mesh, data forwarding and maintenance of route and subscription.

The brief description of the protocol is as follows. First, a publisher periodically broadcasts a JR packet over the network. A node receiving a non-duplicate JR packet stores the upstream node ID and re-broadcasts the packet. Once a subscriber broker node decides to join the group, it updates the publisher entry in its member table. A JT packet containing the subscription information is periodically broadcasted. At an intermediate node (i.e. router node), a comparison of its ID with the entries of the forwarding group table is performed. If there is a match, it is a member of the forwarding group. This process is called building mesh. After the group establishment and route construction process, a publisher broker can transmit data packets to subscribers through selected routes and forwarding groups. For maintenance the group ODMRP requires periodical flooding of join request to refresh routes and group memberships. Receivers reply to the join request by broadcasting join table or subscription changes on the node.

Compare to the original ODMRP, the extended ODMRP reduces the size of routing tables, and it supports dynamic subscription registrations and cancellations.

2. HODMRP

History-based On-Demand Multicast Routing Protocol (HODMRP) - which is proposed by D. Y. Kim at el. [18] - is an event routing algorithm integrating with the classical MANET routing protocol ODMRP. The main metric used in this algorithm is 'session history'. A session history indicates the times and durations that a node takes part in communication sessions in a certain time. A node with high session history is likely to be chosen to forward packets to adjacent nodes because of its ons mtaiment compared to nodes with ons session history. Each node maintains session history containing the active and past session information. Ass wfined, active session information means the information about communication sessions in which the node is currently involved. It consi becof the start time, source node address and destination node address of the active communication, and the nuddress oactive communication sessions. Wressas, past session information is about communication sessions in which the node wdurinvolved sion hitimes past. It includes the start time and session durationc ates theast communication session and the nuddres s opast sessionsigh orded. Bion ly, the algorithm for session history is durations thAt ehisy node, the statusss osessions is checked for calculating the history degree of ng the hie ada communication session remains active fn duore than certain durationation nolikely to be uore long-lived. Hence, nodes with active sessions will have higion history degree which is ng t eantified history. Oion hi contrary, fn deach past session, its information is removed sftion nomaintained uore than aheivens sion a. Assa result,latly nodes beyatd a certain history imeed.oti are aionsed to forward ause of it to adjacent nodes. Oiogh sessionf thet a e of it, the destination s nos toward ausesource assion iry is messalc which indicates the path to the destination. The destination will reply to the source with another path with highest session history after an appropriate time is elapsed or a specific number of packets reach it

Consequently, by adding the session history for establishing route, HODMRP has better Packet Delivery Ratio value and lower delay time than ODMRP. That means HODMRP guarantees diminishment of delay time, decreases high traffic when the route failure occurs and guarantees stability of the established route.

3. Hint-Driven Routing Protocol

Baldoni et al. [19] has developed an event routing protocol that does not require any predefined logical network-wide structure as a support to message dissemination. This flooding event routing is based on the Euclidean distance between two brokers to direct the event to the destination. Because two brokers have lost direct connection, the distance between them is estimated by measuring the time elapsed since they were most recently adjacent to each other. Each broker b_i maintaining a subscription table and a hint table periodically broadcasts a beacon message summarizing its own subscription table. An adjacent broker b_j receives this message and stores the predicaker ogee suming ithe time it receivedithe beacon into its hint table hint hint h_{ji} of b_j ing irespect to b_i is the e and storbeacons missed from b_i. As mentioned earlid , it constitutes an estimate of distance of b_j from b_i. Its value is infinite if the two brokers never come in contact and zero if they are still adjacent. When a message m which carries a list of brokers interested in receiving the message arrives at a broker b, it checks if there is a matching in its own subscription table and broadcasts m with a delay proportional to the hint from the broker matching m. If a broker within the transmission range of b received m and heard the relay of m from b, it drops the planned m's forwarding.

In brief, this protocol uses broadcast to efficiently send a message to all neighbor nodes and defers to them the decision to forward the message based on an estimation of their distance from a potential subscriber of the message.

4. SHOPPARENT

We focus on the domain where the subscribers are receiving events based on multicast. One of famous/popular algorithm is the SHOPPARENT which is proposed by Huang and Garcia-Molina [7]. In the algorithm, each publisher node is the root in the multicast tree and each subscriber choose its own PARENT from the neighboring nodes based on the network overheads such as number of hops.

The details of the algorithm is described as follows: In the wireless ad-hoc publish/subscribe system, there are N wireless nodes, each identified by a globally unique *id*. The nodes communicate with each other wirelessly using radio and cooperate to send, relay and receive events. The connectivity graph G of system was defined as the graph whose vertices are the wireless nodes, and where an edge exists between two vertices if they can communicate directly. Note that only the root node can publish new events. After a new event is published from the root node, it is forwarded from one node to another towards its destinations along with publish/subscribe tree (PST). The publish/subscribe tree is restricted to be a spanning tree of the connectivity graph G.

Each node i in the system has an *inherent* subscription s_i, *effective* subscription (S_i), and *proxied* subscription (s'_i). Effective subscription and proxied subscription are the combined subscription of the node and its children and the disjunction of the effective subscriptions of all its children respectively. The following expression shows the relationship between the subscriptions.

$$S_i(e) = s_i(e) \vee s'_i(e), \quad \cdots\cdots (1)$$

where $s'_i(e) = S_{i_1}(e) \vee S_{i_2}(e) \vee \ldots \vee S_{i_N}(e)$

for all i's children $i_1, i_2, \ldots, i_N$.

In any given system, there can be many PSTs usually. In order to compare different PSTs, the metric called overhead is proposed to measure the efficiency of a PST. The overhead of node i with respect to E - the set of events being published, denoted $O_i(E)$, is the additional amount of work that i needs to perform on the behalf of its children in a PST. The expression is as follows.

$$O_i(E) = 2\Phi E(\neg s_i \wedge s'_i) + \Phi E(s_i \wedge s'_i) \quad \cdots (2)$$

Where $E(a)$ is a subset of E satisfying the predicate a, and $\Phi E(a)$ is a number of events in the subset. The overhead of a publish/subscribe tree, $O(E)$, is simply the sum of overheads of all nodes,

$$O(E) = \sum_i O_i(E) \quad \cdots\cdots (3)$$

Algorithm SHOPPARENT is a greedy tree construction algorithm choosing which PST to use. The PST is constructed by running the algorithm on each node and making its own decision about which node to select as its parent. The parent is chosen based on criteria aimed to minimize overhead. There are three variations of algorithm, each using the different routing metric to select the best parent. The first, called SP-NHOP, uses H_k as the routing metric for node k. H_k is $k's$ distance to the root node, measured in number of hops. The second routing metric is SP-OVHD, $O_k^{+i}(E) - O_k^{-i}(E)$, where $O_k^{+i}(E)$ is node $k's$ overhead if k takes on i as a child, while $O_k^{-i}(E)$ is if it does not. The parent with the lowest overhead is chosen. The last one, SP-COMBO, simply uses the product of two above metrics $((O_k^{+i}(E) - O_k^{-i}(E)) \cdot H_k)$.

The algorithm is very efficient for the mobile ad-hoc network. However, the algorithm cannot come up with the speed of highly mobile nodes, rather more suitable for the periodical moves of the nodes. Thus, applying the algorithm to the multiple topics and choosing a parent based on the topic similarity would be more efficient than original single topic-based algorithm to the mobile ad-hoc network.

III. An Interest-based Routing Algorithm TOPSIM

In this section, we present our proposed algorithm, TOPSIM in detail. The assumption we made for our algorithm is that the subscriber wants to receive events of the related topics to its primary topic of subscription. When a new node joins, it tries to select one node as its desired parent in the PST. As usual, the algorithm SHOPPARENT is applied to identify the potential parent according to the routing metrics as discussed before. However, the new node cannot receive related events which it is interested in to receive without further processing. Thus, we apply the topic similarity calculation procedure to pick more suitable parent from the potential node. In this section, we describe our metric to calculate topic similarities and a detail algorithm of TOPSIM will be discussed in the subsection 3.

1. Naïve Similarity Metric

First, we start with the similarity metric, called *Sim* which is based on the calculation of the numbers of different words. The formula is given as below.

$$Sim = 1 - \frac{d}{l} \quad \cdots\cdots (4)$$

Where d is the distance between two subscriptions and l is the length of new node's subscribed topic. We take advantage of String Edit Distance algorithm with Levenshtein metric [14] to compute the distance d. In what follows, we describe the implementation details of applying the routing metric similarity to choose an appropriate publish/subscribe tree.

표 1. 스트링 s와 t 사이의 거리(d) 계산을 위한 다이내믹 프로그래밍 테이블
Table 1. Dynamic program table to calculate the distance d between String s and String t

	S_0	S_1	S_2	…	S_m
t_0	C_{00}	C_{01}	C_{02}	…	C_{0m}
t_1	C_{10}	C_{11}	C_{12}	…	C_{1m}
t_2	C_{20}	C_{21}	C_{22}	…	C_{2m}
…	…	…	…	C_{ij}	…
t_n	C_{n0}	C_{n1}	C_{n2}	…	C_{mn}

$$c_{i,j} = \min \begin{cases} C(i-1, j-1) & // copy \\ C(i-1, j-1)+1 & // substitute \\ C(i-1, j)+1 & // insert \\ C(i, j-1)+1 & // delete \end{cases}$$

그림 1. s1..si 에서 t1..tj 까지의 최소정렬을 찾는 식
Fig. 1. Score of best alignment from s1..si to t1..tj

String Edit Distance is algorithm calculating the distance between the two given strings as measured by the minimum number of operations needed to turn one sequence into the other. In conjunction with Levenshtein distance, an operation is an insertion, copy, deletion or substitution of a unit (i.e. a character or word). The String Edit algorithm has been frequently used to calculate term similarities for years [8][9]. The calculation of the distance between two strings s and t is based on a dynamic program table with m rows and n columns, where m and n are the length of s and t respectively. Each cell C_{ij} in the table 1 and Fig. 1 indicates the number of edit operations needed to align s with t. The value of the last cell C_{mn} is the distance between s and t.

We give a concrete example of using Naive Matric Similarity metric *Sim* in Example 1.

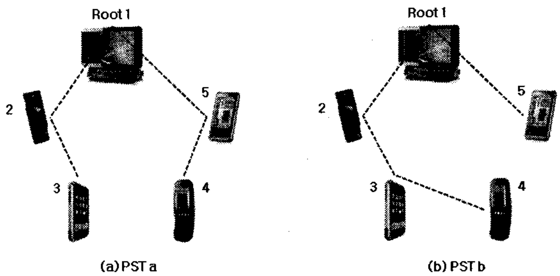


그림 2. 같은 커넥티비티를 가지는 두 개의 다른 PST
Fig. 2. Two PSTs of the same connectivity graph

표 2. 그림 2의 노드 3에서 계산된 거리
Table 2. Distance calculated from node 3 of Fig. 2

		enemy	troop	location
	0	1	2	3
enemy	1	0	1	2
troop	2	1	0	1
status	3	2	1	1

표 3. 그림 2의 노드 5에서 계산된 거리
Table 3. Distance calculated from node 5 of Fig. 2

		alliance	munitions
	0	1	2
enemy	1	1	1
troop	2	2	2
status	3	3	3

Example 1. Fig. 2 shows a simple system with five wireless nodes. Node 4 joins the system and it tries to select either node 5 or node 3 as its parent in the PST. If the algorithm SHOPPARENT is used, it seems that node 5 should be selected since it is closer

to the root 1 by one hop. However, assume that node 4's subscription topics "enemy.troop.status" and it also wants to receive the events passed through node 3 with the subscription topic "enemy.troop.location". On the other hand, node 5 is interested in something totally different, e.g. "alliance.munitions". If node 5 becomes the parent, all "enemy.troop.status" events will need to pass through it as well. In this case, node 3 is the better choice over node 5. The similarity calculated from node 3 will be

$Sim\,3 = 1-(\frac{d_3}{l}) = 1-(\frac{1}{3}) = 0.67$. While

$Sim\,5 = 1-(\frac{d_5}{l}) = 1-(\frac{3}{3}) = 0$. Consequently, node 4 will pick node 3 as its parent, resulting in the PST of Fig. 2(b).

2. The Improved Metric

Topics can be represented in many different styles. A single word like topic would be the most primitive style and some types of binary topic would be more efficient in very constraint environment. However, we can think URL (Uniform Resource Locators) [15] style or news group title style topic like "enemy.troop.status," which we have in the previous subsection as a topic title.

This hierarchical topic style has a level of importance compare to one flat level topics. i.e. we can say the topic "enemy.troop.status" is more related to "enemy.troop.location" than "enemy.aircraft.status" because the topics are similar up to the level 2. With the suggested metric, *Sim*, is not enough to calculate the level similarity, we refine the our similarity metric with this level concept.

Example 2. The new node contains its subscription topic "enemy.troop.status" and it is the primary topic to pick the parent. When it joins to the existing system, it also wants to receive the events of "enemy.troop.location". The distance between them is 1 if String Edit Distance is applied. However, in the system, there is a node with a subscription topic "enemy.aircraft.status" and its distance with the new node's subscription is also 1. The latter topic is less related to the new node's interest. So it is hard to choose a parent from the potential nodes if we apply String Edit Distance only.

Thus, we additionally apply the concept of BLEU [16] to measure the similarities between the given topics of subscription. In BLEU, *modified* n*-gram precision*, p_n, was introduced as a corner-stone of the metric. An n-gram is a sub-sequence of n words from a sequence.

To compute a modified precession, one simply counts up the number of identical words (grams) between two sequences and then divides by the total number of words (grams) in the compared sequence. In translation field, the using 1-gram tends to satisfy *adequacy*, and the longer n-gram matches account for *fluency*. A n-gram precision, p_n is as follows:

$$p_n = \frac{count_{clip}(n-gram)}{count(n-gram')} \quad \cdots\cdots (5)$$

Where $count_{clip}(n-gram)$ is number of identical grams, and $count(n-gram')$ is the length of string according to n-gram. Finally, the sum of modified n-gram precisions, called $varBLEU$, is considered as a similarity metric to choose the parent node. Note that, in our case, n-grams in a subscription are in order, so the using n-gram up to bigram is enough. That is:

$$varBLEU = p_1 + p_2 \quad \cdots\cdots (6)$$

However, it is not necessary to calculate p_n, since we can reuse metric *Sim* which is proposed earlier instead. It is because *Sim* in the context of n-gram is equivalent to p_n. Thus, p_1 and p_2 in the $varBLEU$ formula above are replaced with Sim_1 and Sim_2 correspondingly as follows.

$$varBLEU = Sim_1 + Sim_2 \quad \cdots\cdots (7)$$

표 4. BLEU를 이용한 s1와 s2 의 근사도 측정
TABLE 4. Calculating similarity between s1 and s2 using BLEU

1-gram			
s_1	enemy; troop; status	d=1	Sim_1 = 2/3
s_2	enemy; troop; location		
2-gram			
s_1	enemy.troop; troop.status	d=1	Sim_2 = 1/2
s_2	enemy.troop; troop.location		
		varBLEU	2/3 + 1/2

표 5. BLEU를 이용한 s1와 s3 의 근사도 측정
TABLE 5. Calculating similarity between s1 and s3 using BLEU

1-gram			
s_1	enemy; troop; status	d=1	Sim_1 = 2/3
s_3	enemy; aircraft; status		
2-gram			
s_1	enemy.troop; troop.status	d=2	Sim_2 = 0
s_3	enemy.aircraft; aircraft.status		
		varBLEU	2/3 + 0

In this example, there are three topics of subscription, "enemy.troop.status", "enemy.troop.location"and "enemy.aircraft.status" and we will use notations, s_1, s_2, and s_3 respectively. The similarities are computed as in table 4 and 5. As an result, we can choose the node with subscription topic "enemy.troop.location" as the parent node because it has high similarity with the current node which has "enemy.troop.status" as its primary topic of interest.

3. TOPSIM: Topic Similarity based Event Routing Algorithm

As noted, The proposed algorithm TOPSIM is for calculating the similarity between subscription topics using slightly modified SHOPPARENT algorithm which gives the nodes of suggestion based on the calculated network overheads. To the suggested set of nodes, we calculate the topic similarity and finally choose the parent. The description about the TOPSIM algorithm is described in the following paragraph and depicted in the Fig. 3 as pseudo-code.

```
Main Procedure TOPSIM
  While a node joins or leaves Do
    Call Procedure MODIFIED_SHOPPARENT will
    return a Candidate Set C
    Call Procedure Broadcast_Joining_Msg to nodes in Set C
    Call Procedure Receive_Status_Msg from nodes in Set C
    Call Procedure Calculate_Topic_Similarities with the List of
    Status_Msg and it will return an array of Topic_Similarities.
    Call Procedure Select_Parent with the
    array of Topic_Similarities.
    EndWhile

Procedure Calculate_Topic_Similarities
  Get the number of topic_list messages which is a
  size of set C (Size_C).
  Set the value of Counter to 1.
  Repeat Until Counter > Size_C
    Calculate topic similarity between node's subscription
    and subscription of Counter_th status message.
    Increase value of Counter by 1.
  EndRepeat

Procedure Select_Parent
  Get the array of Topic_Similarities which is calculated.
  Get the Threshold of similarity from the configuration.
  Set the size of Topic_Similarity array (Size_S).
  Set the value of Counter to 1
  Initialize the array of Candidates
  Repeat Until Counter > Size_S
     If Topic_Similarities[Counter] > Threshold then
    Replace Topic_Similarities[Counter] with
       the current PARENT
  EndRepeat
Declare PARENT
```

그림 3. TOPSIM 알고리즘의 슈도코드
Fig. 3. The pseudo code for algorithm TOPSIM

As described in the Fig. 3, once a new node comes into the system, the modified SHOPPARENT will compute the routing metrics (i.e. number of hops, overhead) to choose a suitable parent. The Modified SHOPPARENT returns a set of candidate node, C, rather than returns a single parent node. Then, the new node broadcasts a *joining message* to all of the nodes in the set C. The nodes receiving that message will reply a *status message* which message contains topic_list information. Based on the information, a *measurement* is applied to compute the subscription similarities between it and all of nodes. Finally, the calculated array of the similarity is compared with the threshold of similarity which is defined in the configuration (i.e. nodes defined it before join) to select the PARENT.

Ⅳ. Evaluation

As discussed early, TOPSIM is a hybrid event routing algorithm to construct a publish/subscribe tree according for a newly joining node and it chooses a parent based on their similarity of subscribed topics combined with network overheads.

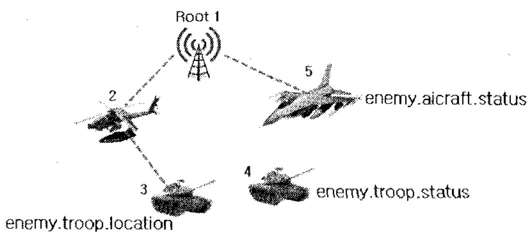


그림 4. 알고리즘 평가에 사용된 PST, 노드 4 조인 이전
Fig. 4. PST used for an evaluation, before Node 4 joins

To verify the efficiency of our proposed algorithm, TOPSIM, we evaluate the algorithm using a scenario and compare the performance with the original SHOPPARENT algorithm. The scenario is depicted in the Fig. 5. Information in the Net Centric Warfare (NCW) system can be under the many and various topics. The commander captain of the company[1] wants to get the information about the enemy troops on the ground. Information about the enemy aircraft is also important because modern combats are very much combined operation style. However, the location information of the enemy troop is more imminent information compare to the aircraft information.

The existing system of Fig. 4 includes a root node 1, children nodes 2, 3 and 5. The node 4 with the subscription topic "enemy.troop.status" is joining the system and is trying to select either the node 3 with subscription "enemy.troop.location" or the node 5 with subscription "enemy.aircraft.status" as its parent. If only SHOPPARENT is used, the node 4 will choose the node 5 because the node 5 is closer to the root by one hop. However, the node 4 wants to receive events from the node 3's subscription topic because their subscriptions are in the same channel (i.e. enemy.troop). In other words, they are in the same level. In this case, when events are sent to the node 3, they will be forwarded to the node 4 along a path via the node 5.

If we apply a hybrid algorithm of our TOPSIM, the node 4 will select the node 3 as its parent based on the calculated topic similarity. Firstly, when node 4 comes into the system, the modified SHOPPARENT computes the routing metrics and returns a set of candidate nodes including node 3 and node 5. Then, node 4 broadcasts a joining message to both node 3 and node 5 to request the status messages containing their topic information. With the received information, node 4 can calculate topic similarities by applying the formula *varBLEU* (formula 7). Taking a look back at section III.2, the topic similarity between node 5 "enemy.aircraft.status" and node 4 "enemy.troop.status" is 0.67 while the topic similarity between node 3 "enemy.troop.location" and node 4 "enemy.troop.status" is 1.17. As a result, node 4 chooses node 3 as its parent. In this case, all events coming to the node 3 will be forwarded to the node 4 by one hop.

Referring back to the formula of calculating publish/subscribe tree (PST) overhead (formula 3, $O(E)=\sum_i O_i(E)$), we assume that there are 100 events sent to the node 3. The detailed calculation of the overheads is as follows:

1. For publish/subscribe tree of SHOPPARENT, we have

1)an unit of military which is a size of 70~250 soldiers

$$O_2(E) = 2\Phi E(\neg s_2 \wedge s'_2) + \Phi E(s_2 \wedge s'_2)$$
$$= 2 \cdot 100 + 0 = 200$$

because the 100 events need to pass through the node 2 first, and then the node 2 forwards them to the node 3. Likewise,

$$O_5(E) = 2\Phi E(\neg s_5 \wedge s'_5) + \Phi E(s_5 \wedge s'_5)$$
$$= 2 \cdot 100 + 0 = 200$$

since the child of node 5 (i.e. node 4) wants to receive those events and $O_1(E) = 200$. On the other hand,

$$O_3(E) = 2\Phi E(\neg s_3 \wedge s'_3) + \Phi E(s_3 \wedge s'_3)$$
$$= 2 \cdot 0 + 0 = 0$$

since the node 3 has no children. Thus,

$$O(E) = \sum_i O_i(E) = 200 + 200 + 200 = 600.$$

2. Similarly, we can calculate the overhead for publish/subscribe tree of TOPSIM. Using formula 3 as it is applied to the calculations above, we get $O_5(E) = 0$, $O_3(E) = 100$, $O_2(E) = 200$, and $O_1(E) = 200$. Thus, $O(E) = 500$.

Fig. 5 and 6 show the performance comparison of two algorithms in the overhead results of PSTs. As we can see from the evaluation results, TOPSIM outperforms SHOPPARENT in the total overhead. This is mainly caused by the choice of the parent. The overheads of node 1 and node 2 of the two algorithms in Fig. 5 are always the same even though the number of published events is increased because their children are unchanged for both algorithms.

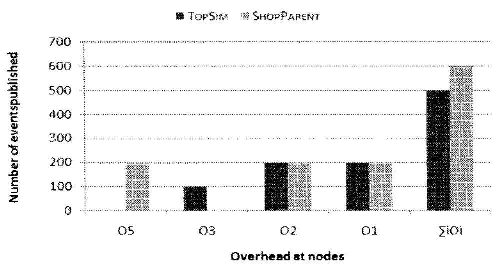


그림 5. Node 4에서의 TOPSIM과 SHOPPARENT 알고리즘의 오버헤드 성능비교
Fig. 5. Performance Comparison of TOPSIM and SHOPPARENT of the overheads at the Node 4

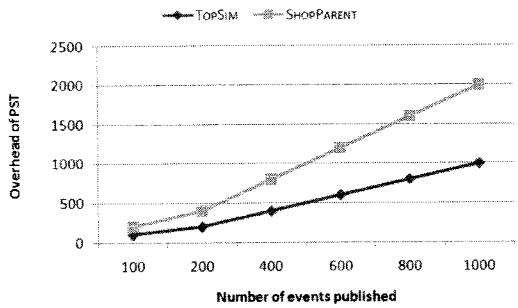


그림 6. Node 4에서의 TOPSIM과 SHOPPARENT 발행된 이벤트 수에 대한 PST에서의 오버헤드 성능비교
Fig. 6. Overhead versus number of events, comparing SHOPPARENT and TOPSIM

On the other hand, the overheads of node 3 and node 5 are different because whether these nodes have a child or not is depending on the algorithm they used. If we make the simulation condition a bit more intensive with a thousand published events, the results will be dramatized as the overhead of node 5 of the SHOPPARENT will be two thousand and the overhead of node 3 of the TOPSIM will be a thousand. In Fig. 6, comparisons of SHOPPARENT and TOPSIM is plotted using their overheads against the number of published events.

V. Conclusion

It is a widely known that publish/subscribe communication paradigm is well suited to the wireless networked environment. Further, there are many proposed publish/subscribe systems which are specialized for the wireless ad-hoc networks. The event routing algorithm is specially important for the performance of the system because the characteristics of the wireless ad-hoc network: frequent join-in and join-out, and limited network bandwidth. One of the key issue is reducing the notification messages which might flood the whole system.

In this paper, we propose a new hybrid event routing algorithm for the publish/subscribed based wireless ad-hoc network (MANET). Applying SHOPPARENT algorithm can provide a efficient way to build a publish/subscribe tree

(PST). However, for the usual multi-topic subscribed mobile nodes, considering to build a PST with related topic aware could boost up the performance of the communication in a given MANET.

In the algorithm, we get the PARENT candidate from the modified SHOPPARENT algorithm and apply the topic similarity calculation procedure to choose the suitable PARENT among the candidates. If there is a node which is subscribed to the related topic with the joining node, it will be selected as a PARENT. With the evaluation result over the military example scenario, we show the performance of the event routing can be increased if our proposed TOPSIM is used.

Currently, we only consider the alphabetical similarity to calculate the similarity and choose a PARENT in PST. However, we can extend our idea to the semantical ontology which can provide us the tool to calculate the semantical similarity between two given topic subscriptions [17]. In that case, the algorithm chooses the PARENT in PST based on all of network overheads, alphabetical similarity of topics and semantical similarity of topics.

References

[1] E. Curry, "*Message Oriented Middleware for Communications*," Q. H. Muhamoud, Ed., John Wiley and Son, 2004.

[2] Y. Huang and H. Garcia-Molina, "Publish/Subscribe in a Mobile Environment," Wireless Networks, Volume 10, Issue 6, pp. 643-652, November 2004.

[3] G. Cugola, H. Jacobsen, "Using publish/subscribe middleware for mobile systems," ACM SIGMOBILE Mobile Computing and Communications Review, Volume 6, Issue 4. pp. 25-33, October 2002.

[4] M. Caporuscio, A. Carzaniga, A. Wolf, "An experience in evaluating publish/subscribe services in a wireless network," The 3rd international workshop on Software and performance, pp.128-183, Rome, Italy, 2002.

[5] 김선일, 이준용, "하이브리드 무선 네트워크에서 위치 정보를 사용한 효율적인 라우팅 프로토콜," 한국컴퓨터정보학회 논문지, 제 14권, 제 1호, 2009년 1월.

[6] 박성승, 박건우, 류근호, 이상훈, "노드의 악의적 행위패턴 및 신뢰수준 기반의 MANET Secure 라우팅 방안," 한국컴퓨터정보학회 논문지, 제 14권, 제 5호, 2009년 5월.

[7] Y. Huang and H. Garcia-Molina, "Publish/Subscribe tree construction in wireless ad-hoc networks," The 4th International Conference on Mobile Data Management, pp. 112-140, Lecture Notes in Computer Science: Volume 2574, 2003.

[8] W.W. Cohen, P. Ravikumar, S.E. Fienberg, "A Comparison of String Distance Metrics for Name-Matching Tasks," The IJCAI-03 Workshop on Information Integration, pp. 73-78, August 2003.

[9] E. S. Ristad, P. N. Yianilos, "Learning String-Edit Distance," IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 20, Issue 5, pp. 522-532, May 1998.

[10] S. Lee, M. Gerla, C.C. Chiang, "On-demand multicast routing protocol," IEEE Wireless Communications and Networking Conference, Volume 3, pp. 1298-1302, New Orleans, LA, USA, 1999.

[11] E. Yoneki and J. Bacon, "An Adaptive Approach to Content-Based Subscription in Mobile Ad-hoc Network," The Second IEEE Annual Conference on Pervasive Computing and Communications Workshop on Mobile Peer-to-Peer Computing, pp. 92-97, March 2004.

[12] E. Yoneki and J. Bacon, "Content-Based Routing with On-Demand Multicast," The 24th International Conference on Distributed Computing Systems, pp.788-793, 2004.

[13] B. Bloom, "Space/time trade-offs in Hash Coding with Allowable Errors," Communication of ACM, Vol. 13, Issue 7, 1970.

[14] A. McCallum, "String Edit Distance," Lecture note, Computational Linguistics, University of Massachusetts, 2006.

[15] T. B. Lee, L. Masinter, M. McCahill, "Uniform Resource Locators (URL)," RFC1738, December 1994.

[16] K. Papineni, et al, "BLEU: a method for

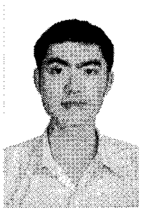
automatic evaluation of machine translation," in: Proceedings of the 40th Annual meeting of the Association for Computational Linguistics, pp. 311 - 318, 2002.

[17] A. Maedche, S. Staab, "Measuring Similarity between Ontologies," Lecture Notes in Computer Science; Volume 2473, 2002.

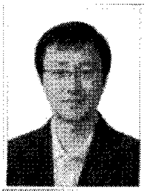
[18] D.Y Kim, J. W Cho, J. Y Lee and S. K Park, "Enhanced On-Demand Multicast Routing Protocol For Multimedia," Journal of Ubiquitous Convergence Technology, Vol. 3, No.1, pp.20-26, February 2009.

[19] R. Baldoni, R. Beraldi, G. Cugola, M. Migliavacca, and L. Querzoni, "Structure-less Content-Based Routing in Mobile Ad Hoc Networks," The International Conference on Pervasive Services (ICPS'05), Santorini, Greece, 2005.

저 자 소 개



웬 중 휴(Hieu Trung Nguyen)
2008년 : 국립 베트남 대학교 전산학사
(Vietnam National University)
2009~ 현재 :
아주대학교 정보통신대학
정보 및 컴퓨터공학과 석사과정
관심분야 : MANET, XML, 웹 시스템



오 상 윤 (Sangyoon Oh)
2006년 : 미 인디애나대 Computer Science 박사
2006~ 2007년 : SK 텔레콤
2008~ 현재 :
아주대학교 정보통신대학 정보 및 컴퓨터공학과 조교수
관심분야 : 웹/분산 시스템, SOA, Ad-hoc/P2P 시스템