This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2007 proceedings.

Efficient Event Delivery in Publish/Subscribe Systems for Wireless Mesh Networks

Yuan Zheng[1], Jiannong Cao[1], Ming Liu[2], Jinling Wang[3]

[1]Internet and Mobile Computing Lab.， Department of Computing，Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
{csyuazheng, csjcao}@comp.polyu.edu.hk

[2] School of Computer Science and Engineering, University of Electronic Science and Technology of China
wing.ml@gmail.com

[3] NEC Laboratories China, Beijing, China
wangjinling@research.nec.com.cn

***Abstract*—Publish/Subscribe (Pub/Sub) systems have been widely used in distributed computing systems for event notification and delivery. However, there is no existing work on Pub/Sub systems for wireless mesh networks (WMNs) which are regarded as a promising infrastructure for providing wireless Internet services to a wide area. In this paper, we propose the design of a Pub/Sub system for WMNs. First, we describe a Pub/Sub system model for WMNs to support mobile clients. Then, based on geographical routing and mobility prediction, we propose an event delivery protocol with low transmission overhead and delay to support mobile clients in a WMN. Our theoretical analysis shows that the transmission overhead and delay per event of the proposed protocol are only affected by the area of the region where the clients move, but not the speeds of mobile clients and the arrival rates of events. The analysis has been validated by the simulation results which show that our protocol can significantly improve the performance of event transmission in Pub/Sub systems for WMNs compared with the previous solutions designed for other kinds of networks.**

Keywords- Middleware, Mobility, Publish/Subscribe, Wireless Mesh Networks

I. Introduction

A wireless mesh network (WMN) is composed of a collection of base stations inter-connected by wireless links and cooperatively transmitting data for users [1]. Taking the advantages of rapid deployment and flexible topology, this emerging networking paradigm is regarded as a promising solution to provide wireless Internet services to users in a large area, such as a high way, a large-scale community, and a metropolitan. One of the major objectives of WMNs is to provide users data services, such as data transmission and data storage, on the Internet through wireless communications.

Publish/Subscribe is a widely used middleware paradigm to provide event services, where the data are encapsulated into events and delivered between distributed event publishers and subscribers [2]-[4]. Many works have been done in the context of the traditional distributed computing environments [2]-[4]. However, these previous solutions are not suitable to WMNs, because WMNs are quite different from the traditional distributed systems due to the existence of a large number of mobile clients and multi-hop wireless communications between network nodes. Some solutions have been recently proposed for designing Pub/Sub systems in the cellular networks to support mobile clients [5],[8]-[14]. They all work in a similar way by establishing a pub/sub tree. On the other hand, the existing Pub/Sub systems [15]-[17] for mobile ad hoc networks (MANETs) have been mainly concerned with the reliable and fault-tolerant event delivery in highly dynamic networks. However, new solutions are needed to fully take the advantages of mesh-based topologies of WMNs, which can help achieve low overhead and delay via multiple disjointed paths. Also, we need to tailor the solutions to consider the characteristics of WMNs of static or very lowly dynamic topology and powerful capability of the WMNs' base stations, which can effectively enhance the reliability of the event delivery compared with MANETs.

In this paper, we propose a solution to the problem of how to support mobile clients in a Publish/Subscribe system for WMNs. We are motivated by the observation that the base stations and mobile clients in WMNs can easily obtain their locations from the attached positioning devices, for example GPS [18]. Inspired by geographic routing and mobility prediction, we propose an event delivery protocol with low transmission overhead and delay to support mobile clients in a Publish/Subscribe system for WMNs. To our best knowledge, our design is the first solution for WMNs, in the sense that it is the first attempt on applying geography-aided routing in Pub/Sub system for WMNs, and using mobility prediction to support continuous event delivery for mobile clients in Pub/Sub systems. Our theoretical analysis shows that the transmission overhead and the delay per event of the proposed protocol are affected only by the area of the region where the clients move, but not the clients' speed and the event arrival rates. The analysis has been validated by the simulation results which show that our protocol can significantly improve the performance of event transmission in Pub/Sub systems for

WMNs compared with the previous solutions designed for other kinds of networks.

The remaining part of our paper is organized as follows. Section 2 describes the system model and basic assumptions. Section 3 describes the design of our Pub/Sub system for WSNs. Section 4 proposes the event delivery protocol based on mobility prediction. Section 5 presents the theoretical analysis and describes the results of the experimental evaluation of our protocol. Section 6 reviews the related work. Finally, Section 7 concludes the whole paper with discussion on our future work.

II. System Model and Assumptions

In this section, we first describe the network model and the mobility model of the mobile clients in WMNs. Then, we describe the model and assumptions of events and filters in our Pub/Sub system.

A. *Network and Mobility Model*

In this paper, we assume that all WMN nodes, including mesh base stations and user devices, are deployed in a square region, which is divided into grids of identical size. Every grid accommodates only one mesh base station and the users in a grid can directly communicate with only the base station located in the same grid. In addition, a base station can directly communicate with only the base stations in the neighboring grids sharing at least one side or one corner with the grid of the base station. All nodes are aware of their locations.

The mobility of the client nodes in our system is assumed to follow the Random Waypoint Model (RWM) [19]. According to RWM, a mobility process of a node is started by the node holding on a position for a certain period of time, or called a pause time. Once the time expires, the node randomly selects a destination in a specific candidate area surrounding it and a speed that is uniformly distributed between [minspeed, maxspeed]. The node will then move towards the new destination at the selected speed. Upon arrival, the mobile node starts a new mobility process again. In a real system, the candidate area will be determined by the applications and the practical environments.

B. *Events and Filters in Our System*

Similar to the existing content-based Pub/Sub system [2]-[4], an event in our system is described by a tuple of Attribute-Value, while a subscription filter is described by a tuple of Attribute-Value-Operator. For example, event A <stock price, 10.9> means that the stock price is $10.9 and filter A' < stock price, 10, '>'> means that the subscribers need to receive the event where the stock price is higher than $10. We say that an event can match a filter when the topic of the filter and the event are the same and the comparison between the values of the filter and the event satisfies the "operator" in the filter [2]. Obviously, event A can match filter A' in the above example.

III. Overview of our Pub/Sub System for WMNs

This section provides an overview of our system. We first briefly describe the broker/client system architecture and then describe the event subscription and publication in the system.

Our Pub/Sub system uses the broker/client architecture, where brokers perform the filter/event transmission and management. The clients in the system subscribe or unsubscribe their interests to the system by just registering the corresponding subscription or un-subscription to their brokers. Every mesh base station is a broker and every mobile user is a client. The functions of the broker and client are implemented by running corresponding programs on the base stations and user devices, respectively. Every broker and client has a unique system-wide ID. A broker maintains an event sequence number, which is incremented by 1 for every newly generated event. This new event can be uniquely identified in the system by its ID, which is composed of local sequence number of the event and the broker's ID.

Similar to most of the existing Pub/Sub systems [2]-[4], all brokers in our system are organized into an acyclic undirected graph (AUG), where every broker use the reverse-path-forwarding protocol to receive the events subscribed by its clients. A broker will propagate the subscriptions received from its clients in the AUG to all other brokers, establishing a data collection tree rooted at this broker. The events matching the subscription will be reversely forwarded along the tree until they reach the root, i.e., the subscribing broker. The subscribing broker will further push these events to the corresponding clients.

IV. Our Protocol Based on Mobility Predication

In this section, we propose an event delivery protocol for mobile clients with low transmission overhead and delay. Using the mobility prediction based on RWM and mobile statuses of clients, a broker can determine how to deliver the received events to a runaway client previously served by this broker.

A. *Mobile Status Update*

In our system, a mobile client can continuously move in different grids of the deployment region of the WMN. The client needs to report its mobility status to a broker. After receiving the report from the client, the broker will become the new serving broker of the client and delivery the event to the client thereafter.

Using RWM, the mobility status of a client can be expressed as a 5-tuple, <C, D, V, P, R>, where C is the current position, D is the position of the next destination, V is the velocity, P is the pause time, and R is the remaining pause time, respectively. When a new client wants to participate in our system or when a client changes its mobility status, it will report its mobility status together with its ID and its subscriptions to the broker in its current grid, which will become its new serving broker.

B. *Event Delivery Based on Predication*

In our system, a broker maintains a client table, where each entry records a client's information, including the client's ID, mobility status, and subscriptions. After receiving an event subscribed by one of its mobile clients, if a broker can directly communicate with the client, it will deliver the event to the client; otherwise, the broker will use Algorithm 1 (shown in

Figure 2) to determine a forwarding broker that can directly forward the event to the client. The broker will then use the geographical forwarding protocol [20] to deliver the event to the forwarding broker via multi-hop relay among brokers. When an intermediate broker receives the event, it will also geographically forward the event until the event is received by the forwarding broker. The forwarding broker will then directly forward the received event to the moving client.

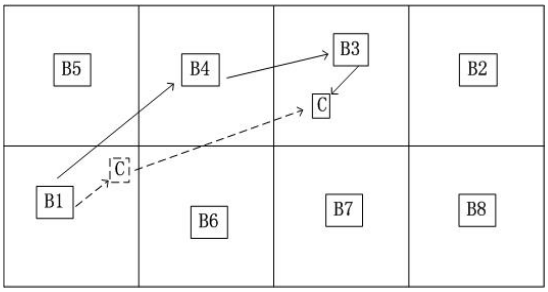


Figure 1. the moving of a client

Take the example shown in Figure 1. Client C is moving from the grid of broker B1 to the grid of broker B2. After T' time, B1, the serving broker of C, receives an event subscribed by C. B1 determines B3 to be the forwarding broker of C by using Algorithm 1, and then use geographical forwarding to deliver the event to B3 via an intermediate broker B4. After receiving the event, B3 will deliver the event to C.

In Algorithm 1, a broker first uses the mobility status information of a runaway client to predict the moving path of the client. The broker will sequentially search the forwarding broker from the grids passed by the moving path. In our algorithm, both Line 2 and Line 6.2 have the time complexity of the length of the client's moving path. According to RWM, the length of the client's moving path is O($\sqrt{A}$), where *A* is the area of deploying region of the WMN. Thus, the time complexity of our algorithm is also O($\sqrt{A}$).

C. Handover between New and Old Serving Brokers

When a mobile client employs a new serving broker, the new one propagates the subscriptions submitted by the client in the Pub/Sub system. Meanwhile, the old serving broker still forwards the events for the client to the new serving broker. Since every event has a unique ID, the new serving broker can discard the replicated events so the client will receive every subscribed event *exactly once*. After certain duration, the subscription propagation will be finished and this broker will inform the old broker to stop the event forwarding. After being informed, the old broker can delete the entry of the mobile client from its client table. If there is no any other client subscribing the same events, the old serving broker will also perform the corresponding unsubscribing.

V. PERFORMANCE EVALUATION

We have evaluated the performance of our protocol. In this section, we first theoretically analyze the performance of our protocol based on RWM. Then, we describe the simulation study and report the results measuring the performance of our protocol in comparison with the existing protocols.

```
INPUT:
<D, C, V, P, R>  // the mobility status of a client
Delay            // the transmission delay between two
                    neighboring brokers
T                // the duration from the last updating
                    of mobility status to the current time
OUTPUT:
GID             //  the ID of the forwarding broker which the
                    event should be sent to
1.     Dist C,D ← the distance between D and C;
2.     G_Array ← the IDs of grids passed by Path CD
                 sequentially from C to D;
               // the array kept all grids passed by Path CD
                  and the index starts from 0.
3.     G_No ← the number of grids passed by Path CD;
4.     Cur_GIND ← the index of the grid where the client
                  stays currently;
5.     D_GID ← the ID of the grid where D is;
6.     if ((Dist C,D / V) + R) + P < T
6.1.   then  GID←D_GID;
6.2.   else  for i = Cur_GIND to G_No do
                 T' ← index * Delay;
                   // the transmission delay from this broker
                      to the broker in G_Array[i];
                 pos' ← the position of the mobile client at
                        future T' time;
                 if  pos' is in the the grid whose index is i
                 then GID← G_Array[i];
                       Break;
                 endif
             endfor
         endif
```

Figure 2. Algorithm 1

A. Theoretical Analysis

In our theoretical analysis and experimental evaluation, we assume that the event arrival follows the Poisson model [21], where the interval of time between the arrivals of two sequential events obeys the exponential distribution. The probability density function (PDF) of exponential distribution is $f(x) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0 \\ 0, & x < 0 \end{cases}$, where λ is the expectation of the interval. $1/\lambda$ is the arrival rate of the events. For example, λ=8 means that expected number of events arriving in 8 seconds is 1.

We assume that a mobile client can randomly choose a destination in a square region composed of $(2k+1)*(2k+1)$ grids, where k is a positive integer, and the grid where the client starts off is the central grid of the region. See Figure 3 for an example. We define one hop to be a move from a grid to any of the neighboring grid. Obviously, the number of grids which is i hops away from the central grid are $8i$, where i =1, 2, 3, …, k. In addition, given i, these grids that is i hops away from the central grid consist of a square circle, as the green or yellow grids in Figure 3. Thus, the expected number of grids passed by a client moving between two sequential destinations is

$$E(\text{grids}) = (\sum_{i=1}^{k} 8*i*i+1*0)/\sum_{i=1}^{k} 8i \qquad (1)$$

Simplifying (1), we can have

$$E(\text{grids}) = \frac{1}{3}(2k+1), \qquad (2)$$

i.e. the expected hops between two sequential destinations is (2k+1)/3.

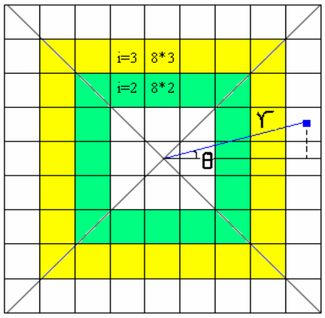


Figure 3 square circles in the grids

Assuming the starting position of a mobile client is the center of the central grid (Figure 3), the expected distance from the starting point to any position in a square circle is given by

$$E(dist) = \int_{square\ circle} r\sec\theta ds \Big/ (8*i*l^2), \qquad (3)$$

where l is the lateral length of one grid and ds, the integral element, can be expressed as $rdrd\theta$. Following (5), we have

$$E(dist) = \frac{1}{il^2}\int_{\frac{2i-1}{2}l}^{\frac{2i+1}{2}l}\int_0^{\frac{\pi}{4}} r\sec\theta rdrd\theta = il\ln(\sqrt{2}+1). \qquad (4)$$

In RWM, a client keeps a stable velocity, denoted as v, during a mobility process. Thus, the expected time spent by a client from the starting position to any position of the square circle can be calculated as follows.

$$E(t) = \frac{il(\sqrt{2}+1)}{v}. \qquad (5)$$

According to (5), we can find that the mobility time of a client from one destination to another is proportional to the expected number of grids passed by this client.

We define the transmission overhead per event as the number of packets transmitted by the brokers for delivering the event using geographically forwarding. The overhead will be incremented by 1 when one packet has been transmitted over 1 hop. Then, the expected overhead of our protocol for the event delivery during the movement of a client from its current position to the next destination can be calculated as follows.

$$E(overhead) = E\left(\frac{\int_0^T (1/\lambda)*i(t)*dt}{T/\lambda}\right) = E\left(\frac{1}{2}i(T)\right) = \frac{1}{2}E(i(T)) = \frac{1}{2}E(grids) = \frac{2*k+1}{6} \qquad (6)$$

where $i(t)$ is the number of grids passed by a client in a time period t and $i(t)$ in direct to the time t of mobility according to expression (5). More specifically, $i(T)$ is the expected number of grids passed by a client from the starting position to the next destination, which equals to $E(grids)$ defined in (2).

According to (6), the expected overhead per event of our protocol is only determined by the area of the region, but not affected by the client's speed and event arrival rate. For example, when k=10, the overhead is 3.5 and when k=20, the overhead is 6.8. Our simulation results presented later validates this our theoretical analysis. Also, the transmission delay per event of our protocols is not affected by the client's speed and event arrival rate, either.

B. *Experimental Evaluation*

We have carried out simulations to measure the performance of our protocol. In our simulation, we used two values, 4 and 8, for λ, which means that expected interval between two continuously arriving events is 4 and 8 seconds, respectively. We used two values of 10 and 20, for k. Thus, the number of grids in the deploying region of WMN is 21*21 and 41*41, respectively. The size of one grid is 250m*250m.We use 3, 6, 9, 12, and 15 m/s as the velocities of a client. For every combination of the test parameters, we carried out 10,000 runs of the simulation and the results shown here are the average values.

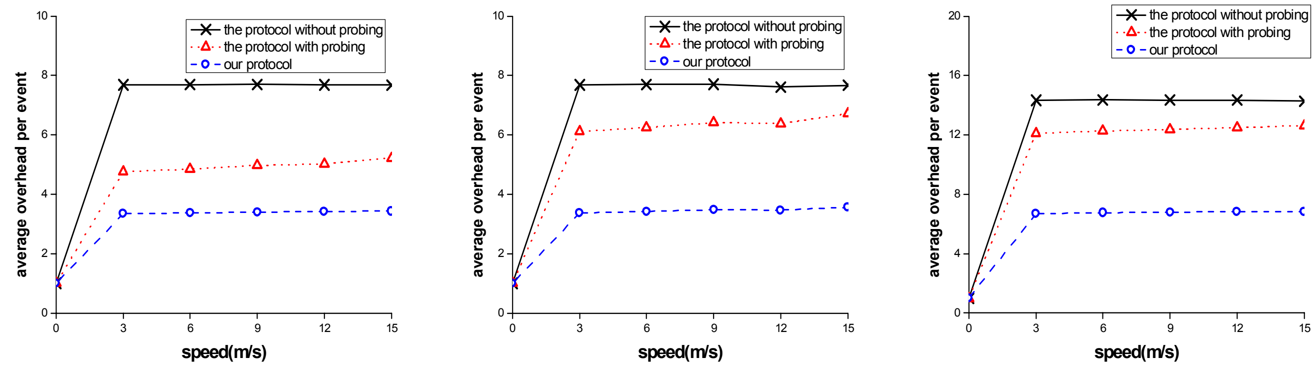
We also simulated other two protocols in [5] [8]-[10] for performance comparison. One protocol is based on the periodic probing method where a mobile client periodically probes its current serving broker and the serving broker replies the probing with the newly received events subscribed by clients. We set the probing interval to be 10 seconds. Another protocol is a batch forwarding method. Using this method, the old serving broker of the mobile client forwards all the events received during the moving of the mobile client to the new broker after the client reaches the destination.

The following two performance metrics have been used.

1) *Average overhead per event delivery* is the total number of transmitted packets from the serving broker to the client for event delivery using our protocol and the protocol without probing. As for the protocol with probing, the cost of transmitting probing packet will also be included in the overhead. We assume that one packet can contain only one probing. The overhead will be increase by 1 when one packet is transmitted over 1 hop.

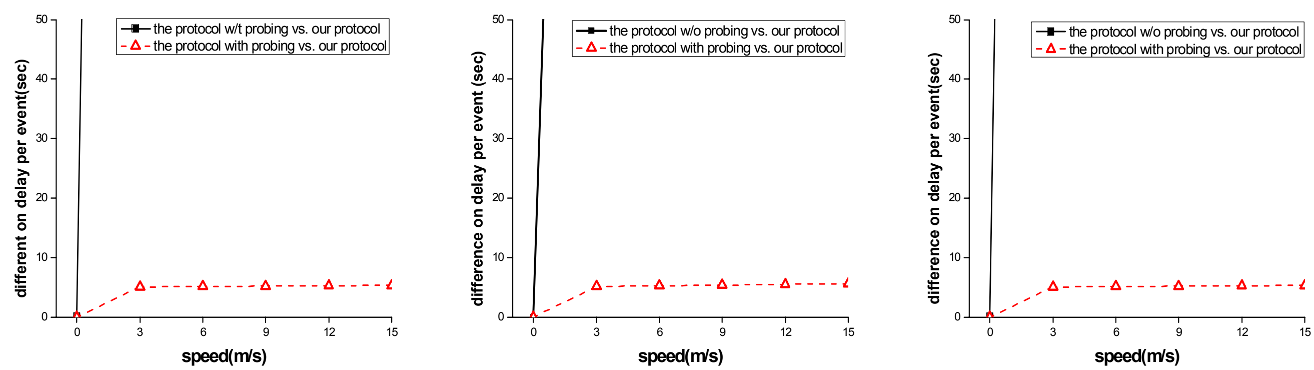
2) *Difference on delay per event delivery* is the difference on transmission delay between our protocol and both of the other protocols. The transmission delay per event delivery is the duration from the time when event received by the serving broker to the time when it is received by client, including waiting time of events in the cache of the broker, link delay, broker processing delay, etc. In fact, we believe that the major delay in a WMN is the waiting time of events in the cache of a broker. Thus, this metric mainly shows the waiting time of events in brokers when we use the protocol with or without probing.

The simulation results are shown in Figure 4 and Figure 5. From both figures, we can see that our protocol can obtain the best performance on both metrics: overhead and delay. Figure 4 shows the average overhead per event delivery under the combinations of the different event arrival rates and different areas of the region. The overhead of our protocol can be much lower than the protocol without probing. Comparing to the protocol with probing, our protocol achieve 1/3 and 1/2 lower



(a) λ=4, k=10 (b) λ=8, k=10 (c) λ=8, k=20

Figure 4. Comparison of average overhead per event between our protocol and the other ones with different areas of the region and different event arrival rates



(a) λ=4, k=10 (b) λ=8, k=10 (c) λ=8, k=20

Figure 5. Difference on delay per event between our protocol and the other ones with different event arrival rates and different areas of the region

overhead when k=10 and k=20, respectively. Since the average number of hops in every transmission is doubled along with the increase of k, the transmission overhead for every event is also doubled. Meanwhile, the increase in speed does not affect the transmission overhead of our protocol. These results validated our theoretical analysis.

Figure 5 shows the difference on delay per event delivery between our protocol and of the other two protocols with different areas of the region and different event arrival rates. For the delay, our protocol significantly outperforms the protocol without probing and is about 5 seconds lower than the protocol with probing for every event when λ=4 and λ=8 in the region of different areas. The Difference of delay per event delivery between our protocol and any of the other protocols keeps stable along with the rising of the event arrival rate.

VI. RELATED WORK

In recent years, many works have been done on Pub/Sub middleware for the traditional distributed computing systems, such as SIENA [2], CEA [3], and Gryphon [4]. However, these systems have not been concerned with the problem on how to support mobile clients.

L. Fiege et al. [5] have adapted the Rebeca [6] system to support mobile clients by employing a handover protocol. Using the protocol, a broker buffers the events for a client which has already moved away. Once the mobile client re-connects to the system via another broker, the system send it the events buffered in the original broker, which can be found according to the subscriptions and event sequence number re-submitted by the client.

The extension for supporting mobile client on Elvin [7] system has been proposed by P. Sutton et al. [8]. The main idea is to employ a central proxy that tackles subscriptions for disconnected clients. After reconnecting to the system, a mobile client will first connect to the central proxy to obtain the events published during their leaving. The central proxy, however, tends to become a performance bottleneck and the system does not have good scalability.

A 2-phase handover protocol has been proposed in [9]. The operations supporting a mobile client include handover request phase and handover response phase. The authors also proposed the approaches to handle concurrent handover and deadlock in handover caused by the mobility of multiply clients.

A persistent notification protocol has been proposed in [10] to support mobile clients in Pub/Sub systems. In this approach, every broker keeps a list for the IDs of the events published by itself and buffers the published events according to their lifetime. When a mobile client connects a new broker, the client will submit the IDs of latest events received by it to the new broker. The new broker will search the new events for the client in the whole system.

Two handover protocols have also been proposed in [12][13], respectively, to support mobile clients in pub/sub systems but the authors have not given enough details. However, the above four protocols have not considered the mobility properties of the clients, such as, possible destinations, mobility speed, etc.

Burcea, I., H.-A. Jac, et al. [11] have proposed a simple handover protocol based on predicting the destination of mobile clients. However, their work has ignored the problem of continuous event delivery for a client during its moving and this problem is more challenging than the client's simply movein/moveout problem solved in their work.

The mobility of publishers have been concerned by H.-A. Jac in [14]. The authors proposed four solutions to alleviate the impact of publisher's mobility on the performance of the pub/sub system. However, their work on the publishers' mobility is quite different from ours on the subscriber's mobility.

In addition, there are some studies [15]-[17] concerning the reliability and fault-tolerance in event transmission of mobile clients in Pub/Sub systems. The mainly contributions of these studies are using the redundant event transmission and the mechanism of event sequence number to improve the reliability and fault-tolerance of event transmission in a highly dynamic mobile networks.

VII. Conclusion and Future Work

In this paper, we studied the problem of how to support mobile clients in Pub/Sub systems for WMNs. We proposed the first event delivery protocol for mobile clients using geographical routing and the mobility prediction. The protocol achieves low transmission overhead and delay. Our theoretical analysis shows that the performance of our protocol will not be affected by clients' speeds and the arrival rate of events. This analysis has been validated by our simulation results which also show that our protocol can significantly improve the performance of event transmission in Pub/Sub systems for WMNs compared with the previous solutions designed for other kinds of networks.

We are now working on how to model and evaluate the impact of the imprecision of the mobility prediction on the performance of our protocol. We are also going to alleviate the effect of the imprecise prediction by designing new algorithms and protocols.

Acknowledgment

This research is partially supported by Hong Kong University Research Grant Council CERG Grant PolyU 5103/06E.

References

[1] I.F. Akyildiz, X. Wang, and W. Wang. *Wireless Mesh Networks: A Survey*. In Elsevier Computer Networks Journal, Vol. 47(4):445-487, 2005.

[2] A. Carzaniga, D. S. Rosenblum, A. L. Wolf. *Design and evaluation of a wide-area event notification service*. ACM Transactions on Computer Systems (TOCS) Vol.19(3): 332-383, 2001.

[3] G. Banavar, G., T. Chandra, B. Mukherjee, J. Nagarajarao, R.E. Strom, D. C. Sturman. *An efficient multicast protocol for content-based publish-subscribe systems*. In IEEE International Conference on Distributed Computing Systems (ICDCS), 1999.

[4] J. Bacon, K. Moody, J. Bates, R. Hayton, C. Ma, A. McNeil, O. Seidel, M. Spiteri. *Generic support for distributed applications*. In IEEE Computers Vol. 33(3): 68-76, 2000.

[5] L. Fiege, F. C. Gartner, O. Kasten, and A. Zeidler. *Supporting mobility in content-based publish/subscribe middleware*. In ACM/IFIP/USENIX International Middleware Conference (Middleware), 2003.

[6] G. Muhl. *Large-Scale Content-Based Publish/Subscribe Systems*. PhD thesis. Darmstadt University of Technology, Germany. 2002.

[7] B. Segall, D. Arnold, J. Boot, M. Henderson, and T. Phelps. *Content based routing with elvin4*. In AUUG2K conference. 2000.

[8] CP. Sutton, R. Arkins, and B. Segall. *Supporting disconnectedness – Transparent information delivery for mobile and invisible computing*. In IEEE International Symposium on Cluster Computing and the Grid (CCGrid), 2001.

[9] J. Wang, J. Cao, and J. Li. *Support Mobile Clients in Publish/Subscribe Systems*. In IEEE International Workshop on Mobility in Peer-to-Peer Systems (MPPS), 2005.

[10] I. Podnar and I. Lovrek. *Supporting Mobility with Persistent Notifications in Publish/Subscribe Systems*. In International Workshop on Distributed Event-Based Systems (DEBS), 2004.

[11] I. Burcea, H.-A. Jacobsen, E. DeLara, V. Muthusam,and M. Petrovic. *Disconnected Operation in Publish/Subscribe Middleware*. In IEEE International Conference on Mobile Data Management (MDM), 2004

[12] M. Ionescu and I. Marsic. *Stateful Publish-Subscribe for Mobile Environments*. In ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots (WMASH), 2004.

[13] S. Tarkoma, J. Kangasharju and K. Raatikainen. *Client Mobility in Rendezvous-Notify*. In International Workshop on Distributed Event-Based Systems (DEBS), 2003.

[14] V. Muthusamy, M. Petrovic, H. Jacobsen. *Effects of routing computations in content-based routing networks with mobile data sources*. In ACM International Conference on Mobile Computing and Networking 2005.

[15] M. Petrovic, M. Vinod, H.-A. Jacobsen. *Content-based routing in mobile ad hoc networks*. In IEEE Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous), 2005.

[16] R. Baldoni, R. Beraldi, G. Cugola, M. Migliavacca, L. Querzoni. *Structure-less content-based routing in mobile ad hoc networks*. In IEEE International Conference on Pervasive Services (ICPS), 2005.

[17] P. Costa and G. Picco. *Semi-probabilistic Content-based Publish/subscribe*. In IEEE International Conference on Distributed Computing (ICDCS), 2005.

[18] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice*, 4th ed., Springer Verlag, 1997.

[19] T. Camp, J. Boleng, and V. Davies, *A Survey of Mobility Models for Ad Hoc Network Research*. In Wireless Communication & Mobile Computing (WCMC) 2(5):483-502, 2002.

[20] B. Karp, H.T. Kung. *GPSR: greedy perimeter stateless routing for wireless sensor networks*. In ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom), 2000.

[21] R. Nelson. *Probability, Stochasitic processes, and queueing theory*. 2nd ed., Springer Verlag, 1998.