This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2008 proceedings.

Evaluation of RSVP and Mobility-aware RSVP Using Performance Evaluation Process Algebra

Hao Wang and David I. Laurenson
Institute for Digital Communications,
Joint Research Institute for Signal & Image Processing,
School of Engineering & Electronics,
The University of Edinburgh
Email: {H.Wang, Dave.Laurenson}@ed.ac.uk

Jane Hillston
Laboratory for Foundations of Computer Science,
School of Informatics,
The University of Edinburgh
Email: Jane.Hillston@ed.ac.uk

***Abstract*—As a resource reservation mechanism, the Resource ReSerVation Protocol (RSVP) faces a lot of challenges when applying it to the wireless and mobile networks. The interworking problems of RSVP and mobility management protocols have been extensively discussed over the last decade. As the solutions of this problem, mobility-aware RSVP schemes that integrate RSVP and micro-mobility management are becoming more and more popular. Therefore, the investigation on how much they improve the performance of the basic RSVP is necessary and useful. Instead of the traditional simulation based approaches, in this paper we introduce a formal performance evaluation formalism, named Performance Evaluation Process Algebra (PEPA), and employ it to investigate the performance of the basic RSVP and mobility-aware RSVP. Important performance metrics such as handover blocking probability and signalling cost are presented.**

I. INTRODUCTION

As many real-time services and multimedia applications become popular, providing guaranteed quality of service (QoS) to Internet users is an important issue for the next generation of traffic management. One of the proposed solutions is the Integrated Service [1] that utilises a signalling protocol such as Resource ReSerVation Protocol (RSVP) [2] to control end-to-end packet delay. However, due to the mobility of mobile users, RSVP becomes inefficient because there is a disruption of QoS traffic when a mobile node changes its point of attachment to the network. A lot of variants of the basic RSVP have been proposed and most of them tackle the problem from the perspective of the macro-mobility management protocols such as Mobile IP. Detailed surveys of RSVP over Mobile IP can be found in [3]–[5]. On the other hand, it is proposed in [6] that for every mobile node's movement to a new IP subnet, the micro-mobility management protocol is preferable to its global counterpart and a global-mobility management protocol is not even strictly required to provide node mobility. Moreover, as we will see in section II, a micro-mobility management protocol such as Hierarchical Mobile IPv6 (HMIPv6) [7] has inherent characteristics which facilitate the deployment of RSVP in a mobile environment. Therefore, schemes that integrate RSVP and micro-mobility management mechanisms have become widely accepted as the best approach to combining mobility and QoS, and it is necessary and useful to investigate their expected performance.

Most of the previous efforts on evaluating the enhancements of the mobility-aware RSVP are carried out by simulation. Specific network topologies and traffic scenarios are used in the simulations and performance metrics such as packet delay and throughput are obtained. However, simulation is not always a reliable means of determining performance metrics since the results are usually subject to the specific simulation setup. The contribution of our work is that we are the first to build Markovian models of both basic RSVP and mobility-aware RSVP to assess their performance. Furthermore, these models are built using a formal performance evaluation formalism, named Performance Evaluation Process Algebra (PEPA). From these PEPA models, we derive important performance metrics such as handover blocking probability and signalling cost, and demonstrate the advantages of the mobility-aware RSVP. Moreover, we should point out that our models are independent of the specific implementations of RSVP and mobility-aware RSVP schemes and capture the essential characteristics underlying them.

The rest of the paper is structured as follows. In Section II we introduce the basic RSVP and the mobility-aware RSVP schemes that integrate RSVP and micro-mobility management mechanisms. In Section III we give a short introduction to the PEPA formalism. In Section IV the PEPA models of both basic and mobility-aware RSVP are presented. The performance of the two RSVP schemes are analysed in Section V and we give our conclusion in Section VI.

II. RSVP AND MOBILITY-AWARE RSVP

RSVP is a receiver-oriented resource reservation setup protocol for simplex data flows. It can be used by a host to request specific qualities of service from the network and by routers to establish and maintain the required QoS. Since in RSVP the resource reservation in a network is identified by the IP addresses of the communicating ends, one of the major incompatibilities between RSVP and mobility management when providing QoS guarantees in a mobile network is that the receivers must re-establish reservation whenever a mobile node performs a handover. This disruption during handover significantly degrades QoS-sensitive services. To reduce the resource re-establishment time, one of the solutions is to

978-1-4244-2075-9/08/$25.00 ©2008 IEEE

localise the reservation signalling within the affected part of the path in the network [8].

Previous work on deploying RSVP in a micro-mobility management enabled network [9]–[11] takes advantage of the two-layer care-of addresses of a mobile node. Here we take HMIPv6 as the example. In HMIPv6, there is a new mobility agent called Mobility Anchor Point (MAP) that covers a group of access routers (ARs). Every time a mobile node moves into a MAP domain, it acquires an on-link care-of address (LCoA) referring to the AR which it is connected to and a regional care-of address (RCoA) referring to the MAP. Outside the MAP domain, the mobile node is identified by its RCoA and all the packets addressed at RCoA are intercepted by the MAP. The MAP will then forward these packets to the mobile node at LCoA. Therefore, when the mobile node performs a handover within a MAP domain, i.e., switches to a new AR connecting to the same MAP, only the LCoA is changed and the RCoA remains the same. It then follows that a mobile node actually only needs to change the reservation path between the AR and the MAP, and maintains the same reservation path outside the MAP domain as long as it uses the same RCoA. In the proposed mobility-aware RSVP schemes, there is an agent (Mobility Proxy in [9] and QoS Agent in [10]) in the access network that assists the mobile node to make this kind of partial resource reservation. This agent can be located at the gateway of the access network. Every time the mobile node performs a handover, it notifies the agent of the current binding between its LCoA and RCoA. Upon receiving this information, the agent is capable of intercepting and looking into the RSVP messages and swapping the LCoA and RCoA in a way that the reservation below and above the agent is identified by the LCoA and the RCoA respectively. Therefore, as long as the mobile node moves within the same MAP domain, the RSVP signalling only traverses to the agent and the reservation re-establishment time is reduced. For details about the operation of these schemes, see [9], [10].

III. PERFORMANCE EVALUATION PROCESS ALGEBRA

The term *process algebras* refers to mathematical theories which model and reason about the structure and behaviour of a system in an algebraic framework. Performance Evaluation Process Algebra (PEPA) [12] is a timed and stochastic extension of classical process algebra such as CCS [13] and CPS [14] that can be used for performance modelling of computer and communication systems. PEPA is a compositional approach that decomposes the system into subsystems that are smaller and more easily modelled. In PEPA a system is usually composed of a group of *components* that engage in *activities*. Generally, components model the physical or logical elements of a system and activities characterise the behaviour of these components. Each activity a in PEPA is defined as a pair (α, r) — the action type α, which can be regarded as the name of the activity, and the activity rate r, which is an exponentially distributed random variable and specifies the duration of the activity. If a component P behaves as P' after completing activity a, then we can regard this behaviour as a component changing from state P to state P', through transition (α, r).

The PEPA formalism provides a small set of operators which are able to express the individual activities of components as well as the interactions between them. We only present the operators we used in our model in this section. For more details about PEPA operators, see [12].

Prefix: $(\alpha, r).P$

This component has a designated first activity which is of action type (or name) α and has a duration that is exponentially distributed with rate r, which gives a mean time of $1/r$. A larger rate implies a faster completion of an activity. After completing this activity, the component $(\alpha, r).P$ behaves as P.

Choice: $P + Q$

This component may either behave as P or Q. All the enabled activities in P and Q are enabled in this component and compete with each other. The first activity to be completed will be an activity of P or Q and this will distinguish which component wins the race. When the first activity is completed, all the other activities will be abandoned. For example, the component $(\alpha, r_1).P' + (\beta, r_2).Q'$ is more likely to subsequently behave as P' if r_1 is larger than r_2.

Cooperation: $P \underset{L}{\bowtie} Q$

This component represents the interaction between P and Q. The set L is called the *cooperation set* and denotes a set of action types that must be carried out by P and Q together. For all activities whose action type is included in set L, P and Q must cooperate to complete it. However, other activities of P and Q which have types that are not included in set L will proceed independently. The rate of the *shared* activity is determined by the rate of the slower participant and is the smaller of the two rates. In PEPA an activity can have an unspecified rate making it a $passive\ activity$ and its rate is labelled as $\top$. This means that although the component which has this passive activity is required to engage in the cooperation, it has no influence on the rate at all.

Parallel: $P \| Q$

This component represents two concurrent but completely independent components. This is simply a shorthand notation for $P \underset{\emptyset}{\bowtie} Q$.

Constant: $P \stackrel{\mathit{def}}{=} Q$

This expression is used to assign names to components.

System Definition:

Since PEPA is a compositional approach, in PEPA a system is described as an interaction of components. The system definition specifies how the system is constructed from the defined components.

To generate a stochastic process which represents the PEPA model, we can associate a state with a component, and the transitions between states are defined by the activities between them. Since the duration of the transition in PEPA is exponentially distributed, it has been shown that the stochastic process underlying a PEPA model is a discrete state space, continuous time Markov chain (CTMC). By deriving the steady state probability vector of the CTMC, and with the help of the Markov reward models (MRMs) [15], performance measures such as utilisation and throughput can be derived. These measures can facilitate model verification and system optimisation.

IV. PEPA MODELS OF BASIC RSVP AND MOBILITY-AWARE RSVP

In this work, we build CTMC-based analytical models using PEPA. PEPA is chosen because firstly its component structure directly reflects the system structure, thereby providing a clear description of the system it models. Secondly, since PEPA is a process algebra language, it is quicker and easier to construct models than working directly at the state space level. Thirdly, since PEPA models can be solved numerically, some restrictions, which other modelling approaches such as queueing networks must follow to exhibit a product form solution, do not constrain PEPA models. Last but not least, sophisticated tools [16] have been developed which make both steady state and transient analysis of PEPA models convenient. In this section, the PEPA models of the basic RSVP and the mobility-aware RSVP are presented. We remind the reader that our models are not restricted to any specific implementation of these schemes but are general models which capture the essential characteristics of basic RSVP and mobility-aware RSVP.

A. PEPA Model of Basic RSVP

The scenario used in our model is a mobile node moving within a local domain and communicating with other nodes. We refer to the networks below and above the merge point of the old and new RSVP path as the *lower network* and the *upper network* respectively. The lower network consists of the whole or part of the mobile node's access network and the upper network consists of the Internet core network which is usually heavily loaded. Here we borrow the concept of *channel* from cellular networks to represent the network resources. Therefore, there are three elementary types of PEPA components in the model, which are *Mobile Node* (*MN*), *Lower Network Channel* (*LNC*) and *Upper Network Channel* (*UNC*). The last two components represent the resources in the lower network and upper network respectively.

Mobile Node: The *MN* models the behaviour of a mobile node. The *MN* is initially in the idle state MN_0. It requests a reservation of both *LNC* and *UNC* (state MN_1) after receiving a call request which arrives at the rate of λ. If both *LNC* and *UNC* are available, the request is accepted and the *MN* can start its RSVP session (state MN_2). Otherwise, the request is blocked and the *MN* keeps requesting a reservation until it is finally allocated one (state MN_1). The average length of an RSVP session is assumed to be $1/\mu$. During this session, the *MN* can perform a localised handover at the rate of α, and then it needs to request a new reservation of both *LNC* and *UNC* in order to continue its session (state MN_3). (We assume the *MN* implements the *local repair* [2] option, so it can request a new reservation almost immediately after a handover.) After the session is finished, the *MN* tears down its current reservation (state MN_4). The component *MN* is defined as:

$$\begin{aligned} MN_0 &\overset{\mathit{def}}{=} (call\_arrive, \lambda).MN_1 \\ MN_1 &\overset{\mathit{def}}{=} (reserve\_all, r).MN_2 \\ &\quad + (block, b).MN_1 \\ MN_2 &\overset{\mathit{def}}{=} (session, \mu).MN_4 \\ &\quad + (handover, \alpha).MN_3 \\ MN_3 &\overset{\mathit{def}}{=} (reserve\_all, r).MN_2 \\ &\quad + (block, b).MN_3 \\ MN_4 &\overset{\mathit{def}}{=} (tear\_all, t).MN_0 \end{aligned}$$

Lower Network Channel: The *LNC* component models the resources in the lower network. It can be reserved and torn down explicitly by a mobile node in a way similar to a queue. If the mobile node performs a handover, the old reservation of the mobile node expires after an average period of $1/\gamma$. (Note that the basic RSVP [2] only suggests a node explicitly tears down its old reservation at the end of an RSVP session.) When the *LNC* is fully engaged, it blocks the requests of the mobile nodes. If the capacity of the *LNC* is M, it is defined as:

$$\begin{aligned} LNC_0 &\overset{\mathit{def}}{=} (reserve\_all, \top).LNC_1 \\ LNC_i &\overset{\mathit{def}}{=} (reserve\_all, \top).LNC_{i+1} \\ &\quad + (tear\_all, \top).LNC_{i-1} \\ &\quad + (expire, \gamma).LNC_{i-1} \quad (\forall\ i \in [1, M-1]) \\ LNC_M &\overset{\mathit{def}}{=} (block, \top).LNC_M \\ &\quad + (tear\_all, \top).LNC_{M-1} \\ &\quad + (expire, \gamma).LNC_{M-1} \end{aligned}$$

Upper Network Channel: The *UNC* component models the resources in the upper network and its behaviour is the same as *LNC*. If the capacity of the *UNC* is N, it is defined as:

$$\begin{aligned} UNC_0 &\overset{\mathit{def}}{=} (reserve\_all, \top).UNC_1 \\ UNC_i &\overset{\mathit{def}}{=} (reserve\_all, \top).UNC_{i+1} \\ &\quad + (tear\_all, \top).UNC_{i-1} \\ &\quad + (expire, \gamma).UNC_{i-1} \quad (\forall\ i \in [1, N-1]) \\ UNC_N &\overset{\mathit{def}}{=} (block, \top).UNC_N \\ &\quad + (tear\_all, \top).UNC_{N-1} \\ &\quad + (expire, \gamma).UNC_{N-1} \end{aligned}$$

Channel Monitor: The *CM* component is an assistant component in our model. Its function is to guarantee that the *expire* activity is only performed after a *handover* (by requiring a cooperation on it between *CM* , *LNC* and *UNC*) and the number of performed *expire* and *handover* activities are the same. It is defined as:

$$\begin{aligned} CM_0 &\overset{\mathit{def}}{=} (handover, \top).CM_1 \\ CM_i &\overset{\mathit{def}}{=} (handover, \top).CM_{i+1} \\ &\quad + (expire, \top).CM_{i-1} \ (\forall\ i \in [1, M-1]) \\ CM_M &\overset{\mathit{def}}{=} (expire, \top).CM_{M-1} \end{aligned}$$

System Definition: Since in basic RSVP the mobile node reserves and releases resources in both lower and upper network at the same time, the activity $reserve\_all$ and $tear\_all$

must be carried out by *MN*, *LNC* and *UNC* together. Either *LNC* or *UNC* can cooperate with the *MN* on the *block* activity when they are fully engaged. The *CM* synchronises with *MN* on the $handover$ activity and with *LNC* and *UNC* on the $expire$ activity. In this way, the *expire* activity can only be carried out after a *handover*. To guarantee the built model is numerically tractable while keeping the generality, in our model there are 3 parallel mobile nodes and M and N are set to be 5 and 3 respectively. Therefore, the RSVP model is constructed as:

$$System \overset{def}{=} (MN_0 \| MN_0 \| MN_0) \underset{L_1}{\bowtie} \Big((UNC_0 \underset{L_2}{\bowtie} LNC_0) \underset{L_3}{\bowtie} CM_0 \Big)$$

where

$$L_1 = \{reserve\_all, tear\_all, block, handover\},$$

$$L_2 = \{reserve\_all, tear\_all, expire\},\ L_3 = \{expire\}.$$

B. PEPA Model of Mobility-aware RSVP

In the mobility-aware RSVP, the *MN* only requests a new reservation in the lower network after a handover. Since a PEPA component essentially describes the behaviour of an entity, we can simply modify the *MN* component so that when it is in state MN_3 it performs $reserve\_lnc$ instead of $reserve\_all$. The activity $reserve\_lnc$ represents a reservation request for lower network resource only. The component *MN* is modified as:

$$\begin{aligned}
MN_0 &\overset{def}{=} (call\_arrive, \lambda).MN_1 \\
MN_1 &\overset{def}{=} (reserve\_all, r).MN_2 \\
&\quad + (block, b).MN_1 \\
MN_2 &\overset{def}{=} (session, \mu).MN_4 \\
&\quad + (handover, \alpha).MN_3 \\
MN_3 &\overset{def}{=} (reserve\_lnc, r).MN_2 \\
&\quad + (block, b).MN_3 \\
MN_4 &\overset{def}{=} (tear\_all, t).MN_0
\end{aligned}$$

The *LNC* component also needs to be modified so that it is aware of the new type of request which only asks for reservation in the lower network. To make it a fair comparison, in our model the *MN* does not explicitly remove the old reservation after a handover as required in the proposed mobility-aware RSVP schemes. The component *LNC* is modified as:

$$\begin{aligned}
LNC_0 &\overset{def}{=} (reserve\_all, \top).LNC_1 \\
&\quad + (reserve\_lnc, \top).LNC_1 \\
LNC_i &\overset{def}{=} (reserve\_all, \top).LNC_{i+1} \\
&\quad + (reserve\_lnc, \top).LNC_{i+1} \\
&\quad + (tear\_all, \top).LNC_{i-1} \\
&\quad + (expire, \gamma).LNC_{i-1}\ (\forall\ i \in [1, M-1]) \\
LNC_M &\overset{def}{=} (block, \top).LNC_M \\
&\quad + (tear\_all, \top).LNC_{M-1} \\
&\quad + (expire, \gamma).LNC_{M-1}
\end{aligned}$$

Accordingly, since there is no need for a new upper network reservation after a handover, an upper network reservation is established and torn down at the start and the end of an RSVP session. It never expires because it is always active during an RSVP session. The component *UNC* is modified as:

TABLE I
PARAMETERS VALUES

Type (Role)	Average Time (sec.)	Rate (1/sec.)
λ (call arrival interval)	10-100	0.01-0.1
μ (session duration)	180	1/180
α (handover interval)	120-600	1/600-1/120
γ (soft state lifetime)	90	1/90
r (reserve signalling)	0.1	10
b (block signalling)	0.1	10
t (tear signalling)	0.1	10

$$\begin{aligned}
UNC_0 &\overset{def}{=} (reserve\_all, \top).UNC_1 \\
UNC_i &\overset{def}{=} (reserve\_all, \top).UNC_{i+1} \\
&\quad + (tear\_all, \top).UNC_{i-1}\ (\forall\ i \in [1, N-1]) \\
UNC_N &\overset{def}{=} (block, \top).UNC_N \\
&\quad + (tear\_all, \top).UNC_{N-1}
\end{aligned}$$

System Definition: The system definition of mobility-aware RSVP is the same as the basic RSVP model except for the cooperation sets L_1 and L_2. The L_1 now includes the $reserve\_lnc$ activity, and the $expire$ activity is removed from L_2 since the upper network reservation does not expire.

$$System \overset{def}{=} (MN_0 \| MN_0 \| MN_0) \underset{L_1}{\bowtie} \Big((UNC_0 \underset{L_2}{\bowtie} LNC_0) \underset{L_3}{\bowtie} CM_0 \Big)$$

where

$$L_1 = \{reserve\_all, reserve\_lnc, tear\_all, block, handover\},$$

$$L_2 = \{reserve\_all, tear\_all\},\ L_3 = \{expire\}.$$

V. PERFORMANCE EVALUATION

Since the network between the two communicating ends usually consists of the Internet core network where the traffic is highly congested, an optimum utilisation of it is both practically and economically required. A more congested network usually results in a higher handover blocking probability and a larger signalling delay implies a longer interruption of QoS sensitive traffic. The mobility-aware RSVP schemes are especially designed to eliminate the unnecessary consumptions of the network resources and reduce signalling overhead. Therefore, the performance measures we investigate are the probability that the mobile nodes are rejected for continuing their session after handover and the signalling cost of both basic RSVP and mobility-aware RSVP. Before deriving these metrics, we first need to set the activity rates within the model. We make the traditional assumption that the call arrival interval, session duration and handover interval are exponentially distributed. We assume the average lifetime of an RSVP soft state is 90 seconds as suggested in [2]. For the RSVP signalling such as requesting and blocking, they are set to be 0.1 second. These activity rates are shown in Table I.

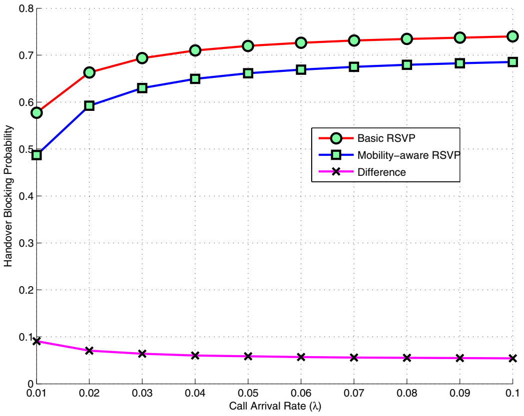

Fig. 1. Handover Blocking Probability vs. Call Arrival Rate

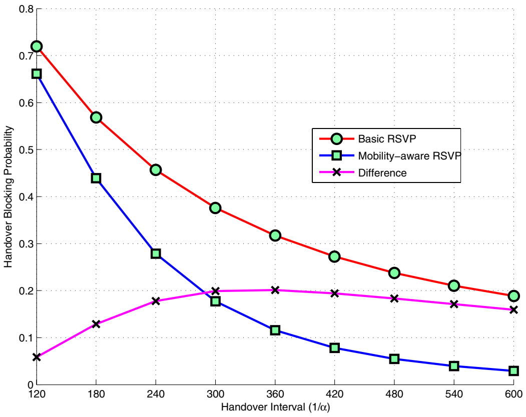

Fig. 2. Handover Blocking Probability vs. Handover Interval

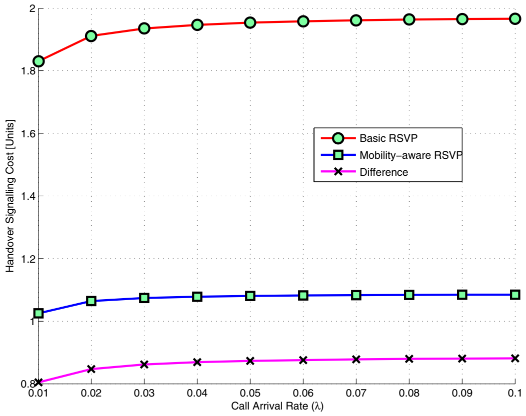

Fig. 3. Handover Signalling Cost vs. Call Arrival Rate

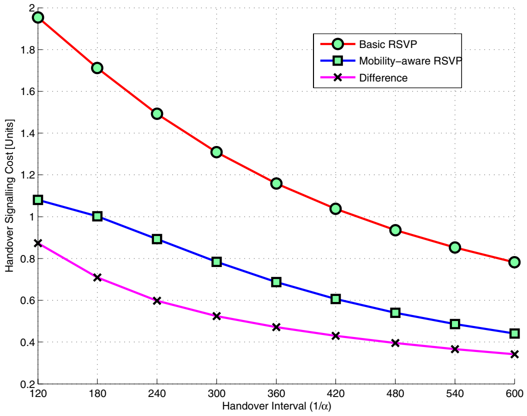

Fig. 4. Handover Signalling Cost vs. Handover Interval

A. *Handover Blocking Probability*

To derive the handover blocking probability, we just need to calculate the probabilities of the states in which the mobile nodes request reservations after handover (*MN* in state MN_3) but the lower or upper network is fully engaged (*LNC* in state LNC_5 or *UNC* in state UNC_3). Fig. 1 shows the effects of the call arrival rate on the handover blocking probability. The handover interval, i.e., the residence time of a mobile node in a subnet, is set to be 120 seconds. It can be observed that the blocking probability increases as expected for both basic RSVP and mobility-aware RSVP and their performance gets closer as the arrival rate of RSVP sessions grows. However, since the mobility-aware RSVP does not require a new reservation in the upper network after a handover, it has a lower blocking probability. We should point out that the reason why the blocking probability is so high is because in our model the network capacity is relatively much smaller than the number of mobile nodes, and we do it particularly to emphasise the congestion of the network and highlight the benefits of the mobility-aware RSVP.

The impact of the mobile node's mobility is also investigated, as shown in Fig. 2. The call arrival rate is set to be 0.05. It is easy to see that when the handover frequency decreases, the reservation requirements for network resources are reduced and thereby a lower handover blocking probability. Moreover, the mobility-aware RSVP has a much lower blocking probability compared to the basic RSVP when the handover interval is around 360 seconds and the difference between them gets smaller when the mobile nodes slow down. We can also observe that the performance of the two schemes gets close at small handover intervals, and this is because the network is overcongested and the mobility-aware RSVP does not improve the performance very much. Therefore, it can be concluded that in most typical scenarios, mobility-aware RSVP is less affected by the mobile node's mobility.

B. *Handover Signalling Cost*

Since one of the major benefits of the mobility-aware RSVP is reducing the scope which the RSVP signalling messages traverse after a handover, another performance measure of interest is the handover signalling cost. By employing the Markov reward model (MRM) [15] on a CTMC, we can easily compute the signalling costs associated with the two schemes. MRMs have been widely used in Markov decision theory to assign rewards (or costs) to states of Markov processes for system optimisation [17]. In our models, we associate rewards with the activities of interest, then the reward associated with a state is calculated by summing up the rewards of the activities that the state enables. If r_i is the reward associated with state S_i, and $\boldsymbol{\pi}(\cdot)$ is the steady state probability vector of the CTMC, then the total reward R is

$$R = \sum_i r_i * \boldsymbol{\pi}(S_i).$$

To derive the handover signalling cost, the activities we investigate are $reserve\_lnc$ and $reserve\_all$ after a handover, i.e. when *MN* is in state MN_3. We assign the costs of one unit and two units to $reserve\_lnc$ and $reserve\_all$ respectively. (That is, we assume the cost of sending a basic RSVP signalling message is two times that of a mobility-aware RSVP signalling message.) The effect of the call arrival rate on the signalling cost is depicted in Fig. 3. The handover interval is set to be 120 seconds. We can find that as the call arrival rate grows, the signalling costs for both RSVP schemes only increases a little at the beginning and then remain almost unchanged. This is mainly because we only take account of the signalling after a handover. Although the mobile nodes generate RSVP sessions more frequently at a larger call arrival rate, the handovers take place during an RSVP session and thus the associated cost is not sensitive to the call arrival rate. Therefore, the handover signalling cost is mostly dependent on the mobility of the mobile nodes, as shown in Fig. 4. The mobility-aware RSVP experiences a lower signalling cost than the basic RSVP since the former restricts the signalling within the affected area of the network. For the large handover intervals, the difference between the two schemes gets smaller because the mobile nodes seldom change their points of attachment and the benefits of the mobility-aware RSVP is less apparent. This again shows that the mobility-aware RSVP is more suitable in a mobile environment.

VI. CONCLUSION

Since RSVP and mobility management protocols were designed independently, the efficient integration of them is necessary to provide a QoS guaranteed mobility to the mobile node. Several mobility-aware RSVP schemes are proposed and it is necessary and practical to investigate how much they improve the basic RSVP. Instead of the traditional simulation based approaches, in this paper, we build Markovian models of both basic and mobility-aware RSVP schemes to evaluate their performance. Moreover, these models are built using a formal performance modelling formalism named PEPA. The PEPA models are built in a general way and so they are independent of the specific implementations of the schemes. Owing to PEPA's component structure, these models exhibit clear representations of the mechanisms underlying the proposed schemes. We investigate the impacts of the call arrival rate and handover interval on the probability of being blocked and the signalling cost after a handover. The results indicate that the mobility-aware RSVP outperforms the basic RSVP on both handover blocking probability and signalling cost as expected. These enhancements are achieved by avoiding unnecessary resource reservation in the unaffected part of the network and limiting RSVP signalling to the lower network. In our future work, we will investigate other problems of combining mobility and QoS, such as efficient resource pre-reservation schemes, in the PEPA framework.

ACKNOWLEDGEMENT

The work reported in this paper has formed part of the Ubiquitous Services Core Research Programme of the Virtual Centre of Excellence in Mobile & Personal Communications, Mobile VCE, www.mobilevce.com. This research has been funded by the Industrial Companies who are Members of Mobile VCE, with additional financial support from the UK Governments Technology Strategy Board (previously DTI). Fully detailed technical reports on this research are available to Industrial Members of Mobile VCE.

J. Hillston is also supported by EPSRC Advanced Research Fellowship EP/c543696/01 and EU FET-IST Global Computing 2 project SENSORIA (Software Engineering for Service-Oriented Overlay Computers (IST-3-016004-IP-09)). H. Wang and D. Laurenson acknowledge the support of the Scottish Funding Council for the Joint Research Institute with the Heriot-Watt University which is a part of the Edinburgh Research Partnership.

REFERENCES

[1] R. Braden, D. Clark, and S. Shenker, "Integrated Services in the Internet Architecture: an Overview," RFC 1633, Jun. 1994.
[2] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification," RFC 2205, Sep. 1997.
[3] B. Moon and H. Aghvami, "RSVP extensions for real-time services in wireless mobile networks," *IEEE Commun. Mag.*, vol. 39, no. 12, pp. 52–59, 2001.
[4] J. Manner, A. L. Toledo, A. Mihailovic, H. L. V. Munoz, E. Hepworth, and Y. Khouaja, "Evaluation of mobility and quality of service interaction," *Comput. Netw.*, vol. 38, no. 2, pp. 137–163, 2002.
[5] S.-J. Leu and R.-S. Chang, "Integrated service mobile Internet: RSVP over mobile IPv4&6," *Mob. Netw. Appl.*, vol. 8, no. 6, pp. 635–642, 2003.
[6] J. Kempf, "Problem Statement for Network-Based Localized Mobility Management (NETLMM)," RFC 4830, Apr. 2007.
[7] H. Soliman, C. Castelluccia, K. E. Malki, and L. Bellier, "Hierarchical Mobile IPv6 Mobility Management (HMIPv6)," RFC 4140, Aug. 2005.
[8] H. Chaskar, "Requirements of a Quality of Service (QoS) Solution for Mobile IP," RFC 3583, Sep. 2003.
[9] S. Paskalis, A. Kaloxylos, E. Zervas, and L. Merakos, "An efficient RSVP-mobile IP interworking scheme," *Mob. Netw. Appl.*, vol. 8, no. 3, pp. 197–207, 2003.
[10] N.-F. Huang and W.-E. Chen, "RSVP extensions for real-time services in hierarchical mobile IPv6," *Mob. Netw. Appl.*, vol. 8, no. 6, pp. 625–634, 2003.
[11] H.-W. Ferng, W.-Y. Kao, J.-J. Huang, and D. Shiung, "A dynamic resource reservation scheme designed for improving multicast protocols in HMIPv6-based networks," in *Proc. Vehicular Technology Conference-Spring '06*, 2006, pp. 961–965.
[12] J. Hillston, *A Compositional Approach to Performance Modelling*. Cambridge University Press, 1996.
[13] R. Milner, *Communication and Concurrency*. Prentice Hall, 1989.
[14] C. A. R. Hoare, *Communicating Sequential Processes*. Prentice Hall, 1985.
[15] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi, *Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications*. John Wiley & Sons, 1998.
[16] PEPA Tools. [Online]. Available: http://www.dcs.ed.ac.uk/pepa/tools/
[17] R. A. Howard, *Dynamic Probabilistic Systems, Volume 2: Semi-Markov and Decision Processes*. John Wiley & Sons, 1971.