QoS-aware Adaptive Services in Mobile Ad-hoc Networks

Baochun Li

Department of Electrical and Computer Engineering
University of Toronto
bli@eecg.toronto.edu

Abstract. Ad-hoc wireless networks consist of mobile nodes interconnected by multi-hop wireless paths. Unlike conventional wireless networks, ad-hoc networks have no fixed network infrastructure or administrative support. Because of the dynamic nature of the network topology and limited bandwidth of wireless channels, Quality-of-Service (QoS) provisioning is an inherently complex and difficult issue. In this paper, we propose a fully distributed and adaptive algorithm to provide statistical QoS guarantees with respect to ***accessibility*** of services in an ad-hoc network. In this algorithm, we focus on the optimization of a new QoS parameter of interest, ***service efficiency***, while keeping protocol overheads to the minimum. To achieve this goal, we first theoretically derive the lower and upper bounds of service efficiency based on a novel model for group mobility, followed by extensive simulation results to verify the effectiveness of our algorithm.

1 Introduction

Wireless ad-hoc networks are self-created and self-organized by a collection of mobile nodes, interconnected by multi-hop wireless paths in a strictly peer-to-peer fashion. Each node may serve as a packet-level router for its peers in the same network. Such networks have recently drawn significant research attention since they offer unique benefits and versatility with respect to bandwidth spatial re-use, intrinsic fault tolerance, and low-cost rapid deployment. Furthermore, near-term commercial availability of Bluetooth-ready wireless interfaces may lead to the actual usage of such networks in reality. However, the topology of ad-hoc networks may be highly dynamic due to unpredictable node mobility, which makes Quality of Service (QoS) provisioning to applications running in such networks inherently hard. The limited bandwidth of wireless channels between nodes further exacerbates the situation, as message exchange overheads of any QoS-provisioning algorithms must be kept at the minimum level. This requires that the algorithms need to be fully distributed to all nodes, rather than centralized to a small subset of nodes.

Previous work on ad-hoc networks has mainly focused on three aspects: general packet routing [1, 2], power-conserving routing [3], and QoS routing [4]. With respect to QoS guarantees, due to the lack of sufficiently accurate knowledge, both instantaneous and predictive, of the network states, even statistical QoS guarantees may be impossible if the nodes are *highly mobile*. In addition, *scalability* with respect to network size becomes an issue, because of the increased computational load and difficulties in propagating network updates within given time bounds. On the other hand, the users of an ad-hoc network may not be satisfied with pure best-effort services, and may demand at least statistical QoS guarantees. Obviously, scalable solutions to such a contradiction have to be based on models which assume that a subset of the network states is sufficiently accurate.

The objective of this work is to provide statistical QoS guarantees with respect to a new QoS parameter in ad-hoc networks, ***service efficiency***, used to quantitatively evaluate the ability of providing the best service coverage with the minimum cost of resources. Our focus is on the generic notion of a *service*, which is defined as a collection of identical *service instances*, each may be a web server or a shared whiteboard. Each instance of service runs in a single mobile node, and is assumed to be critical to applications. Since service instances may be created (by replication) and terminated at run-time, we refer to such a service as an ***adaptive service***.

Since nodes are highly mobile, the ad-hoc network may become partitioned temporarily and subsequently reconnected. In this paper, the subset of nodes is referred to as ***groups***. Based on similar observations, Karumanchi et al. [5] has proposed an update protocol to maximize the availability of the service while incurring reasonable update overheads. However, the groups that it utilized were fixed, pre-determined, and *overlapping* subsets of nodes. Such a group definition may fail to capture the mobility pattern of nodes. Instead, we consider *disjoint sets* of nodes as groups, which are discovered at run-time based on observed mobility patterns. This is preferred in a highly dynamic ad-hoc network. With such group definitions, two critical questions are still not addressed:

– **Group division.** How to divide nodes into groups, so that when the network becomes partitioned, the probability of partitioning along group boundaries is high?
– **Service adaptation.** Assuming the first issue is solved, how can we dynamically create and terminate service instances in each of the groups, so that the *service efficiency* converges to its upper bound with a high probability?

In short, we need an optimal algorithm that maximizes *service efficiency*, i.e., covering the maximum number of nodes with the least possible service instances, especially when the network is partitioned. Obviously, if we assume that node mobility is completely unpredictable, it is impossible to address the issue of group division and service adaptation. We need to have a more constrained and predictable model for node mobility. For this purpose, Hong et al. [6] has proposed a *Reference Point Group Mobility model*, which assumes that nodes are likely to move within *groups*, and that the motion of the *reference point* of each group defines the entire group's motion behavior, including location, speed, direction and acceleration. Since in ad-hoc networks, communications are often within smaller teams which tend to coordinate their movements, the group mobility model is a reasonable assumption in many application scenarios, e.g., emergency rescue teams in a disaster scene or groups of co-workers in a convention. Such a group mobility model was subsequently utilized to derive the *Landmark Routing* protocol [7], which showed its effectiveness to increase scalability and reduce overheads. To further justify the group mobility model, prior research work in the study of the behavioral pattern of wild life [8] has shown extensive grouping behavior in nature, which may be useful as far as ad-hoc sensor networks are concerned.

However, a major drawback of the previously proposed group mobility model was its assumptions that all nodes have prior knowledge of group membership, i.e., they know which group they are in, and that the group membership is *static*. These assumptions have provided an answer to the first unaddressed question, but they are too restrictive and unrealistic. In this work, we relax these assumptions and focus on *dynamic* and *time-varying* group memberships[1] to be detected at run-time by running a distributed algorithm on the nodes, based on *only* local states of each node. With such relaxed assumptions, groups are practically formed and adjusted on-the-fly at run-time, without any prior knowledge about static memberships.

In this paper, our original contributions are the following. First, we provide a mathematical model to rigorously characterize the *group mobility* model using normal probability distributions, based on the intuition proposed in previous work [6]. Second, we define our new QoS parameter of focus, *service efficiency*. Third, based on our definition of group mobility, we theoretically derive lower and upper bounds of the *service efficiency*, which measures the effectiveness of provisioning adaptive services in ad-hoc networks. Fourth, we propose a fully distributed algorithm, referred to as the *adaptive service provisioning algorithm*, to be executed in each of the mobile nodes, so that (1) The group membership of nodes are identified; (2) Service instances are created and terminated dynamically; and (3) message exchange overheads incurred by the algorithm are minimized. Finally, we present our simulation testbed and an extensive collection of simulation results to verify the effectiveness of our distributed algorithm for QoS provisioning.

The remainder of the paper is organized as follows. The mathematical formulation of the group mobility model is given in Section 2. Section 3 shows a theoretical analysis of service efficiency, based on the group mobility model. Section 4 presents the adaptive service provisioning algorithm, in order to identify group membership and manage the adaptive service. Section 5 shows extensive simulation results. Section 6 concludes the paper and discusses future work.

2 Group Mobility Model

The definition of group mobility model given in [6] was intuitive and descriptive, but lacked a theoretical model to rigorously characterize its properties. Furthermore, the model was based on the existence and knowledge of a centralized *reference point* for each group, which characterizes group movements. However, assuming that per-group information such as reference points are known *a priori* to all mobile nodes is unrealistic. For example, when a new node is first introduced to an ad-hoc network, it does not have prior knowledge about the reference points, or even which group it is in.

In this work, we assume that the nodes only have access to its local states, which include its distance to all its neighboring nodes, derived from the physical layer. With this assumption, the group mobility model needs to be redefined so that it is characterized based on fully distributed states, e.g., distances between nodes, rather than the

[1] Strictly speaking, we need to impose some restrictions on the degree of dynamics with respect to group membership changes. It may not exceed the frequency of running the adaptive service provisioning algorithm.

availability of a reference point. Intuitively, nodes within the same group tends to have a high probability of keeping stable distances from each other.

In this paper, we assume that all nodes have identical and fixed transmission range r in the ad-hoc network, and that if the distance between two nodes $\|AB\| \leq r$, they are *in-range nodes* (or *neighboring nodes*) that are able to communicate directly with a *single-hop* wireless link, denoted by $\overline{AB} = 1$, otherwise they are *out-of-range nodes* with $\overline{AB} = 0$. If there exists a multi-hop wireless communication path between A and B interconnected by in-range wireless links, we claim that A and B is mutually *reachable*.

We first define the term *Adjacently Grouped Pair* (AGP) of nodes.

Definition 1 *Nodes A and B form an* ***Adjacently Grouped Pair*** *(AGP), denoted by $A \overset{0}{\sim} B$, if $\|AB\|$ obeys* ***normal distribution*** *with a mean $\mu < r$, and a standard deviation $\sigma < \sigma_{max}$, where $\|AB\|$ denotes the distance between A and B.*

In practice, rather than the absolute value of σ_{max}, one is often interested in the ratio of the standard deviation to the mean of a distribution, commonly referred to as *coefficient of variation* CV_{max}, $(0 < CV_{max} < 1)$, where $\sigma_{max} = r * CV_{max}$.

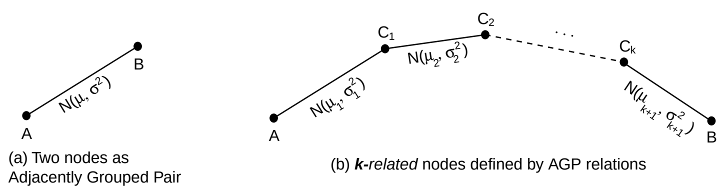


Fig. 1. Using Adjacently Grouped Pairs to form a group

Figure 1(a) shows such a pair of nodes. Intuitively, this definition captures the fact that if two adjacent nodes are in the same group over a period of time, the distance between them stabilizes around a mean value μ with small variations, while $\mu < r$ so that they can communicate wirelessly. Although it is possible that they may be out of range from each other ($\|AB\| > r$) intermittently, the probability is low based on the density function of normal distribution. In addition, σ represents the degree of variations. The mobility patterns of nodes are more similar to each other with a smaller σ.

We now define the term ***k-related*** with Adjacently Grouped Pairs.

Definition 2 *Nodes A and B are* ***k-related****, denoted by $A \overset{k}{\sim} B$ $(k \geq 1)$, if there exist intermediate nodes $C_1, C_2, \ldots, C_k$, such that $A \overset{0}{\sim} C_1, C_1 \overset{0}{\sim} C_2, \ldots, C_i \overset{0}{\sim} C_{i+1}, \ldots, C_k \overset{0}{\sim} B$.*

Figure 1(b) illustrates such definition. We further define the nodes A and B as ***related***, denoted by $A \sim B$, if either $A \overset{0}{\sim} B$ or there exists $k \geq 1$, such that $A \overset{k}{\sim} B$. Note that even if $A \sim B$, $\|AB\|$ does not necessarily obey normal distribution. In addition, it may be straightforwardly derived that the relation $A \sim B$ is both *commutative* (in that if $A \sim B$, then $B \sim A$), and *transitive* (in that if $A \sim B$ and $B \sim C$, then $A \sim C$).

We now formally define the term ***group*** in our group mobility model.

Definition 3 *Nodes $A_1, A_2, \ldots, A_n$ are in one* ***group*** *G, denoted by $A \in G$, if $\forall i, j, 1 \leq i, j \leq n, A_i \sim A_j$.*

It may be proved[2] that for nodes A and B and groups G, G_1, G_2,

- if $A \in G$ and $A \sim B$, then $B \in G$.
- if $A \in G$ and $\neg(A \sim B)$, then $B \notin G$.
- if $A \in G_1$, $B \in G_2$ and $A \sim B$, then $G_1 = G_2$.
- if $A \in G_1$, $B \in G_2$ and $\neg(A \sim B)$, then $G_1 \neq G_2$.
- if $A \in G_1$ and $A \in G_2$, then $G_1 = G_2$.

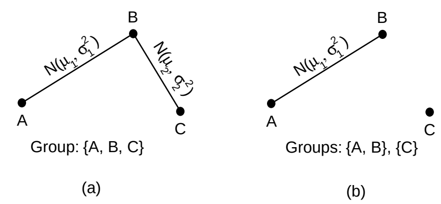


Fig. 2. Grouping Nodes

These properties ensure that groups defined by Definition 3 are *disjoint sets* of nodes in an ad-hoc network. Note that Definition 3 is novel in that group memberships are determined by similarity of mobility patterns (or relative stability of distances) discovered over time, not geographic proximity at any given time. This rules out the misconception that as long as A and B are *neighboring* nodes, they belong to the same group. Figure 2 gives an example. Figure 2(a) shows that $A \overset{0}{\sim} B$ and $B \overset{0}{\sim} C$, hence $A \sim B \sim C$, which forms one group $\{A, B, C\}$. In comparison, Figure 2(b) shows only $A \overset{0}{\sim} B$. In this case, although A and C (or B and C) are neighboring nodes, $A \sim C$ (or $B \sim C$) does not hold. We thus have two disjoint groups $\{A, B\}$ and $\{C\}$. This scenario may arise when two groups are briefly merged geographically but separated again, due to different directions of travel.

3 Theoretical Analysis

The motivation of proposing the group mobility model is to accurately identify groups of nodes that show similar mobility pattern and maintain a stable structure over time. Therefore, it is with high probability that nodes within the same group tend to be mutually reachable. For an adaptive service that includes multiple identical service instances running on individual nodes, this is particularly beneficial to the goal of improving service accessibility with minimum resources. Intuitively, the ideal case is that, should we have an algorithm to capture grouping information with perfect accuracy at any given time, we would have placed *one* service instance in each of the groups, and trivially achieved the best service accessibility with minimum resource overheads.

However, in reality there are two difficulties that prevent us to achieve the ideal scenario. First, groups are detected on the fly with a distributed algorithm based on local states, and thus may not be able to be identified with perfect accuracy. Second, with

[2] Proof omitted for space limitations.

dynamic group membership, service instances may need to be created and terminated even with a perfect grouping algorithm. To address these problems, a realistic approach is to first quantitatively define a QoS parameter as the optimization goal with regards to the adaptive service, then theoretically derive the upper and lower bounds of such a QoS parameter, and finally design a best possible algorithm in realistic scenarios.

3.1 Service Efficiency

We first define two parameters to quantitatively analyze different aspects of QoS in service provisioning. At any given time t, let N be the total number of nodes in the network, $N_s(t)$ be the number of service instances, and $N_a(t)$ be the number of nodes that are reachable from at least one node that runs a *service instance*, thus having access to the adaptive service. We then define ***service coverage*** S_{cover} and ***service cost*** S_{cost} as

$$S_{cover}(t) = \frac{N_a(t)}{N}, \text{ and } S_{cost}(t) = \frac{N_s(t)}{N} \tag{1}$$

The objective is obviously to have the maximum service coverage while incurring the lowest possible service cost. This objective is characterized by the definition of a new QoS parameter, ***service efficiency*** S, defined as

$$S(t) = \frac{S_{cover}(t)}{S_{cost}(t)} = \frac{N_a(t)}{N_s(t)} \tag{2}$$

There is one additional detail related to the definition $S(t)$. Our primary goal for the adaptive service is to reach as many nodes as possible, while reducing the service cost is only secondary. However, (2) treats $S_{cover}(t)$ and $S_{cost}(t)$ with equal weights, which may not yield desired results. For example, assume that we have N nodes and two groups with a split of $2N/3$ and $N/3$, all nodes in each of the groups are reachable from each other. To maximize $S(t)$ in (2), we only need to place one service instance in the larger group and enjoy a service efficiency of $2N/3$, rather than placing two service instances in both groups, having a service efficiency of only $N/2$. Therefore, assuming we have $K(t)$ groups at time t, we need to rectify the definition of $S(t)$ as:

$$S(t) = \frac{S_{cover}(t)}{S_{cost}(t)} = \frac{N_a(t)}{N_s(t)}, \text{ while satisfying } N_s(t) \geq K(t) \tag{3}$$

We may then proceed with the optimization objective of maximizing the service efficiency $S(t)$.

3.2 Theoretical Analysis

In this section, we derive the lower and upper bounds for $S(t)$. Ideally, in an ad-hoc network with N nodes and $K(t)$ groups at time t, if there exists a perfect grouping algorithm to accurately group all nodes, we may trivially select one representative node in each group to host an instance of the adaptive service. We thus have

$$S_{cost}(t) = K(t), \text{ and } S(t) = \frac{N_a(t)}{K(t)} \quad (4)$$

If it happens that for all groups, at time t_0, all nodes in each of the group are able to access the representative node, we have $N_a(t_0) = N$, and $S(t_0) = N/K(t_0)$, which is its optimal value. However, this may not be the case since there exists a low probability that a small subset of nodes in the same group is not reachable from the service instance. For the purpose of deriving global the lower and upper bounds for $S(t)$, we start from examining a group consisting of m nodes. In such a group, the service efficiency is equivalent to the number of nodes that have access to the service instance. We show upper and lower bounds of the average service efficiency in such a group, and then extend our results to the ad-hoc network.

Lemma 1 *If $A \overset{0}{\sim} B$, i.e., $\|AB\|$ obeys $N(\mu, \sigma^2)$ based on Definition 1, then $Pr(\overline{AB} = 0) = 1 - \Phi(\frac{r-\mu}{\sigma})$, where $\Phi(x)$ is defined as $\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-y^2/2} dy$.*

Proof. The probability that A and B are out-of-range nodes is $Pr(\|AB\| > r)$. Hence, $Pr(\overline{AB} = 0) = Pr(\|AB\|) > r) = 1 - Pr(d \leq r) = 1 - \Phi(\frac{r-\mu}{\sigma})$. □

When the transmission range r is definite, $Pr(\overline{AB} = 0)$ increases monotonically as μ increases to approach r and as σ increases.

Lemma 2 *Assume that (1) $A \overset{0}{\sim} B$, i.e., $\|AB\|$ obeys $N(\mu, \sigma^2)$ based on Definition 1; (2) μ obeys uniform distribution in the interval $[0, r]$; and (3) σ also obeys uniform distribution in $[0, \sigma_{max}]$. The* ***average probability*** *$p = Pr(\|AB\|) > r) = 1 - \frac{\int_0^r \int_0^{\sigma_{max}} \Phi(\frac{r-\mu}{\sigma}) d\mu d\sigma}{r * \sigma_{max}}$.*

Proof.

$$\begin{aligned} p &= \frac{\int_0^r \int_0^{\sigma_{max}} 1 - \Phi(\frac{r-\mu}{\sigma}) d\mu d\sigma}{\int_0^r \int_0^{\sigma_{max}} 1 d\mu d\sigma} \\ &= \frac{\int_0^r \int_0^{\sigma_{max}} 1 d\mu d\sigma - \int_0^r \int_0^{\sigma_{max}} \Phi(\frac{r-\mu}{\sigma}) d\mu d\sigma}{\int_0^r \int_0^{\sigma_{max}} 1 d\mu d\sigma} \\ &= 1 - \frac{\int_0^r \int_0^{\sigma_{max}} \Phi(\frac{r-\mu}{\sigma}) d\mu d\sigma}{r * \sigma_{max}} \quad \square \end{aligned}$$

We may then derive the upper and lower bounds for the average number of nodes that are reachable from the service instance, in a group G with m nodes ($m > 0$).

Lemma 3 *If node A_1 hosts the only service instance in G and $A_1 \overset{0}{\sim} A_i, i = 2, \ldots, m$ (Fig. 3), then the average number of nodes that are reachable from A_1 is $m - (m-1)p$.*

Proof. Consider X, the number of nodes that are **not** reachable from the service instance A_1 in group G. Its distribution is a ***binomial distribution*** $B(m-1, p)$, i.e.,

$$Pr(X = k) = \binom{m-1}{k} p^k (1-p)^{m-1-k}$$

X has a mean of $(m-1)p$, and the average number of nodes that are reachable from A_1 is $m - (m-1)p$. □

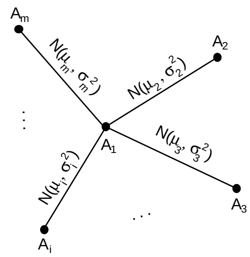


Fig. 3. Star-Grouped Nodes

Theorem 1 *The upper bound of $S(t)$ in G is $m-(m-1)p$.*

Proof. We claim that the average number of nodes that are reachable from A_1 is maximized when group G is formed as a star structure as in Fig. 3. This may be proved as follows. Assume that a node A_i is only reachable from A_1 via an intermediate node A_j. The average probability of this reachability is $Pr(\|A_iA_j\| \leq r)Pr(\|A_jA_1\| \leq r) = (1-p)^2$, obviously it is smaller than $1-p$, which is the average probability of reachability via only a single-hop link. The star structure in a group ensures that all nodes $A_i, i = 2, \ldots, m$ enjoy single-hop reachability to A_1 without depending on intermediate nodes. □

Theorem 2 *Given an ad-hoc network with average group size m, the upper bound of $S(t)$ is $m-(m-1)p$.*

Proof. Assume that there are K groups $G_1, G_2, \ldots, G_K$, with sizes $m_1, m_2, \ldots, m_K$. The total number of nodes in this network is $N = \sum_{i=1}^{K} m_i$, and the average group size $m = N/K = (\sum_{i=1}^{K} m_i)/K$. To achieve the global upper bound of $S(t)$ in the network, upper bounds of $S_i(t)$ should be achieved locally within each of the groups. According to Theorem 1, each group should be formed as a star structure, which achieves an ***upper bound*** of $S_i(t) = m_i - (m_i - 1)p, 1 \leq i \leq K$. Therefore, the upper bound of $S(t)$ is:

$$\frac{\sum_{i=1}^{K}[m_i - (m_i - 1)p]}{K} = \frac{\sum_{i=1}^{K} m_i + Kp - \sum_{i=1}^{K} m_i p}{K} = \frac{N + Kp - Np}{K}$$
$$= m - (m-1)p \quad \square$$

Lemma 4 *In group G with m nodes ($m > 0$), if node A_1 hosts the only service instance and $A_i \overset{0}{\sim} A_{i+1}$, $i = 1, 2, \ldots, m-1$ (Fig. 4), then the average number of nodes that are reachable from A_1 is $\frac{1-(1-p)^m}{p}$.*

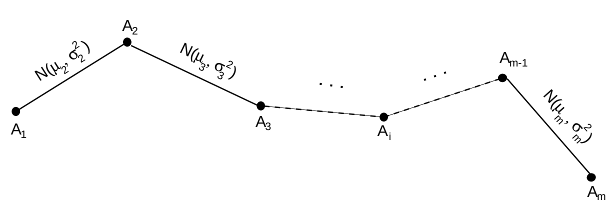


Fig. 4. Chain-Grouped Nodes

Proof. Consider X, the number of nodes that are **not** reachable from the service instance A_1 in group G. The reachability of A_i from A_1 depends on all the links $\overline{A_j A_{j+1}}$, $j = 1, 2, \ldots, i-1$, i.e., if A_{i-1} is not reachable, $A_j, j \geq i$ are not reachable as a result. This leads to

$Pr(X = k)$ = Pr(nodes $A_i, i = m, m-1, \ldots, m-k+1$ are not reachable, all other nodes A_j, $j = 1, 2, \ldots, m-k$ are reachable) =$Pr(\overline{A_{m-k}A_{m-k+1}} = 0$, $\overline{A_j A_{j+1}} = 1$, $j = 1, 2, \ldots, m-k-1)$ =$p(1-p)^{m-k-1}$

Here, if $\overline{A_{i-1}A_i} = 0$, nodes A_j are not reachable for all $j \geq i$, independent from $\overline{A_j A_{j+1}} = 0$ or $\overline{A_j A_{j+1}} = 1$. Observing this, the average number of nodes that are not reachable is $p\sum_{k=1}^{m-1} k(1-p)^{m-k-1}$, denoted by A_{m-1}.

Let $B_{m-1} = \sum_{k=1}^{m-1} k(1-p)^{m-k-1}$, multiply by $1-p$ on both sides, we have

$$
\begin{aligned}
(1-p)B_{m-1} = (1-p)\sum_{k=1}^{m-1} k(1-p)^{m-k-1} &= \sum_{k=1}^{m-1} k(1-p)^{m-k} \\
&= \sum_{j=0}^{m-2} j(1-p)^{m-j-1} + \frac{(1-p)-(1-p)^m}{p} \\
&= B_{m-1} - (m-1) + \frac{(1-p)-(1-p)^m}{p}
\end{aligned}
$$

Subtract B_{m-1} on both sides, we have

$$
pB_{m-1} = (m-1) - \frac{(1-p)-(1-p)^m}{p} \quad (5)
$$

Therefore, the average number of nodes that are **not** reachable from A_1 is A_{m-1} = pB_{m-1} = $m - \frac{1-(1-p)^m}{p}$, and the average number of nodes that are reachable from A_1 is $m - A_{m-1} = \frac{1-(1-p)^m}{p}$. □

Theorem 3 *The lower bound of $S(t)$ in G is* $\frac{1-(1-p)^m}{p}$.

Proof. Based on the proof of Theorem 1, for any node A_i, the more intermediate nodes required from the service instance A_1, the less probable that it is reachable from A_1 at time t. Obviously, the worst case is reached when all nodes in the group form a chain structure as in Fig. 4. □

Theorem 4 *Given an ad-hoc network with minimum group size m_{min}, the lower bound of $S(t)$ is* $\frac{1-(1-p)^{m_{min}}}{p}$.

Proof. Assume there are K groups $G_1, G_2, \ldots, G_k$, with sizes $m_1, m_2, \ldots, m_K$. The smallest group size is $m_{min} = \min\{m_1, m_2, \ldots, m_K\}$. To achieve the global lower bound of $S(t)$ in the network, lower bounds of $S_i(t)$ should be achieved locally within each of the groups. According to Theorem 3, each group should be formed as a chain structure, achieving a ***lower bound*** of $S_i(t) = \frac{1-(1-p)^{m_i}}{p}, 1 \leq i \leq K$. Hence, for the entire network, we have

$$
\begin{aligned}
S(t) = \frac{\sum_{i=1}^{K}[1-(1-p)^{m_i}]/p}{K} &= \frac{K - \sum_{i=1}^{K}(1-p)^{m_i}}{Kp} \\
\geq \frac{K - \sum_{i=1}^{K}(1-p)^{m_{min}}}{Kp} &= \frac{1-(1-p)^{m_{min}}}{p}
\end{aligned}
$$

Therefore, the lower bound of $S(t)$ is $\frac{1-(1-p)^{m_{min}}}{p}$. □

4 Adaptive Service Provisioning Algorithm

Taking the definition of group mobility model (Sect. 2) and its analytical properties, we propose a fully distributed algorithm, referred to as the *adaptive service provisioning algorithm*, that enables dynamic service instance creation and termination in each of the nodes. For this purpose, the algorithm first identifies group memberships of nodes by leveraging the definition of the group mobility model, then selects representative nodes that require creating and terminating service instances. The objective is to maximize *service efficiency* in the network, so that it converges to the upper bound derived in our theoretical analysis. From the proofs of previous theorems, we believe that the upper bound is achieved by having exactly one service instance for each group, if the group mobility model can be utilized to accurately identify the groups at any given time.

In order to address the group division problem in the network, we start by determining if, at time t_0, two neighboring nodes form an *Adjacently Grouped Pair* (AGP). For this purpose, the distance between two neighboring nodes is measured and recorded for a fixed number of rounds l, where l is a pre-determined size of the sampling buffer. The average distance $\bar{d}$ and the standard deviation s may thus be derived from these l samples, which are used to approximate the mean value μ and standard deviation σ in the normal distribution. If the approximated μ and σ complies with **Definition 1** in Sect. 2, the two nodes are identified as an AGP. The advantage of this measurement-based approach is that *Adjacently Grouped Pairs* can be identified at run-time by only relying on local states of each node, e.g., its distances to all neighboring nodes. This conforms with our design objective of minimizing local states and message exchange overheads.

In this algorithm, we assume that each node $A_i (i = 1, \ldots, N)$ has an unique physical ID $id(A_i)$, and at the initial time t_0, there are K_s nodes ($1 \leq K_s << N$) in the network that host service instances of the adaptive service. Our goal is to converge to the upper bound of service efficiency by dynamically initiating new service instances or terminating existing ones, based on identification of groups. On each node, the following local states are maintained:

- **Service Instance ID** $[sid(A_i)]$: the physical ID of the node that hosts the service instance that is currently reachable from A_i.
- **Profile of Measurements** $[P(A_i)]$: a two-dimensional profile in which each row represents one of the neighboring nodes, and each column represents distances to all neighboring nodes obtained from one round of measurements. After l measurements, l samples of distances to A_i are obtained for each neighboring node, denoted by $d_i^{(k)}, k = 1, \ldots, l$.
- **Neighboring nodes in the same group as** A_i $[G_n(A_i)]$: the subset of neighboring nodes that has been identified as in the same group as A_i itself. Note that rather than maintaining all nodes in the same group as A_i, this set only contains *neighboring nodes* that are in the same group.

The algorithm to be executed on a specific node A is given in Fig. 5. Its highlights are illustrated as follows. Initially, at time t_0 when the algorithm starts, all nodes are

At time t**:**

out-list := $\emptyset$;
for each node C_i in $G_n(A)$, $1 \leq i \leq |G_n(A)|$ **do**
  if A and C_i are out-of-range nodes **then** out-list := out-list + $\{C_i\}$;
Current list of neighboring nodes of A is $\{B_1, B_2, \ldots, B_k\}$;

At time $t + (i-1) * \Delta t, 1 \leq i \leq l$**:**

for each neighboring node B_j, $1 \leq j \leq k$ **do**
  Record the distance $d_j^{(i)}$ between A and B_j;
  if A and B_j are out-of-range **then** $d_j^{(i)} := +\infty$;

At time $t + (l-1) * \Delta t$**:**

if $d_j^{(2)} == +\infty$ **then** $d_j^{(2)}$:= r;
for $i = 3, \ldots, l$ **do if** $d_j^{(i)} == +\infty$ **then** $d_j^{(i)}$:= **max**$(r, d_j^{(i-1)} + |d_j^{(i-1)} - d_j^{(i-2)}|)$;
for each neighboring node B_j, $1 \leq j \leq k$ **do**
  calculate the average distance $\bar{d_j} := \sum_{i=1}^{l} d_j^{(i)}/l$;
  calculate the sample estimate of a standard deviation $s_j := \sqrt{\frac{\sum_{i=1}^{l}(d_j^{(i)} - \bar{d_j})}{l-1}}$;
  if $\bar{d_j} < r$ and $s_j < \sigma_{max}$ **then**
    $G_n(A) := G_n(A) + \{B_j\}$;
    if $A \notin G_n(B_j)$ **then** $G_n(B_j) := G_n(B_j) + \{A\}$;
  else if $B_j \in G_n(A)$ **then** $G_n(A) := G_n(A) - B_j$**;**

for each node C_i in out-list, $1 \leq i \leq$ |out-list| **do**
  if A and C_i are out-of-range **then** $G_n(A) := G_n(A) - \{C_i\}$;

if A hosts a service instance **then**
  $sid(A) :=$ max$\{id(A), sid(A), sid(A_j)$ while $A_j \in G_n(A)\}$;
  if $sid(A) \neq id(A)$ **then** terminate the service instance on A;
else
  $sid(A) :=$ max$\{sid(A), sid(A_j)$ while $A_j \in G_n(A)\}$;
if $sid(A) == -1$ **then**
  if there exists a neighboring node $B_j \notin G_n(A)$, $sid(B_j) \neq -1$
  and $sid(B_j)$ is reachable from A **then**
    A sends a *service replication request* to the group of (B_j);
    A starts to execute a new service instance;
    $sid(A) = id(A)$;

Fig. 5. The Adaptive Service Provisioning Algorithm

assigned initial states $G_n(A_i) = \emptyset, i = 1, \ldots, N$, and $sid(A_i) = i$ if A_i hosts a service instance, otherwise $sid(A_i) = -1$. The algorithm then starts to be executed periodically in each of the nodes, updating local states $sid(A_i)$, $P(A_i)$ and $G_n(A_i)$. For a node A, the algorithm may be divided into four phases.

– **Preparation Phase.** As A starts to run the algorithm, it first examines if any nodes in $G_n(A)$ is currently out of range. If so, this node may be previously added to $G_n(A)$ by mistake[3], we thus temporarily add it into a locally maintained *out-list*.
– **Measurement Phase.** For each neighboring node, A measures the distance for l times between itself and its neighboring node[4].
– $G_n(A)$ **Calculation Phase.** According to the profile, if A finds that a neighboring node, e.g., A_k, is in its group, A and A_k will add each other in their respective $G_n(A_i)$. On the other hand, existing nodes in $G_n(A)$ may be removed if measurements do not show AGP properties.
– $sid(A)$ **Update Phase.** If A is hosting a service instance, the service instance ID of A is updated as:

$$sid(A) = \max\{id(A), sid(A), sid(A_j) \text{ while } A_j \in G_n(A)\} \qquad (6)$$

If the updated service instance ID is not $id(A)$, which means another node in the same group is currently hosting a service instance, A will then terminate its own instance. On the other hand, if another node A_i does not host any service instances, and it can not find any service instances in its $G_n(A_i)$, it will probe its non-AGP neighboring nodes and examine if they have access to any service instances. If so, it creates an identical replication of the service instance. Otherwise, the group that A_i is in will continue to be out of reach from any service instances, and they will regularly poll their new non-AGP neighboring nodes to examine if a service instance may be replicated. Once it is replicated, the changes of service instance IDs will be propagated to the entire group.

5 Performance of Adaptive Service Provisioning Algorithm

We conduct simulation experiments to evaluate the performance of the adaptive service provisioning algorithm. The performance metrics that are measured include (1) number of identified groups with different CV_{max} values; (2) service coverage and service cost; (3) service efficiency; (4) *service turnovers*, i.e., migration of service instances due to creations and terminations. This gauges the probability of having stable service instances remain on the same nodes.

The simulated mobile ad-hoc network consisted of 100 mobile hosts roaming in a square region of $800 * 800$ meters, with all boundaries connected, i.e., nodes reaching one edge of the region will emerge on the opposite edge and continue to move on in its previous direction. The transmission range r is set to be $60m$.

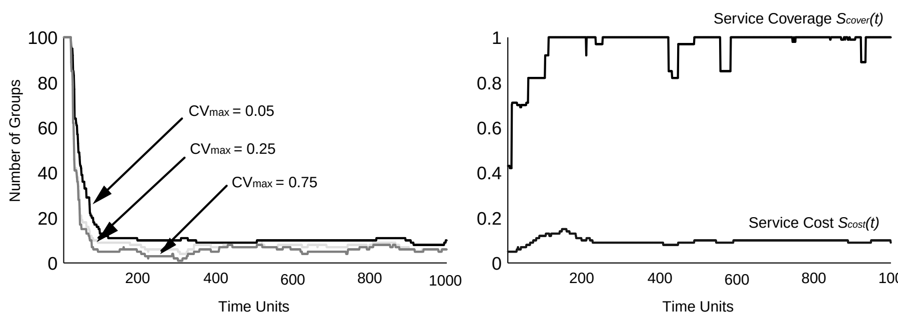
[3] For example, the two nodes may happen to be close to each other when the algorithm was executed in a previous round.

[4] If the samples can not be obtained momentarily because of node mobility, the distance is estimated assuming constant velocity.

We assume that there exists group mobility behavior in the network. When approximating such group movements, we divide the nodes into 10 disjoint sets, each set has a randomly generated size and has independent group-wise mobility pattern. The movement of a particular node consists of a motion vector following group mobility, and another motion vector showing its own random movement. Please note that nodes in the same set are not necessarily identified as in one ***group*** based on our adaptive service provisioning algorithm, since the algorithm is designed to detect groups strictly based on local states in the nodes themselves.

Initially each node A_i is assigned a unique ID $id(A_i)$ and it is in a group of its own. Other parameters in the simulation include (1) $CV_{max} = 0.25$ (except for Fig. 6(a), where we investigate the impact of CV_{max} on the number of groups); (2) The initial number of service instances is 5. (3) Distance sampling size l is 20; (4) Each node runs the algorithm every 100 time units. The simulation runs for 1000 time units.

Figure 6(a) shows that the algorithm is effective and efficient in classifying nodes into groups. The number of groups converges rapidly to a stable value with a small degree of fluctuations. In addition, we have observed that the parameter CV_{max} may affect both the convergence rate and stable values. Such observations are as expected, since larger CV_{max} represents more relaxed criteria for identifying groups. However, we have observed that the effects of CV_{max} on the stable number of groups are insignificant.



(a) Number of Groups with Different CV_{max} Values

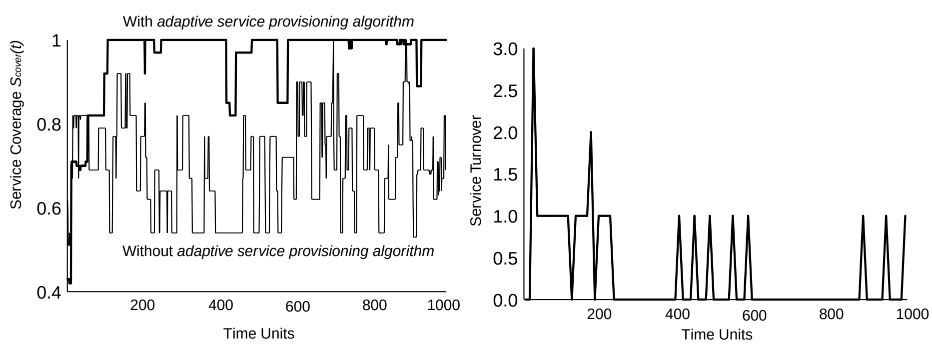
(b) Service Coverage and Service Cost

Fig. 6. Experimental Results: Part I

Figure 6(b) shows the service coverage $S_{cover}(t)$ and service cost $S_{cost}(t)$. We have observed that after the initial stage of convergence, $S_cover(t)$ is generally stable. There are some brief time periods that $S_cover(t)$ decreases, due to the fact that a subset of nodes roam away from a larger group and are thus temporarily out of service. However, the adaptive service resumes after this subset of nodes creates a new service instance by replicating from another passing-by group. With respect to the service cost $S_{cost}(t)$, Figure 6(b) has shown that it remains near a constant and low level.

Figure 7(a) compares the service coverage achieved with and without executing the adaptive service provisioning algorithm. Since the average number of service instances is approximately 8 to 12 when the algorithm is executed on all nodes, we assign 10 service instances in the simulation in which the algorithm is not used. Since the initial number of service instances is only 5 for the case with the algorithm, it is normal that initially the service coverage with the algorithm is less, compared to that without the algorithm executing. However, after a stabilizing period, the service coverage with the algorithm is shown to be better and much more stable than that without the algorithm.

During the simulation, we record the list of service nodes every 10 time units, which is compared to its counterpart in previous time instants. *Service turnovers* are characterized as follows. When a particular node begins to host a new service instance, or when an existing node terminates its service instance, we increment the measurement by 1. Shown in Fig. 7(b), the measured values essentially indicates the frequency of service migration from one node to another. It is only during the starting stage of the simulation that service turnovers are as large as 3, due to initial service replication. Afterwards, service turnovers remain around 0, except for very few time periods when the service instances are rearranged to adapt to behavioral changes in the network.



(a) Service coverage with and without our algorithm

(b) Service Turnovers

Fig. 7. Experimental Results: Part II

Figure 8 shows the service efficiency $S(t)$ with its upper and lower bounds, theoretically derived in Sect. 3. Recall that the upper and lower bounds depend on the m and m_{min}, which is the average and smallest group size, respectively. Since m and m_{min} varies over time, the upper and lower bounds are not constants and vary accordingly. After the initial stabilizing period, $S(t)$ generally remains between the derived upper and lower bounds, except for very rare cases where the observed $S(t)$ is slightly over the upper bound. The reason is as follows. When the upper bound is derived and proved, a node is considered to be out of service if it is not able to access the service in its group; however, in our simulations, there are rare cases in which a particular node can not access any service instances in its own group, but is able to occasionally eavesdrop within its neighboring group. Finally, we may also observe from Fig. 8 that our fully distributed algorithm is able to achieve a service efficiency that effectively converges to long-term stable values of its derived upper bound.

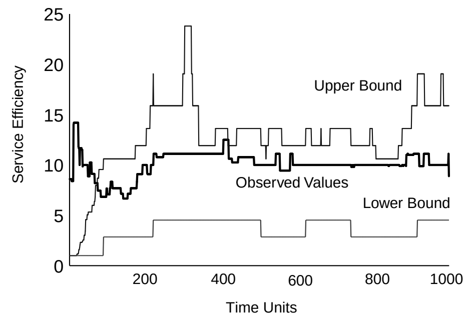


Fig. 8. Service Efficiency $S(t)$ and its Derived Upper and Lower Bounds

6 Conclusions and Future Work

In this paper, we have presented a novel group mobility model that depends only on distances between pairs of nodes to identify groups in an ad-hoc network. Based on such a model, we show a fully distributed and adaptive algorithm that dynamically rearranges the placement of service instances, with an objective of achieving the maximum possible service efficiency. We have illustrated through simulations that our algorithm is effective to achieve such an objective. As part of the future work, we are investigating the problem of network partition prediction. From Fig. 6(b), there is a period of service interruptions when a set of nodes have partitioned from its original group. Should such partitioning be predicted and service instances be replicated, the adaptive service could have been guaranteed without interruptions.

References

[1] D. Johnson and D. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," *Mobile Computing (ed. T. Imielinski and H. Korth)*, 1996.

[2] C. Perkins and E. Royer, "Ad-hoc On-Demand Distance Vector Routing," in *Proceedings of 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99)*, 1999.

[3] J. Chang and L. Tassiulas, "Energy Conserving Routing in Wireless Ad-hoc Networks," *IEEE INFOCOM 2000*, 2000.

[4] S. Chakrabarti and A. Mishra, "QoS Issues in Ad Hoc Wireless Networks," *IEEE Communications Magazine*, pp. 142–148, February 2001.

[5] G. Karumanchi, S. Muralidharan, and R. Prakash, "Information Dissemination in Partitionable Mobile Ad Hoc Networks," in *Proceedings of the 18th IEEE Symposium on Reliable Distributed Systems*, 1999, pp. 4–13.

[6] X. Hong, M. Gerla, G. Pei, and C. Chiang, "A Group Mobility Model for Ad Hoc Wireless Networks," in *Proceedings of the 2nd ACM International Workshop on Modeling and Simulation of Wireless and Mobile Systems*, 1999.

[7] G. Pei, M. Gerla, and X. Hong, "LANMAR: Landmark Routing for Large Scale Wireless Ad Hoc Networks with Group Mobility," in *Proceedings of IEEE/ACM MobiHOC 2000*, 2000.

[8] C. W. Reynolds, "Flocks, Herds, and Schools: A Distributed Behavioral Model," in *Proceedings of ACM ACM SIGGRAPH 87*, July 1987.