Performance Characteristics of Location-Based Group Membership and Data Consistency Algorithms in Mobile Ad Hoc Networks

Jeffery W. Wilson and I.R. Chen

Abstract— Many applications of mobile ad hoc networks require real-time data consistency among moving nodes within a geographical area of interest to function correctly, such as those that support disaster recovery and battlefield command and control. While it is operationally desirable to maintain data consistency among nodes within a large geographical area, the time and network resources required to propagate state changes to all nodes place practical limits on network size. This paper investigates the notion of location-based data consistency in mobile ad hoc networks, and analyzes the tradeoff between data consistency and timeliness of data exchange among nodes within a location-based group in a geographical area of interest. Using a Petri net performance model, we analyze performance characteristics of location-based data consistency maintenance algorithms and identify design conditions under which the system can tradeoff consistency for timeliness (reflecting the time to propagate a state change) while satisfying the imposed data consistency requirement, when given a set of parameters characterizing the application in the underlying mobile *ad hoc* network.

Index Terms— Group membership, data consistency, mobile ad hoc networks, performance analysis.

I. Introduction

Many applications operate in mobile *ad hoc* networking environments which have no fixed connectivity infrastructure. Many of these applications require that nodes have some degree of data consistency among nodes within a community of interest for them to function correctly. Examples include disaster recovery and battlefield command and control applications in which nodes within a geographical area of interest must have consistent data structures that contain kinematic and other characteristic state data that describe objects (e.g., vehicles, cultural features) to satisfy command and control functionality requirements. Such an *ad hoc* environment is characterized by mobile nodes, multi-hop routing, planned and unplanned node disconnection, node failure, relatively low communication system throughput, and unreliable communication. Node mobility is a particular challenge because mobility translates into multi-hop network topology changes, which are reflected in frequent packet route changes and network partitions. The physical environment exacerbates the challenges caused by node mobility. By nature of their mobility, nodes can be expected to exploit cultural and natural features of the physical environment (e.g., investigating or taking shelter in buildings, maneuvering around hills) that have a deleterious effect on communication.

The problem we address in this paper is analyzing the performance characteristics of algorithms for maintaining *location-based data consistency* among a group of nodes in a mobile *ad hoc* environment. It is well known that the problem of reaching agreement ("consensus") among all nodes in asynchronous distributed systems in the presence of failures is deterministically non-solvable even if communication is reliable and at most one peer may crash [3]. A less constrained problem, that of maintaining a "single agreed view" of the group membership among all peers, was also shown to be non-solvable in asynchronous distributed systems where communication is reliable and at most one peer may crash [1]. The notion of *location-based data consistency* considered here does not require a "single agreed view" to be maintained (for which there is no solution). Rather, it allows mobile nodes to join and leave location-based groups, thus allowing multiple data views to coexist in different location-based groups as long as nodes within the same group have the same view of data. This requirement is very useful for applications where data carry geographical meanings, e.g., tracks of objects entering, traveling and leaving a geographical area that are exchanged among all units in the area.

The general problem of group communication in mobile *ad hoc* wireless networks to maintain consistent group membership and, as an extension, to maintain data consistency among members of a group is a relatively new research area. Roman et al. [4, 10] use the idea of safe distance for implementing consistent group membership wherein membership is based on the location information of mobile nodes. The basic idea is to group nodes logically connected within a safe distance ("close enough") in a geographic area and to perform membership-change operations atomically as nodes move into and out of the geographic area. Physical connections that are susceptible to disconnection are

J.W. Wilson and I.R. Chen are with Department of Computer Science, Virginia Tech, Northern Virginia Center, 7054 Haycock Road, Falls Church, VA 22043, USA; phone: (703) 602-6441 x 202; fax: (703) 602-0524; e-mail: {wilsonjw, irchen}@ vt.edu.

considered as "announced" disconnections so the system can perform membership change in one indivisible operation to ensure group membership consistency. In this way, a group expands and contracts atomically, preserving consistent group membership. However, their safe distance-based algorithm is based on the optimistic assumption that disconnections are only caused by node mobility, so the algorithm breaks down when disconnections are caused by environmental factors (e.g., obstructions, jamming) or node failures. Further, the notion of a group based on safe distance is very different from the notion of a group based on a geographical area, because the former considers membership changes due to members being disconnected from each other because of mobility, while the latter (that is, our location-based grouping notion) also explicitly defines membership changes due to members crossing a geographical boundary (e.g., river, mountain ridge).

Killijian et al. [5] introduced the definition of proximity group communication in which group membership depends on location. They associated each proximity group with a static or mobile area of interest within which the group members should be located. They gave a sketch of using a "partition anticipator" executed by every node to detect suspicious partitioning events within a proximity group due to node and link failures to take preventive actions for consistency group membership maintenance. The proactive design to monitor and anticipate partitioning is costly in terms of the overhead required and in reality may be difficult to implement. Our notion of a location-based group is also based on geographical area. However, we take a hybrid of reactive and proactive approaches to maintain consistency in both membership and data by reacting to node and link failure events in the application protocol design. Instead of maintaining consistency all the time, we allow temporary inconsistency to exist during membership and state changes to trade consistency for timeliness, or vice versa. The degree of inconsistency is bounded by the way we check and perform membership and state changes, the rate of which can be adjusted to satisfy the data consistency requirement of the application, given operational and environmental parameters.

A key design issue for maintaining location-based data consistency within a geographical community of interest is to analyze the effect of its "logical size" and determine an optimal geographical area that could maximize consistency. The size directly impacts the time taken for achieving data consistency among members within a geographical area, thus reflecting the bound on data consistency. The optimal size is affected by environmental conditions characterized by parameters such as mobility rate (for mobile units into and out of an area), node failure rate, membership-change detection rate, and data update rate (e.g., for tracking objects entering, traversing and exiting the area). A smaller area incurs a lower latency for message transfer because fewer hops are required, but incurs a higher overhead for membership change operations because of a higher rate of members leaving and joining geographically smaller areas. Consequently, when mobile hosts have low mobility rates, it may permit a smaller geographical area. On the other hand, a larger area incurs a higher latency for data update operations because of more hops in the larger geographical area but incurs a lower overhead for operations associated with consistent group membership maintenance due to node mobility. Thus, an optimal size exists under a given set of parameters characterizing the mobile application in the underlying *ad hoc* network environment. In this paper, we analyze performance characteristics of location-based data consistency maintenance algorithms in terms of the effect of the logical size of a location-based group and the optimal size that can best tradeoff consistency for timeliness (reflecting the time to propagate a state change) while satisfying the imposed data consistency requirement.

The rest of the paper is organized as follows. Section II describes the system model and states the assumptions used in the paper for characterizing the mobile application and the location-based data consistency algorithm developed for *ad hoc* networking environments. Section III presents a performance model for describing the behavior of an application in a mobile *ad hoc* network for achieving location-based data consistency in the presence of various system events, from which we analyze the effect of system parameters affecting the performance of the location-based data consistency maintenance algorithm in terms of the optimal geographical area size and the associated timeliness and consistency attributes. Section IV presents numerical data with physical interpretation given. Section V summarizes the paper and discusses its applicability.

II. System Model

We assume a mobile *ad hoc* network consisting of one or more peers. The network is heterogeneous, with peers in the system having greatly different capabilities. In a battlefield application, for example, one end of the capability spectrum is represented by large command and control nodes (mobile or fixed), such as an aircraft carrier or fixed surface-to-air missile site. At the other end of the spectrum are human-portable devices or pilot-less vehicles with more modest command and control capabilities. Each peer has one or more communication devices and may have organic sensors whose data is shared with other peers in the distributed system. Additionally, each peer has one database in which sensor and other state data is stored.

A. Geographical Community of Interest

The notion of location-based data consistency considered in this work is based on the concept of a geographical community of interest. This concept allows us to move from a requirement for data consistency among all peers in the system to a requirement for data consistency among peers who belong to the same geographical community of interest. We also note that in battlefield applications, as in any other problem-solving, team-oriented application, it is more important (in most cases) to have greater consistency with nearby peers than with ones farther away. In this paper we will use the terms "geographical community of interest" and "location-based group" (or just "group" for short) interchangeably.

The state of the mobile application is characterized by the values of state variables (e.g., track objects in the example battlefield application) maintained by peers in the system. The location-based data consistency requirement means that all the peers within the same geographical community of interest will have the same values for state variables. We assume that peers exchange data regarding the values of state variables within the same community of interest to maintain data consistency. The variables stored in the local database within a peer can be consistent, missing, inconsistent within bound, or inconsistent out of bound when compared against those stored in other peers. The goal is to ensure that all peers within the same geographical community of interest will have a consistent view on state variables in a timely manner as soon as a state change has occurred within the geographical community of interest.

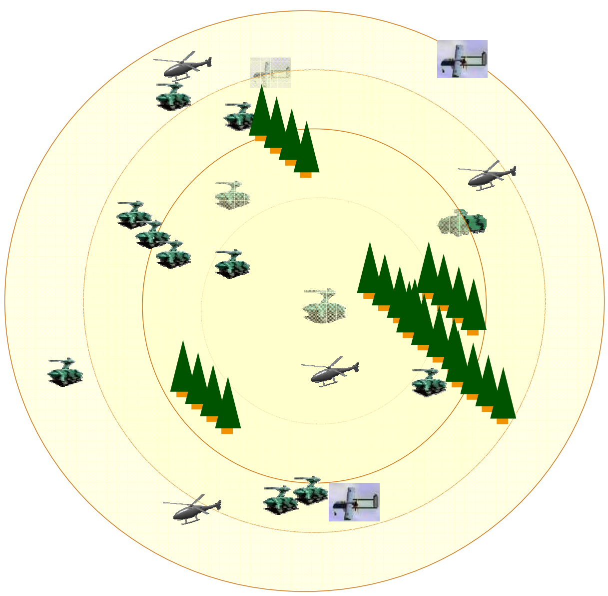

Figure 1: Possible Sizes of a Geographical Area of Interest.

The area of a geographical community of interest can be modeled many ways. Figure 1 shows a plan view of four possible ring sizes for modeling a geographical community of interest. As the area increases, the geographical community of interest covers more mobile hosts (e.g., aircraft, tanks, helicopters shown in the figure). When a mobile host moves into or out of a geographical community of interest bounded by the size chosen (can be any of the four) and the event is detected, a membership change operation is invoked to include or exclude the mobile host in the location-based group membership. We can further divide the terrain into geometric shapes like squares or hexagons and model a geographical area of interest by the number of square or hexagon areas each ring area covers. Figure 2 is a coverage model showing three possible ring sizes for modeling a geographical community of interest based on hexagons, i.e., covering 1, 7 and 19 hexagons, respectively by ring 0, ring 1 and ring 2. Note that two geographical communities of interest may also overlap so that some members located at the overlapping area belong to two communities of interest simultaneously. This is justified for battlefield applications in which each mobile host must be aware of its surroundings so as not to be artificially cut-off by membership boundaries. Also the size of each geographical community of interest may vary depending on the operating conditions. For example, if the mobility rate is low for most mobile hosts, then the size can be small to optimize performance.

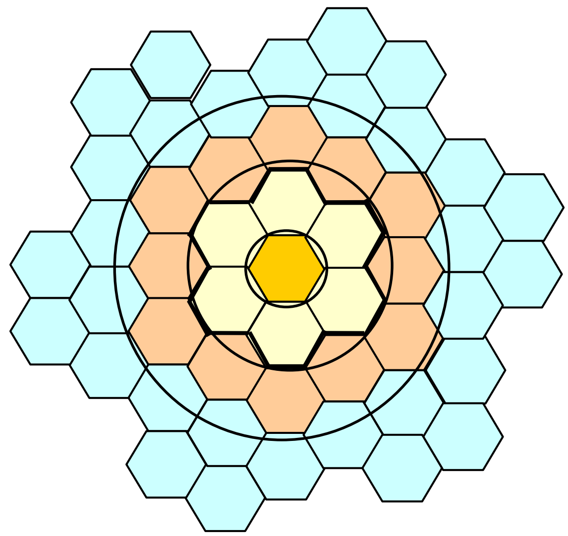

Figure 2: A Representation of a Geographical Area of Interest of Size n based on a Hexagonal Coverage Model.

B. Location-based Group Membership and Data Consistency Algorithm

We assume that each mobile host has a unique host identifier and is equipped with location sensing devices such as a Global Positioning System (GPS) receiver, so it can determine its own location as well as reason about its location relative to the locations of its neighbors within radio range. For a geographical community of interest identified by a group identifier, if the cardinality of the membership set (containing members that are connected in the *ad hoc* environment) is not zero, the mobile host with the smallest host identifier will be elected as the *leader*. The *leader* broadcasts its presence within the community of interest periodically. Should the *leader* fail, the failure event will be detected and the election protocol is used to select a new *leader*. If two or more *leader*s announce their presence, the *leader* with the smallest host identifier wins and the rest relinquish their roles.

When a mobile host enters a new geographical area of interest, it broadcasts a *hello* message containing its location information and host identifier to discover a new location-based group to join. When a host, say A, receives a *hello* message from host B, it informs the corresponding group leader which, in turn, performs a membership change operation to include B in the group. If the leader receives multiple messages regarding B's new membership, it accepts the first and ignores the rest. Conversely, when a mobile host moves out of a geographical community of interest, it informs the group leader of its departure before it moves, who in turn performs a membership change operation to remove the host in the group.

Each mobile host also periodically sends an "I-am-alive" update message to the leader regarding its identifier and location so the leader is aware of who are still within the community of interest. When the leader detects that a member mobile host has not sent its update message, it assumes that the member has been disconnected either due to mobility or failure and removes the mobile host from the group membership. A mobile host can always send a *hello* message to request membership reinstatement if it suspects that it has been removed from the group membership by the leader. This periodic maintenance event allows the leader to actively gather information regarding new and missing members to maintain consistent group membership.

Within a geographical community of interest, if there is a state change detected by any member in the group (e.g., a hostile object approaching), the member will send a message to the leader which in turn forwards the message to all members in the group. For the purpose of this analysis, we adopt the *sensor-pull* and *data-push* design concept [8] by which a peer employs a reactive data dissemination approach to push data to other peers within the community of interest when its *sensor-pull* condition is satisfied, i.e., the cost or value associated with a sensor observation exceeds a predefined threshold (in an absolute sense) or a predefined difference (in a relative sense). Consequently, all peers forward data for which the sensor-pull condition is satisfied to the *leader*. After the *leader* applies application-specific conflict resolution algorithms to resolve differences among different versions of the same data object received from multiple peers (including its own copy), it pushes data that have been updated to all peers in the community of interest. A multicast tree is built dynamically to permit the *leader* to reach all members efficiently, reliably, and securely. Such a multicast tree may be maintained by means of a source-based multicast tree algorithm such as multicast AODV [9].

C. Traffic Model and Performance Metric

We assume that each mobile host has its own distinct mobility rate in and out of geographical communities (groups) of interest. For example, helicopters move faster than tanks which generally move faster than human beings. We assume that the terrain is virtually partitioned into equal-area regions (e.g., hexagons) for ease of analysis and presentation as shown in Figure 2. Let the mobility rate of mobile host i be σ_i moving into and out these regions. Each mobile host also has its own distinct failure rate. Let the failure rate of mobile host i be φ_i. Following the group membership protocol described earlier, let T be the interval at which each mobile host sends its identifier and location information to the leader in an "I-am-alive" update message. The time required for the leader to perform a membership change operation depends on the size of the geographical area. Let $\mu_{mc}(n)$ be the rate at which a membership change operation is executed in a geographical community of interest with a ring size of n (see Figure 2 for illustration), including the time to rebuild a multicast tree by the leader. Similarly the time required for the leader to perform a state update operation also depends on the size of the geographical area. Let $\mu_u(n)$ be the rate at which the leader can propagate an update to members within a geographical community of interest of size n where n is the ring size of the geographical community of interest. Finally, as a larger geographical community of interest is likely to maintain a larger set of state variables (e.g., sensor data in the example battlefield application), let $\delta(n)$ be this sensor-update rate with n again the ring size of the geographical community of interest. This data update rate also depends on the objects to be tracked, for example, the data update rate to track a theater ballistic missile is generally different than that required to track an air-breathing missile or aircraft. All these model parameters identified characterizing the operational conditions of the mobile application in *ad hoc* networking environments are summarized in Table 1. Later in the paper we will show how these parameters can be given appropriate values, properly reflecting the design choice such as the size of a geographical area of interest.

Table 1: Parameters Characterizing Location-Based Data Consistency Applications.

Notation	Meaning
n	Size of a geographical community of interest in terms of the number of concentric rings contained in the area. A size of n rings contains ring 0, ring 1, to ring $n-1$.
$\mu_{mc}(n)$	Rate at which membership changes due to mobility in a geographical area of size n.
$\mu_u(n)$	Rate at which the leader propagates a change of state (or data) information to members within a geographic area of interest with size n.
$\delta(n)$	Rate at which state changes (through data sensing) occur in a geographical area of size n.
σ_i	Rate at which node i moves into and out of a geographical region.
φ_i	Rate at which node i fails.
T	Periodic time for a node to send a "where I am" beacon to the leader.

Our performance metrics measure "timeliness" and "consistency" of state information distributed to members within the geographical community of interest. The timeliness metric is measured by the response time R required to achieve data consistency whenever there is a state change detected by any member within a geographical area of interest. On the other hand, the consistency metric is measured by the proportion of time the system is in a consistent state, which can be broken up into two measures. The first measure PT_m is the proportion of time the group membership is consistent, while the second measure PT_{md} is the proportion of time both membership and state data are consistent among the node members of a location-based group. Our goal is to satisfy the response time requirement while making the consistency measures as high as possible. When there is a constraint in the consistency requirement, the goal is to minimize the response time measure by identifying the best geographical community of interest area size while satisfying the imposed consistency requirement, when given a set of model parameters identified and parameterized characterizing the operational conditions of the mobile application in *ad hoc* networking environments. Table 2 summarizes the performance metrics identified to evaluate location-based data consistency protocols in mobile *ad hoc* networks.

Table 2: Performance Metrics.

Notation	Meaning
R	Response time per state change operation.
PT_m	Proportion of time during which the membership is consistent.
PT_{md}	Proportion of time during which both membership and data are consistent.

III. Performance Model

In this section, we develop a Stochastic Petri net (SPN) performance model to describe the behavior of a mobile application operating under the location-based data consistency algorithm described earlier in Section II. Later in Section IV we utilize this performance model to calculate the timeliness and data consistency metrics to analyze the tradeoff between data consistency and timeliness, given a set of parameter values listed in Table 1 characterizing a given mobile *ad hoc* environment.

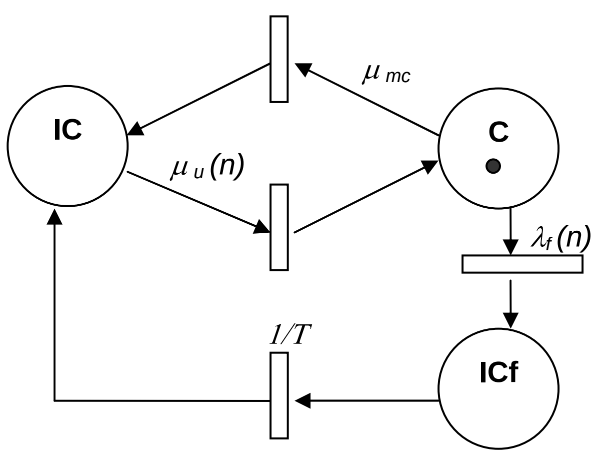

Figure 3: Petri Net Model for Location-Based Data Consistency Algorithm.

Figure 3 shows an SPN model for describing the behavior of the system operating under the location-based membership and data consistency protocol within a geographical area of interest of size n. The SPN model can be viewed as a continuous-time finite state machine [6] which reacts to system events that occur in the system. It is constructed as follows:

- There are 3 places in the Petri net model, with "C" standing for the state in which the system is consistent in membership, "IC" standing the state in which the system is inconsistent in membership due to nodes moving in and out of the geographical area of interest, and "ICf" standing for the state in which the system is inconsistent in membership due to unannounced node failures or disconnections. Initially the system is in a consistent state, represented by having a token deposited in place "C". We use the place at which the token resides to represent the current state of the system as time progresses, so the initial state is "C" as the token is initially placed there.
- Whenever there is a membership change due to arrivals and departures of mobile nodes in and out of the geographical area of interest of size n with rate $\mu_{mc}(n)$, the system migrates from state "C" and state "IC". Our algorithm requires mobile hosts to inform the leader of the membership changes when they move in and out of the location-based group of size n, the discovery rate of which is $\mu_{mc}(n)$. After a membership change detection event occurs, the leader then sends a membership update operation to all members in the location-based group, the rate of which is the same as that for the state-update operation, i.e., $\mu_u(n)$. These behaviors are captured by the two transitions in the upper part of the SPN model with rates $\mu_{mc}(n)$ and $\mu_u(n)$, respectively. Note that the token flows from state "C" to state "IC" and then to state "C" again, reflecting that a membership update event is taken sequentially following a membership change detection event.
- Whenever there is a membership change due to unannounced failure or disconnection of mobile nodes with rate $\lambda_f(n)$, the system migrates from state "C" to state "ICf". This event is modeled by a lower right transition in the SPN model with rate $\lambda_f(n)$. Periodically, the leader will collect and analyze beacon messages sent from mobile members of the location-based group and detect if any member needs to be removed from the membership because of unannounced failure or disconnection events. Consequently, any unannounced failure or disconnection will be detected by the system after a period of time T has elapsed[1]. This detection event is modeled by the lower left transition with a deterministic time period T. Afterward the token flows to place "IC" in which the system performs a membership change operation with rate $\mu_u(n)$ again to bring the group membership consistent. The last event is modeled by having the token flow from place "IC" to place "C" through a transition with rate $\mu_u(n)$ to inform all members of the membership change.

Note that "data change" events are not modeled in the SPN mode. Whenever there is a state change due to sensor detection with rate $\delta(n)$, the system will migrate to another

[1] An alternative SPN model is to use a separate stand-alone transition to model the periodic detection event so it can occur concurrently with other events in the system and then whenever a fixed time period is elapsed check if any unannounced failure has occurred and, if yes, put a token in place "IC". We expect this alternative model would yield the same results but is more computationally expensive so it is not used.

state in which the system will propagate the data update from the mobile user detecting the data change to the leader and then from the leader to all mobile nodes in the location-based group using the multicast tree maintained by the leader with the rate of data propagation being $\mu_u(n)$. For clarity, we do not explicitly model this behavior in the SPN model and instead consider it through probabilistic arguments when we later derive expressions for computing the consistency and timeliness performance metrics.

The system evolves over three states, namely, "C", "IC" and "ICf", as time progresses. Thus, there exists a steady-state probability that the system can be found in one of the three states. Let P_C, P_{IC}, P_{ICf} be the steady state probabilities of states "C", "IC", and "ICf" respectively, which can be obtained by evaluating the SPN model constructed after model parameters are given specific values, characterizing environment- and application-specific operating conditions. Then we can calculate consistency metrics, i.e., PT_m and PT_{md}, as follows:

$$PT_m = P_C \qquad (1)$$

$$PT_{md} = \mu_u(n)\, P_C \,/\, (\mu_u(n) + \lambda(n)) \qquad (2)$$

Equation (1) above gives the proportion of time the system is consistent in membership, which is exactly the same as the equilibrium probability that the system is found in state "C". Equation (2) gives the proportion of time the system is consistent in both membership and data, which is equal to the equilibrium probability that the system is found in state "C" multiplied by the probability that the system is consistent in data, given that the system is consistent in membership. This can be reasoned by considering splitting state "C" into two states "C1" and "C2" such that "C1" is a state that is consistent in both membership and data while state "C2" is a state that is consistent in membership only because a data update propagation operation is still taking place. If one draws a two-state model with "C1" and "C2" such that the rate from "C1" to "C2" is $\delta(n)$ for the data-change transition (due to sensing) while the rate from "C2" to "C1" is $\mu_u(n)$ for the data-update transition (for propagating updated data to members), then one will see that the probability that the system is consistent in both membership and data, i.e., in state "C1", given that it is in state "C", is equal to $\mu_u(n) \,/\, (\mu_u(n) + \lambda(n))$.

The timeliness metric can be calculated by the average of the response times obtained in various states weighted by their respective state probabilities, i.e.,

$$R = (P_C + P_{ICf})/\mu_u(n) + P_{IC}\,(1/\mu_{mc}(n) + 1/\mu_u(n)) \quad (3)$$

Here the first term accounts for the response time when the system is in either state "C" or state "ICf", which incurs an average update propagation time of $1/\mu_u(n)$, while the second term accounts for the response time when the system is in state "IC" which incurs a waiting time of $1/\mu_{mc}(n)$ to account for the extra time required to process the membership change operation before taking another $1/\mu_u(n)$ time to process the data propagation operation by the system (leader). Here we note that while the system is in state "ICf", the leader will only propagate data to members inconsistently since in state "ICf" the system is in a state in which the leader is not aware of the fact that the group membership is inconsistent. Contrarily, the system is fully aware of its membership inconsistency in state "IC", in which case the leader is in the process of performing a membership change operation, so a data propagation operation newly arriving must wait for the membership operation to execute to completion before being processed by the leader, thus incurring a waiting time to the response time.

We also note that when T is small, the probability of the system found in state "ICf" will be small since the moment the system is in state "ICf" it will transit to state "IC" quickly in which a membership change operation will be executed to maintain membership consistency. Thus a small T improves membership and data consistency while compromising the response time performance metric, and vice versa, and there exists a tradeoff between the consistency metrics (as given by Equations (1) and (2)) and the timeliness metric (as given by Equation (3)).

IV. ANALYSIS

In this section, we show how to give appropriate values of the parameters of the SPN model developed in Section III and devise a computational procedure for computing the data consistency and timeliness metrics. We analyze design tradeoffs between the consistency and timeliness metrics and identify conditions under which the system is able to satisfy the response time requirement while maximizing the consistency metrics. We use SPNP [11] as a tool to define and evaluate the SPN models developed to yield numerical results with physical interpretations given.

A. Parameterization

Consider a mobile *ad hoc* network modeled by a hexagonal network coverage model as illustrated in Figure 2 with the center region in ring 0. Also consider a location-based group with a geographical area of interest of size n covering ring 0 through ring $n-1$. For a mobile node, say, node i, in the area, let λ^n_i be the "outward" mobility rate of mobile node i to go out of ring n into ring $n+1$ and μ^n_i be the "inward" mobility rate of the mobile node to go out of ring n into ring $n-1$.

The specific values of λ^n_i and μ^n_i for mobile node i depend on the semantics of the mobile applications and the mobility model of the mobile node. As an example, consider a node that follows a random walk mobility model. It can be shown that [7] when a mobile node is in ring n, the probabilities of the mobile node randomly moving outward to ring $n+1$, moving inward to ring $n-1$, and staying within ring n, upon a movement out of a hexagon region, denoted by P_{omove}, P_{imove} and P_{smove}, respectively, are given by:

$$P_{omove} = \begin{cases} 1 & if\ n = 0 \\ \dfrac{2n+1}{6n} & otherwise \end{cases}$$
$$P_{imove} = \begin{cases} 0 & if\ n = 0 \\ \dfrac{2n-1}{6n} & otherwise \end{cases} \quad (4)$$
$$P_{smove} = \begin{cases} 0 & if\ n = 0 \\ \dfrac{2n}{6n} = \dfrac{1}{3} & otherwise \end{cases}$$

Let σ_i represent the user mobility rate of mobile node *i* moving across hexagonal areas. Again let λ^n_i be the outward mobility rate of mobile node *i* to go out of ring *n* into ring *n+1* and μ^n_i be the inward mobility rate of the mobile node to go out of ring *n* into ring *n-1*. Then,

$$\lambda_i^n = \begin{cases} \sigma_i & if\ n = 0 \\ \dfrac{2n+1}{6n}\sigma_i & otherwise \end{cases}$$
$$\mu_i^n = \begin{cases} 0 & if\ n = 0 \\ \dfrac{2n-1}{6n}\sigma_i & otherwise \end{cases} \quad (5)$$

Now consider that the mobile *ad hoc* network is populated with mobile nodes with an average density of M_i users per hexagonal area located at ring i. For the uniform density case, all M_i's are the same, say, equal to M_0. The more reasonable case is that there are more mobile nodes close to the center of the geographical area (since they are interested in the area and are members of the location-based group) and fewer nodes as we move further away from the center of the geographical area. This inhomogeneous density distribution can be modeled by a population function with an exponential decay behavior. Let M_0 be the density of the center hexagon in a geographical area of interest, then M_i is given by:

$$M_i = \frac{M_0}{b^i} \quad (6)$$

Here *b* is the population decay parameter whose magnitude represents how fast the population density decays as we move away from the center of attention in the geographical area, with the special case *b*=1 being the uniform density case. Note that M_i is also the steady-state population density of a hexagon in ring *i* assuming each node follows the random walk mobility model with the property that it will return to its origin with probability 1. Since a geographical area of interest of size *n* contains $3n^2 - 3n + 1$ hexagons, so there are $(3(n+1)^2 - 3(n+1) + 1) - (3n^2 - 3n + 1) = 6n$ hexagons in ring *n*, with *n*>0. For example, ring 1 contains 6 and ring 2 contains 12 hexagons, and so on. Since only nodes in ring *n* moving inward to ring *n-1* and nodes in ring *n-1* moving outward to ring *n* will trigger a location-based membership change operation, the overall rate at which all the mobile nodes will trigger a membership change for a geographical area of interest of size *n* (consisting of ring 0 to ring *n-1*) due to mobility, defined as $\mu_{mc}(n)$, is given by:

$$\mu_{mc}(n) = 6(n-1)M_{n-1}\lambda^{n-1} + 6nM_n\mu^n \quad (7)$$

Here the first term accounts for the rate at which mobile nodes move out of the geographical area of interest of size *n*, and the second term accounts for the rate at which mobile nodes move into the area, both triggering a membership change operation. Note that we have dropped the subscript *i* from $\lambda^n{}_i$ and $\mu^n{}_i$ to refer to the fact that all mobile nodes have been considered in Equation (7).

Since a geographical area of interest of size *n* on average will contain $M_0+6M_1+12M_2+ \ldots + 6(n-1)M_{n-1}$ mobile nodes, the rate at which mobile users within a geographical area of interest of size *n* fail or disconnect unannounced, $\lambda_f(n)$, is given by:

$$\lambda_f(n) = \phi(M_0 + \sum_{j=1}^{n-1} 6jM_j) \quad (8)$$

The time for a leader to propagate a state-update operation or a membership change operation to all members in the geographical area depends on whether the propagation method is broadcast-based or multicast-based. Suppose that we adopt multicast-based. Then the propagation time depends on the number of members in the group and the way the leader builds the multicast tree to reach all members. Assume a perfect balanced tree. Then on average it takes $\log_2(M_0 + \sum_{j=1}^{n-1} 6jM_j)$ hops to reach all members and the communication time per hop is τ depending on the underlying communication technology deployed in the *ad hoc* network. Consequently, the rate at which the leader performs a state-update operation to all members in the location-based group of size *n*, $\mu_u(n)$, is given by:

$$\mu_u(n) = \frac{1}{\tau \log_2[M_0 + \sum_{j=1}^{n-1} 6jM_j]} \quad (9)$$

Equations (5), (6), (7), (8) and (9) thus parameterize model parameters $\mu_{mc}(n)$, $\lambda_f(n)$ and $\mu_u(n)$ once we are given values of basic parameters M_0 and *b* (density of mobile nodes), σ (mobility rate per node), φ (failure rate per mobile node) and τ (communication delay per hop) characterizing the network and application operating conditions.

B. Computational Procedure for Calculating Timeliness and Consistency Metrics

To calculate PT_m and $PTmd$ based on Equations (1) and (2), we must obtain the steady state probabilities P_C, P_{IC}, and P_{ICf}. SPNP [11] was used to help obtaining these probabilities when given a set of parameter values characterizing the network and workload conditions. Specifically, we used the following reward assignment to calculate P_k:

$$r_k = \begin{cases} 1 & \text{if } mark("k") = 1 \\ 0 & \text{otherwise} \end{cases}$$

We use the following reward assignment based on Equation (3) to calculate R:

$$r_k = \begin{cases} \dfrac{1}{\mu_u(n)} & \text{if } mark("C") = 1, \text{ or } mark("ICf") = 1 \\ \dfrac{2}{\mu_u(n)} & \text{if } mark("IC") = 1 \end{cases}$$

In effect, this calculates the average reward weighted by the state probabilities, which in this case, is exactly the weighted response time per data update operation in a geographical area of size n.

Table 3: Basic Parameters and Their Default Values.

Notation	Meaning	Default Value
τ	Hop-by-hop communication delay in the *ad hoc* network.	1
M_0	Population density per hexagon for the center hexagon in the location-based group.	2
b	Decay parameter for the population density based on (6).	4
$\delta(n)$	Rate at which state changes (through data sensing) occur in a geographical area of size *n,* assuming it is a constant not sensitive to the size *n*.	0.01
σ	Rate at which a node moves across hexagonal regions.	0.05
φ	Rate at which a node fails.	0.001
T	Periodic time for a node to send a "where I am" beacon to the leader.	5

C. Numerical Data

Here we present numerical data obtained from evaluating the Petri net model developed based on Equations (1), (2) and (3) to show design tradeoffs between the timeliness (R) and consistency metrics (PT_m and PT_{md}) obtained, as a result of applying our location-based data consistency algorithm in mobile *ad hoc* networking environments. The set of parameters characterizing the mobile application in the underlying *ad hoc* network environment is given in Table 3 with their default values. These parameters are normalized with respect to τ=1 (hop-by-hop delay) for ease of presentation, e.g., φ=0.001 means that the failure rate on average is once per 1000τ and T=5 means that the periodic check is about once every 5τ period. We analyzed the effects of some of these parameters by changing their values to observe their effect on R, PT_m and PT_{md}.

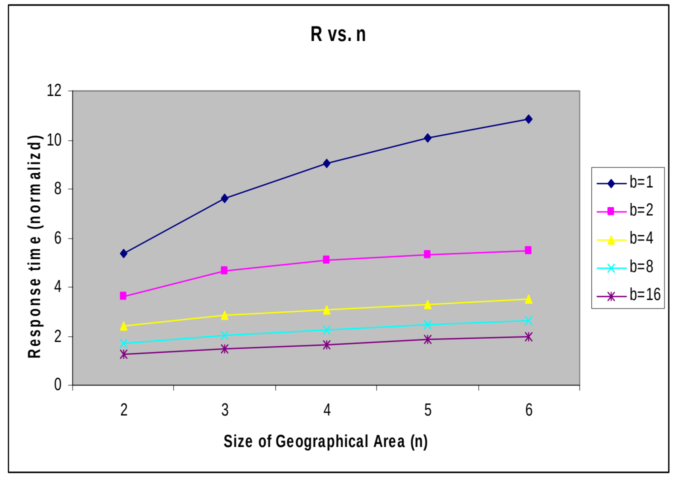

Figure 4: Timeliness Metric (Response Time) vs. Geographical Area Size (n) under various Population Density Models.

Figure 4 shows the timeliness metric (in the form of the response time R normalized with respect to τ) versus the size of the geographical area of a location-based group under various density models ranging from b=1 (uniform density) to b=16 (extremely inhomogeneous density). As expected, as the size (n) of a location-based group grows from 2 to 6, the response time increases monotonically. At the extreme case b=1 in which all hexagons have the same uniform density of M_0 (whose default value is 2) regardless of their locations relative to the center of the location-based group, the growth of response time is much more rapid compared to other inhomogeneous population density cases in which the node population decreases exponentially by a factor of b^n. For the extreme case b=16, the response time virtually remains much the same because the number of nodes (or number of members in the group) hardly grows as n increases.

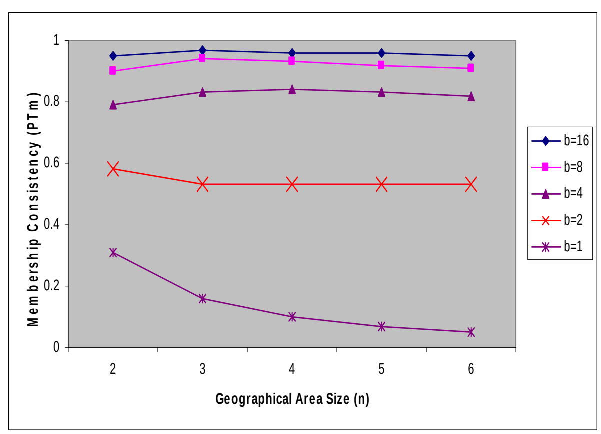

Figure 5: Membership Consistency (PT_m) vs. Geographical Area Size (n) under various Population Density Models.

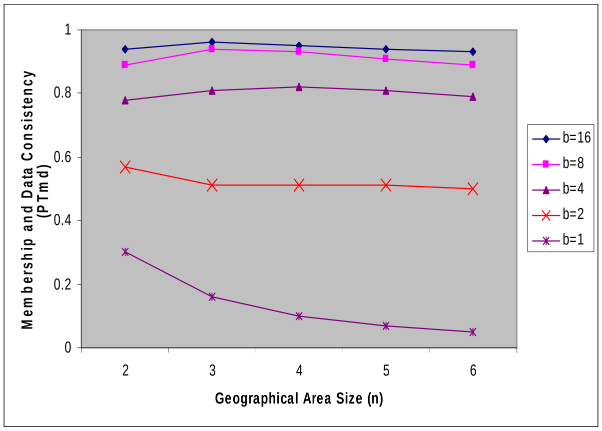

Figure 6: Membership and Data Consistency (PT_{md}) vs. Geographical Area Size (n) under various Population Density Models.

Figures 5 and 6 show the consistency metrics PT_m and PT_{md} versus the size of geographical area of a location-based group. Unlike the timeliness metric that increases monotonically with n, we observe that there exists an optimal n, say, n_{opt}, at which the consistency metric is maximized. For example, when b=1 or 2, n_{opt} =2, when b=4, n_{opt} =4 and when b=8 or 16, n_{opt},=3 in both Figures 5 and 6.

The reason that an optimal area size exists for maximizing membership and data consistency is that membership inconsistency is attributed to the system being in state "IC" due to mobility events for nodes in and out of the group, and also in state "ICf" due to failure events for member nodes. The rate of node failure events is directly proportional to the number of member nodes in the location-based group. Thus, as n increases, more failure events are likely to occur, as there are more member nodes in the group. On the other hand, the rate at which mobility events occur due to nodes moving into and out of the geographical area is not necessarily proportional to n. For the inhomogeneous population model defined by Equation (5), the rate of membership changes induced by user mobility actually decreases as n increases when b>2, because there are fewer nodes residing at the outer hexagons (due to exponential population decay) as we move away from the center hexagon of the location-based group, so most nodes in the group are likely to be contained within the area when n is large. These two effects counterbalance each other, thus resulting in an optimal area size that maximizes the membership and data consistency metrics.

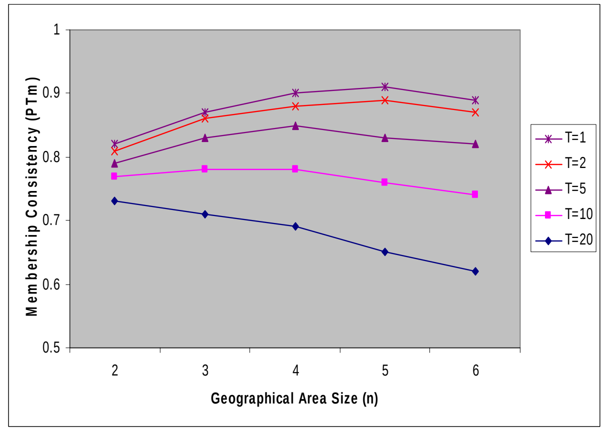

Figure 7: Effect of T on Membership Consistency (PT_m) and Optimal n_{opt}.

At a large group size (i.e., a large n) we can expect that practically there will be little mobility-induced membership changes since all mobile nodes would be reasonably contained within the area most of the time if b>2. Most membership change operations incurred in this case would be due to node failures whose rate increases as n increases. Consequently, we would expect that if we detect node failures more frequently (with a smaller T), a large area would become more favorable than a small area to obtain higher membership and data consistency. This is exactly the case illustrated in Figure 7 in which we show the effect of T on the optimal size n_{opt} for membership consistency. (The graph for the effect of T on the optimal size n_{opt} for both membership and data consistency exhibits the same trend and is not shown for clarity.) We see that for the same operational condition (b=4 is chosen as the example), as T decreases n_{opt} increases, e.g., n_{opt} goes from 1 to 4 as T goes from 10τ to τ, because with a smaller T, membership changes due to node failures can be performed more rapidly, thus favoring a larger area for which membership changes are mostly due to node failures.

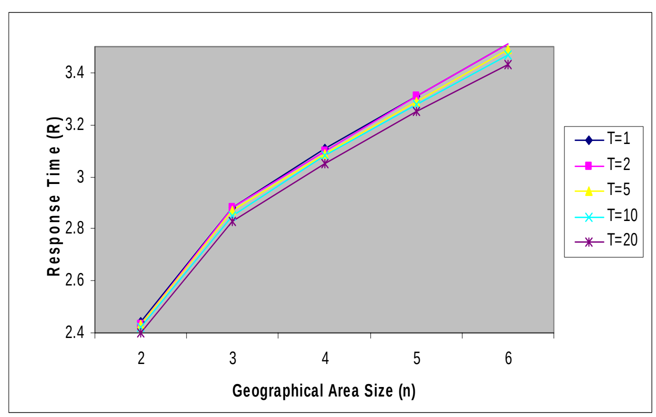

Figure 8: Adverse Effect of T on Response Time R.

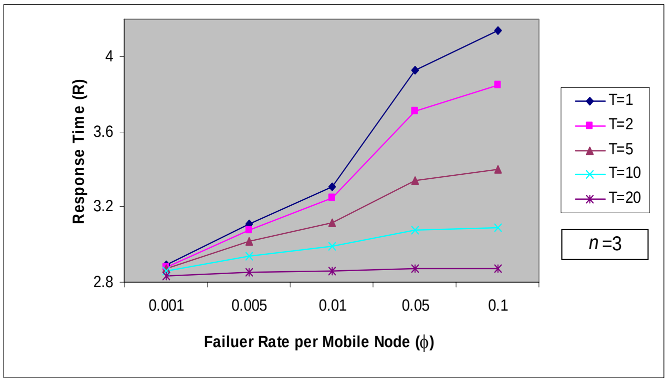

Figure 9: Pronounced Adverse Effect of T on R at High Failure Rate (φ).

With the above analysis, we know that we could achieve reasonably high membership consistency with a large geographical area size n, and a small T. Unfortunately, both a large n and a small T adversely degrade the response time metric. (The effect of n on the response time metric was shown in Figure 5.) The effect of T on the response time is shown in Figure 8 where we see that a more frequent periodic detection activity (i.e., a smaller T) degrades the response time metric more because more time will be spent by the leader to do membership maintenance induced by node failures, thus causing any concurrent state-change operation to be delayed. Isolating n=3 as a case study, Figure 9 shows that the adverse effect of T on R is especially pronounced when the failure rate (φ) is high at which the leader must perform failure-induced membership change operations frequently to maintain membership consistency, thus causing a high delay in the response time per state-change operation. Here we must emphasize that the response time metric (R) only reflects the average response time per data update operation as calculated by Equation (3); it includes both the case in which data are propagated consistently (in states "C" or "IC") and the case in which data are not propagated consistently (in state "ICf") to all members of the location-based group.

V. APPLICABILITY

To apply the results obtained in the paper, one would consider a set of parameter values characterizing the application and networking conditions, including the population density parameters (M_0 and b), hop-to-hop communication delay of the underlying *ad hoc* network (τ), failure rate per node (φ), mobility rate per node (σ), and state-change rate due to object sensing (δ), as well as a set of threshold values for the timeliness and consistency metrics (R^*, PT^*_m, and PT^*_{md}) specifying the minimum QoS requirements for the application. Then, one would fine-tune the detection period (T) to identify the best area size (n_{opt}) for the location-based group such that the timeliness requirement ($R < R^*$) is satisfied while maximizing the consistency requirements, i.e., obtaining the maximum PT_m and PT_{md} values among all that satisfy $PT_m > PT^*_m$ and $PT_{md} > PT^*_{md}$.

The Petri net model developed in the paper allows such an evaluation to be done at run time efficiently using evaluation tools such as SPNP 6 [11] requiring only a few seconds to run on a SUN workstation.

As an example, consider a specific application [2] in a mobile *ad hoc* network environment for which a hexagonal area roughly covers a single-hop radio range of 300 meters with M_0 = 4 (mobile nodes per hexagon at the center of the location-based group), b = 4, τ = 0.01 (sec), φ = 10 failures or disconnections/node/hour), σ = 0.2 (hexagon areas crossed per sec) and δ = 2 (updates/sec) characterizing the operational and network conditions of the application[2], and with R^* = 0.04 (sec), PT^*_m = 0.98, and PT^*_{md} = 0.92 specifying the imposed QoS requirements.

Table 4: Legitimate *(n, T)* Pairs that Satisfy the Imposed QoS Requirements with the 3rd Entry Yielding the Largest PT_m.

n	T	PT_m	PT_{md}	R
3	0.01	0.9868	0.9261	0.0332
4	0.01	0.9917	0.9256	0.0360
5	0.01	0.9929	0.9231	0.0380
3	0.05	0.9839	0.9234	0.0332
4	0.05	0.9878	0.9220	0.0360
3	0.1	0.9804	0.9201	0.0332

By applying the location-based algorithm and utilizing the analysis methodology developed, we obtained a combination of (n, T) values at which the imposed QoS constraints are satisfied as listed in Table 4. Unlike the consistency metrics, namely PT_m and PT_{md} which are heavily sensitive to T, the timeliness metric R is not sensitive to T over a wide range (i.e., 0.01 sec to 1 sec) because the failure rate per node is relatively low in the application environment. The best combination of (n, T) is found to be the third entry with (n, T) = (5, 0.01) in Table 4 that satisfies $R < R^*$, $PT_m > PT^*_m$ and $PT_{md} > PT^*_{md}$ while maximizing PT_m among all. Note that PT_{md} is calculated based on Equation (2), so it is sensitive to the values of $\mu_u(n)$ and δ and does not necessarily follow the same trend as PT_m with respect to the geographical area size n of the location-based group. Therefore, if PT_{md} is deemed a more important indicator of consistency than PT_m, then the first entry with (n, T) = (3, 0.01) should be chosen to maximize PT_{md}.

On the other hand, if the specific application is operating in a hostile environment in which mobile nodes fail frequently, we would not necessarily choose a (n, T) pair with a small T

[2] Assume that mobile nodes move fast and are in a hostile environment in this application. The mobility rate of 0.2 hexagon areas crossed per second corresponds to 60 meters/sec (or 216 km/hour). Also note that all other parameters values used here are absolute values, as opposed to those used in Section 4 which were normalized with respect to τ =1 to ease the presentation.

because under a higher per-node failure rate operational condition, a small T will adversely affect the response time R. Finally, if the application dictates an area size for the location-based group because of the nature of the mission, one can utilize the methodology developed to determine the best value T that would satisfy the imposed QoS requirements as well as to evaluate if the system is able to satisfy the imposed QoS requirements should the failure rate per node become high when employed in hostile environments.

VI. SUMMARY

The analysis performed in the paper reveals that there is a tradeoff between the timeliness metric (R) and consistency metrics (PT_m and PT_{md}) in location-based groups in mobile *ad hoc* networks, depending on the values of the set of parameters (identified in Table 3) characterizing the application and networking conditions. We explored the tradeoff by examining a population density model where the population densities of hexagons decay exponentially by a factor of b^n with n being the size of the geographical area of the location-based group and b being the decay parameter. Our analysis results showed that if all hexagons are of equal density (i.e., b=1) then the system would always favor a small area (i.e., n=2) to minimize the response time and maximize the membership and data consistency metrics. However, for inhomogeneous population density where the mobile node population decays exponentially for outward hexagons away from the center of the location-based group, as normally is the case for mission-oriented applications in *ad hoc* network environments, we observed that there exists an optimal area size n_{opt} that maximizes the consistency metrics.

We also observed that this optimal area size n_{opt} increases as the time period (T) for mobile nodes to send "I-am-alive" beacon messages to the group leader decreases, because a smaller time period allows failure-induced membership changes to be more rapidly detected and corrected, thus favoring a location-based group with a larger area size for which most of the membership change operations are due to node failures. Although a short period T favors a large area size in terms of improved consistency in both membership and data, a short T will adversely degrade the response time metric, the effect of which is especially pronounced when the node failure rate is high. In addition, a larger area inherently has a higher response time per state-change operation. We conclude that a short T and a large area size n in general improve the consistency metrics at the expense of the timeliness metric, and vice versa. Whether we should select a short T and a large area size n to yield high consistency at the expense of timeliness, or vice versa, depends on the application's QoS requirements in consistency and timeliness.

ACKNOWLEDGMENT

The authors would like to thank Frank Driscoll, Karen Rigopoulos, Phu-Gui Feng, and Holly Xiao of The MITRE Corporation for supporting part of the work.

REFERENCES

[1] T.D. Chandra, V. Hadzilacos, S. Toueg and B. Charron-Bost, "On the impossibility of group membership," *15th ACM Symposium on Principles of Distributed Computing*, New York, USA, 1996, pp. 322-330.

[2] J. Dahmann and J.W. Wilson, "MDA and HLA: Applying standards to development, integration and test of the Single Integrated Air Picture Integrated Architecture Behavior Model," *Fall Simulation Interoperability Workshop (SIW)*, Orlando, Florida, 2003.

[3] M.J. Fisher, N.A. Lynch and M.S. Paterson, "Impossibility of distributed consensus with one faulty process,'' *Journal of the ACM*, Vol. 32, April 1985, pp. 427-469.

[4] Q. Huang, C. Julien and G.-C. Roman, "Relying on safe distance to achieve strong partitionable group membership in ad hoc networks," *IEEE Transactions on Mobile Computing*, Vol. 3, No. 2, 2004, pp. 192-205.

[5] M.-O. Killijian, R. Cunningham, R. Meier, L. Mazare and V. Cahill, "Towards group communication for mobile participants,'' *ACM 1st International Workshop on Principle of Mobile Computing*, Newport, Rhode Island, USA.

[6] L. Kleinrock, *Queueing Systems*, John Wiley and Sons, 1975.

[7] Y.B. Lin, L.F. Chang and A. Noerpel, "Modeling Hierarchical Microcell and Macrocell PCS Architecture", *ICC,* June 1995.

[8] Manjeshwar, A.A. Zeng and D.P. Agrawal, "An analytical model for information retrieval in wireless sensor networks using enhanced APTEEN protocol,'' *IEEE Trans. Parallel and Distributed System*s, Vol. 13, No. 12, Dec. 2002, pp. 1290-1302.

[9] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," *2nd IEEE Workshop on Mobile Computing Systems and Applications*, Feb. 1999.

[10] G.-C. Roman, A. Huang and A. Hazemi, "Consistent group membership in ad hoc networks,'' *IEEE International Conference on Software Engineering, 2001*, pp. 381-388.

[11] Stochastic Petri Net Package (SPNP) User's Manual, Version 6, Department of Electrical and Computer Engineering, Duke University, 1999.

Jeffery W. Wilson is a Ph.D. Candidate at the Virginia Polytechnic Institute and State University, and is a graduate of Virginia Tech (MS, Computer Science and Application, 2002), the Naval Postgraduate School (MS, Physics, 1985), and Oregon State University (BS, Computer Science, 1978).

He is a 26-year veteran of the U.S. Navy, where he currently serves as Technical Director of the Joint Single Integrated Air Picture System Engineering Organization in Arlington, Virginia. In this capacity, he leads technical and programmatic efforts to improve joint tactical warfighting capability within the aerospace warfare mission domain. In prior positions, he had technical leadership responsibilities for Navy Theater Air and Missile Defense programs, including the design, development, testing, and fielding of the AEGIS Combat System in TICONDEROGA Class Guided Missile Cruisers and ARLEIGH BURKE Class Guided Missile Destroyers.

Mr. Wilson is a member of ACM, IEEE, and Sigma Xi.

Ing-Ray Chen received the BS degree from the National Taiwan University, Taipei, Taiwan, and the MS and PhD degrees in computer science from the University of Houston, Texas. He is currently an associate professor in the Department of Computer Science at Virginia Tech. His research interests include mobile computing, pervasive computing, multimedia, distributed systems, real-time intelligent systems, and reliability and performance analysis. Dr. Chen has served on the program committee of numerous conferences, including being the Program Chair of 29th IEEE International Computer Software and Applications Conference (COMPSAC), 14th IEEE International Conference on Tools with Artificial Intelligence, and 3rd IEEE Symposium on Application-Specific Systems and Software Engineering Technology. Dr. Chen serves as an Associate Editor for *IEEE Transactions on Knowledge and Data Engineering*, *The Computer Journal*, and *International Journal on Artificial Intelligence Tools*. He is a member of the IEEE/CS and ACM.