

Data Replication in Mobile Tactical Networks

Yang Zhang*, Sucharita Ray*, Guohong Cao*, Tom La Porta*, and Prithwish Basu†

* The Pennsylvania State University, E-mail: {yangzhan, sxr360, gcao, tlp}@cse.psu.edu

† BBN, E-mail: pbasu@bbn.com

Abstract—In tactical networks, nodes move according to tactical maneuvers and network partitions occur frequently. To mitigate this problem, data replication is commonly used to increase data availability and reduce data access delay. However, different tactical maneuvers lead to different node mobility models, which affect the performance of data replication schemes. In this paper, we comprehensively study the data replication problem in mobile tactical networks. We propose a new intra-group data replication scheme and extensively quantify the effects of mobility on different inter-group data replication schemes from various perspectives. The study is based on several metrics, which include the average access delay and data availability, and the temporal and spatial analysis of these values. Through extensive experiments, we study the effects of three typical mobility models in tactical networks on data replication, and identify the most suitable data replication schemes under various mobility models.

I. INTRODUCTION

In scenarios of military operations such as in a battlefield, squads of reconnaissance, surveillance and attack task forces form a mobile tactical network to accomplish a mission [1, 2]. As the mission progresses, a large amount of information must be shared among soldiers and squads [3, 4]. However, since soldiers move according to tactical maneuvers, network partitions may occur, where nodes in one partition cannot access data held by nodes in other partitions [5]. To mitigate this problem, data replication can be used. By replicating data into a number of nodes in different squads, a data request can be served by the closest node that has the data replica [6]. Then, even if there is a network partition between the requesting node and the original data source, the data request can still be served as long as it can reach a node with a data replica. Moreover, data access delay is reduced since the data request can be served with fewer hops.

Data replication increases data availability and reduces data access delay at the cost of data storage. Since soldiers' communication devices usually have limited storage space, bandwidth, and power, it is impossible for one node to hold all the data. Therefore, it is important for nodes to cooperate with each other to decide which node should hold which data replica. To increase data availability, a node may not hold the data which has already been replicated by neighbors so that its local storage can be used to hold additional data. However, this may increase the hop count for some data and increase data access delay. The problem becomes more complex when mobility is considered, since mobility can change the location of the data replica, and then affect data availability and data access delay.

In this paper, we focus on three popular mobility scenarios in tactical networks. Our aim is to find the characteristics of the mobility in tactical networks and to study the effect of various mobility types on data replication. We then identify the most suitable data replication solutions under various mobility models. More specifically, we study intra-group and inter-group data replication and comprehensively quantify the effect of mobility on data replication based on several new metrics such as

the temporal and spatial data access delay and data availability. Our experimental results illustrate that different replication schemes show quite different features on node cooperation, and thus achieve different data access performance.

The remainder of this paper is organized as follows. In Section II, we introduce three typical tactical models. In Section III, we study the data replication issues for tactical networks, proposing a best-location intra-group data replication and three inter-group data replication schemes. The newly proposed performance metrics are introduced in Section IV. Section V reports the evaluation results of how different data replication schemes perform under various mobility models. Finally, Section VI concludes the paper.

II. TACTICAL MOBILITY MODEL

The users of tactical networks are military forces. These forces are highly structured (e.g., squads, platoons, etc.) and their actions are strictly organized. The nodes in the network do not move randomly. There is one leader or a group of leaders which tells other nodes where and how to move or in which area to work. This implies group mobility [7, 8]. Furthermore, as tactical scenarios take place in areas of conflict, obstacles such as trenches or traps might be encountered. They will have a certain impact on movements. Therefore, the mobility models derived from tactical networks are quite different from the traditional models used for conventional mobile networks where nodes can move randomly. In this paper, we mainly focus on three popular tactical maneuvers in battlefields: *Repeated Traversal (RT)*, *Bounding Overwatch (BO)* and *Pincer (PI)*.

A. Repeated Traversal (RT)

The objective of repeated traversal is to maximize the security of forces moving through an unknown, highly hostile terrain. When forces navigate through an unexamined terrain that is suspected to contain ambushes and traps, a common practice is to traverse through the area along the same path one squad after another. The 'following' squads depart along the same path, only when the 'leading' squad has arrived at a target of confirmed security and has communicated information about the terrain to the other squads. Leading squads may also have to rest and wait for the following squads to provide supply and back up. If an 'unfriendly event' is observed by the leading squad (such as a trap or bomb explosion), it communicates the same to the following squad so that the following squad avoids that route.

B. Bounding Overwatch (BO)

Bounding Overwatch is designed to covertly capture the enemy (usually an enemy post or a small enemy troop) while keeping the enemy engaged in other activity. Two or more troops work in collaboration such that one (or more) troop(s) covers by firing at the enemy and keeps them engaged while the other

troop(s) advances to a new position, closer to the enemy. The firing troop(s) always halts at a position from where it is able to fire at the enemy and is behind an obstacle covering it. Once the advancing troop reaches its new target, it begins to fire from there while one of the troops that were firing previously advances to a new position. This pattern continues till one of the troops is able to reach the enemy post.

C. Pincer (PI)

The third military tactic we consider is known as pincer mobility. The objective of pincer is to surround the enemy troop from two or more directions, making it very easy to destroy them. This movement is a flanking attack from all sides of the enemy troops. A flanking attack is an attack on the sides of an opposing force. More units of the attacking force can move in from the front, rear, left and right sides, and result in encircling the enemy. To avoid being a victim of this a group of nodes can ensure that it is flanked by difficult terrain on different sides (eg, river, marshy land etc.).

III. DATA REPLICATION

Based on the tactical mobility models, we study the data replication problem in this section and present several data replication solutions that will be used in the evaluation. We start by introducing the system model.

A. System Model

We assume there are m nodes in the network. The nodes are denoted by $N = \{n_1, n_2, \dots, n_m\}$, where $n_k (k = 1, \dots, m)$ is the node identifier. The communication range of each mobile node is represented by a circle with radius R . When two nodes move out of their communication range, the link between them will fail. The network can be partitioned due to the limitations of the communication range and link failure. As mentioned in Section II, in tactical networks, nodes show group mobility pattern. We use $g_i (i = 1, \dots, j)$ as the group identifier.

There are p different data items in the network. The set of data items is denoted by $D = \{d_1, d_2, \dots, d_p\}$, where $d_k (k = 1, \dots, p)$ is a data identifier. Each mobile node maintains some amount of data locally. For simplicity, we assume that data are not updated, and similar techniques used in [9] and [10] can be applied to extend the proposed scheme to handle data update or data consistency issues. These data items may be replicated to other nodes based on some data replication algorithm. Because of limited memory (or disk) size, each mobile node can only host $B (B < p)$ replicas including its original data. When a mobile node n_i needs to access a data item d_j , n_i first searches its local memory. If n_i cannot find a copy of d_j in the local memory, n_i communicates with its reachable nodes (through one-hop or multi-hop links in its partition) to get d_j . If the requesting node cannot communicate with any of the nodes that have d_j , d_j is considered to be not accessible to n_i .

B. Data Replication Schemes

Since all tactical mobility patterns usually show group mobility, we take the group mobility into account when we design data replication solutions. For mobile nodes in the same group, because they have similar mobility and reliable connections, they can collaborate to replicate data and share data with each other

by applying a good intra-group data replication scheme. Different mobile groups may or may not have similar mobility. Therefore, different inter-group data replication schemes will lead to different data access performance for nodes. In this subsection, we first present a new intra-group data replication scheme for nodes in the same group to optimize data access, and then we introduce three data replication schemes of different collaboration degree for inter-group data replication.

B.1 Best-Location Intra-Group Data Replication

The basic idea behind the best-location data replication algorithm is to find the best location (node) to place each data replica so that the overall data redundancy and data access cost within the group are minimized. The algorithm is outlined below:

1. First, the average access probability of each data item within the group is calculated. The average access probability of a data item d_i is defined as

$$A_i = \frac{\sum_{j=1}^l a_{ji}}{l} \quad (1)$$

where a_{ji} denotes the access probability of node n_j to access data d_i , which can be estimated by the node itself, and l is the total number of nodes in the group. We assume that group size is known by nodes, and the techniques used in [11, 12] can be used to estimate the group size with low overhead.

2. Let v_{kj} denote the expected access cost if a copy of data d_j is placed at node n_k . Then v_{kj} is calculated as

$$v_{kj} = \sum_{i=1}^l a_{ij} \times c_{ijk} \quad (2)$$

where c_{ijk} is the expected access cost in terms of hops for node n_i to access data d_j placed at node n_k .

3. For each data d_j , starting from the one with the highest A_j , a node i is identified such that

$$v_{ij} = \min\{v_{xj}\} \quad \forall \text{ node } n_x \in \text{group} \quad (3)$$

4. The selected node n_i represents the best location to store d_j , which minimizes the total cost to access d_j for the group. A copy of d_j is placed at node n_i . If the memory of node n_i is full, d_j is given to the node with the next lowest v_{ij} .

5. Step 3 and Step 4 are repeated for all data items in the order of their average access probability until the memory of all nodes in the group are filled.

Formally, the best-location data replication algorithm is presented in Algorithm 1.

Since no redundant replication takes place until every data item is replicated at least once, the best location algorithm can guarantee to have the largest number of different data items in the group. Furthermore, because the function for selecting the best node to place each data replication (i.e. Equation (3)) accounts for the access cost between the query node and the nearest replication node in the group, the best-location algorithm can also reduce the delay required to serve a query.

B.2 Inter-Group Data Replication

To study the effects of mobility on data replication, besides the intra-group data replication scheme, we use the following

Algorithm 1 : Best-Location Intra-Group Data Replication

```

1: Input:
2:    $l$ : the number of node in the group;
3:    $d$ : the number of available data items in the group;
4:    $a[][]$ : the 2-dimensional matrix that records the access probability of each
   node to each data;
5:    $c[][][]$ : the 3-dimensional matrix that records the access cost of each data
   between any two nodes in the group;
6: Variables:
7:    $A[]=0$ : the average access probability of each data;
8:    $v[][]=0$ : the total access cost of each data for all nodes in the group;
9: FOR each available data  $D_i$ 
10: FOR each node  $n_j$ 
11:    $A[i] = A[i] + a_{ij}$ ;
12:   FOR each node  $n_k$ 
13:      $v[k][j] = v[k][j] + a_{ij} \times c_{ijk}$ ;
14:   END FOR
15: END FOR
16: END FOR
17: WHILE there is memory available in the group
18:   Start from the  $d_i$  with the highest  $A[i]$ ;
19:   DO
20:     Find node  $n_j$ , s.t.  $v[j][i] = \min\{v[x][i]\}$  for all  $n_x$  in the group;
21:     Make a copy of data  $d_i$  at node  $n_j$ ;
22:     Start from the data with the next highest  $A[i]$ ;
23:   END DO
24: END WHILE

```

three representative data replication schemes with different degrees of collaboration for inter-group data replication.

Greedy Data Replication

The Greedy data replication is a naive data replication algorithm. In this algorithm, each group replicates its most frequently accessed data until the memory is full. More specifically, each group always replicates the data with the highest A_i . Since each group only takes its own data access pattern into account during data replication, this replication scheme is non-cooperative among different groups.

Pairing Cooperation Data Replication

Unlike the Greedy data replication, in the Pairing algorithm (e.g., the OTOO scheme in [13] and the DAFN scheme in [6]), each mobile group cooperates with one of its neighboring groups to decide which data to replicate. More specifically, each group pair g_i and g_j calculates a combined access frequency value to data item d_k at g_i and g_j , called CAF_{ij} , respectively. For example, for g_i :

$$CAF_{ij}(k) = A_{ik} + A_{jk} \times (1 - f_{ij}) \quad (4)$$

where f_{ij} represents the disconnecting probability between groups g_i and g_j . Similarly g_j calculates its combined access frequency. Each group sorts the data according to the CAF value and picks data items with the highest values to replicate in its memory until no more data items can be replicated. The data replication decision does not simply depend on the access frequency of one single group. It depends on the access frequency of the other pairing group and the partition probability between them.

Reliable Neighboring Data Replication

The Pairing scheme considers neighboring groups when making data replication choices. However, it still considers its own group access frequency as the most important factor and only cooperates with one neighboring node. As described in [13], the reliable Neighboring data replication scheme further increases the degree of cooperation and allows groups to replicate and

share data with multiple reliable neighboring groups. The replication decision is made depending on data access frequency and the link stability. More specifically, in this scheme, part of the group's memory is used to hold the most interesting data for itself and others are for its expected neighbors. The combined access frequency function for group g_i to data d_k in the Neighboring algorithm is defined as:

$$CAF_i(k) = \sum_{g_j \in nb(i)} A_{jk} \times (1 - f_{ij}) \quad (5)$$

where $nb(i)$ is the set that includes all reliable neighboring groups of g_i ; i.e., whose disconnecting probability to g_i is less than a threshold.

IV. EVALUATION METRICS

Based on the mobility model and data replication schemes, we define several metrics that represent the performance of different data replication solutions.

A. Metrics on Data Access Delay

A.1 Average Access Delay (\mathcal{D})

We use t_{ij} to denote the delay of the j th successfully served request of node n_i . The definition of this metric can be formalized as

$$\mathcal{D} = \frac{\sum_{i=1}^m \sum_{j=1}^{\mathcal{R}(i)} t_{ij}}{\sum_{i=1}^m \mathcal{R}(i)} \quad (6)$$

Here, $\mathcal{R}(i)$ is a function to return the number of requests initiated by node n_i during the experiment.

A.2 Temporal Distribution of Access Delay (\mathcal{D}_k)

Since different mobility models may lead to different deployment patterns of nodes at different time, we study the temporal distribution of data access delay at the k th time interval as \mathcal{D}_k .

$$\mathcal{D}_k = \frac{\sum_{i=1}^m \sum_{j=1}^{\mathcal{R}_k(i)} t_{ij}}{\sum_{i=1}^m \mathcal{R}_k(i)} \quad (7)$$

Similar to the definition of $\mathcal{R}(i)$, here, $\mathcal{R}_k(i)$ is a function to return the number of requests initiated by node n_i during the k th time interval.

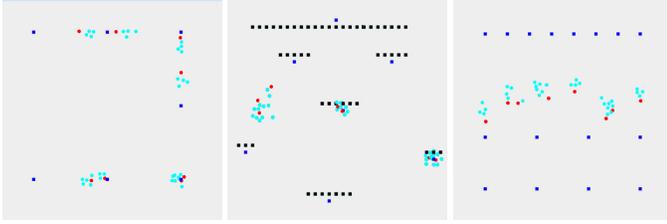
A.3 Spatial Distribution of Access Delay ($\mathcal{D}_{\langle h_x, h_y \rangle}$)

For the spatial distribution of data access delay, we divide the entire network area into $h \times h$ small subareas and compare the results in different subareas. The geographical distribution of access delay at subarea $\langle h_x, h_y \rangle$ is expressed by the following equation:

$$\mathcal{D}_{\langle h_x, h_y \rangle} = \sum_{i=1}^m \sum_{j=1}^{\mathcal{R}(i)} \mathcal{L}(t_{ij}, \langle h_x, h_y \rangle) \quad (8)$$

where $\mathcal{L}(t_{ij}, \langle h_x, h_y \rangle)$ is a function that returns if the request takes place in the subarea $\langle h_x, h_y \rangle$.

$$\mathcal{L}(t_{ij}, \langle h_x, h_y \rangle) = \begin{cases} t_{ij} & \text{if the } j\text{th request of node } n_i \\ & \text{initiated in the subarea } \langle h_x, h_y \rangle; \\ 0 & \text{else.} \end{cases}$$



(a)Repeated Traversal Scenario (b)Bounding Overwatch Scenario (c)Pincer Scenario

Fig. 1. Screen shots of simulation scenarios

B. Metrics on Data Availability

B.1 Average Availability (\mathcal{A})

Average availability is the average probability that the query can be served successfully. Similarly, we use a binary variable s_{ij} to denote if the j th request of node n_i is satisfied or not, the definition of this metric can be formalized as

$$\mathcal{A} = \frac{\sum_{i=1}^m \sum_{j=1}^{\mathcal{R}(i)} s_{ij}}{\sum_{i=1}^m \mathcal{R}(i)} \quad (9)$$

where

$$s_{ij} = \begin{cases} 1 & \text{the } j\text{th request of node } n_i \text{ is satisfied;} \\ 0 & \text{else.} \end{cases}$$

B.2 Temporal Distribution of Availability (\mathcal{A}_k)

Similarly, we use \mathcal{A}_k to represent the temporal distribution of data availability at time interval k .

$$\mathcal{A}_k = \frac{\sum_{i=1}^m \sum_{j=1}^{\mathcal{R}_k(i)} s_{ij}}{\sum_{i=1}^m \mathcal{R}_k(i)} \quad (10)$$

B.3 Spatial Distribution of Availability ($\mathcal{A}_{\langle h_x, h_y \rangle}$)

Similar to the definition of spatial distribution of access delay, the spatial distribution of availability is represented by the following equation:

$$\mathcal{A}_{\langle h_x, h_y \rangle} = \frac{\sum_{i=1}^m \sum_{j=1}^{\mathcal{R}(i)} \mathcal{L}(s_{ij}, \langle h_x, h_y \rangle)}{\sum_{i=1}^m \mathcal{R}(i)} \quad (11)$$

where $\mathcal{L}(t_{ij}, \langle h_x, h_y \rangle)$ is a function that returns if the request takes place in the subarea $\langle h_x, h_y \rangle$.

$$\mathcal{L}(t_{ij}, \langle h_x, h_y \rangle) = \begin{cases} s_{ij} & \text{if the } j\text{th request of node } n_i \text{ is} \\ & \text{initiated in the subarea } \langle h_x, h_y \rangle; \\ 0 & \text{else.} \end{cases}$$

V. EXPERIMENTS

To evaluate the performance of the data replication schemes, we developed a simulator based on CSIM 19 [14]. In this section, we measure the performance of data replication schemes under three tactical mobility models we introduced above.

A. Simulation Settings

There are 8 groups of nodes in a $3000m \times 3000m$ square area. Each group has 5 nodes. All nodes move based on a mobility model. Figure 1(a), (b), and (c) show the screenshot of three mobility models in our simulations. In these figures, we use dots to denote the moving nodes (red as the group leader and light blue as the group member). The small dark blue

TABLE I
PARAMETER CONFIGURATION

Parameter	Symbol	Value	Range
Number of nodes	m	40	
Group size	S	5 nodes per group	
Communication range	C	100m	
Number of data	n	100	
Memory Size	B	5 data items	
Zipf access	θ	0.8	

TABLE II
ACCESS DELAY

	Repeated Traversal	Bounding Overwatch	Pincer
Greedy	1.822	1.963	1.695
Pairing	1.826	2.019	1.704
Neighboring	1.832	1.974	1.727
Naive-Greedy	1.467	1.523	1.436

squares are used to represent the targets and the small black squares are used to represent trenches. For the Repeated Traversal (Figure 1(a)), we have 7 targets which are located at the coordinates (500,500), (1500,500), (2500,500), (2500,1500), (2500,2500), (1500, 2500), (500, 2500). All squads are starting from (500,500), and move towards the next target. For Bounding Overwatch (Figure 1(b)), 8 squads are organized into 3 troops. The first troop is formed by 3 squads of nodes moving following the left route. The second troop consists of 2 squads moving centrally. The last 3 squads belong to the third troop. They cover the target from the right route. In Pincer (Figure 1(c)), 8 squads form 2 troops, each with 4 squads, to cover two different enemy forces. Table I further summarizes the parameters and their values used in the experiments.

B. Average Delay and Average Data Availability

In this subsection, we study average delay and average data availability of the data replication schemes under three mobility models. By combining Table II and Table III, we can see that all group based data replication schemes (Greedy, Pairing, and Neighboring) outperform the Naive-Greedy scheme significantly. Note Naive-Greedy scheme is the data replication scheme that each node replicates data based on its own interest and it is not group-based. Although the Naive-Greedy scheme has relatively shorter delay, its data availability is much lower than other schemes. The benefit of the group based data replications comes from the fact that they take the group mobility feature into account and cooperatively replicate data among nodes in the same group by using our proposed best-location intra-platoon data replication schemes. The data deployment within each group is more appropriate, which helps achieve relatively short data access delay and high data availability.

Furthermore, by comparing the performance of three group based data replication schemes, we can see that with Repeated Traversal, all three schemes have similar data access delay but the Pairing scheme has a much higher data availability than other

TABLE III
DATA AVAILABILITY

	Repeated Traversal	Bounding Overwatch	Pincer
Greedy	0.604	0.617	0.632
Pairing	0.826	0.823	0.517
Neighboring	0.603	0.893	0.463
Naive-Greedy	0.325	0.332	0.323

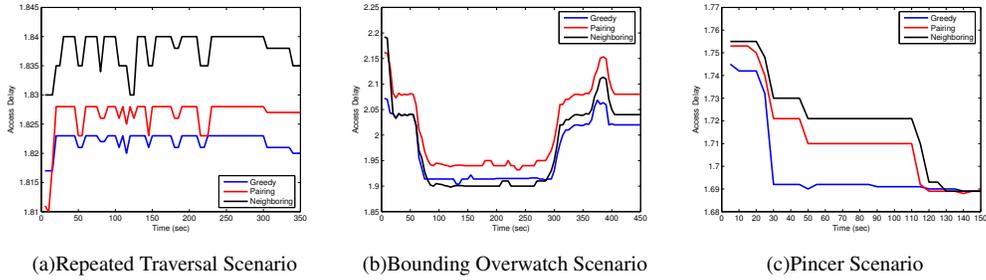


Fig. 2. Temporal results of access delay

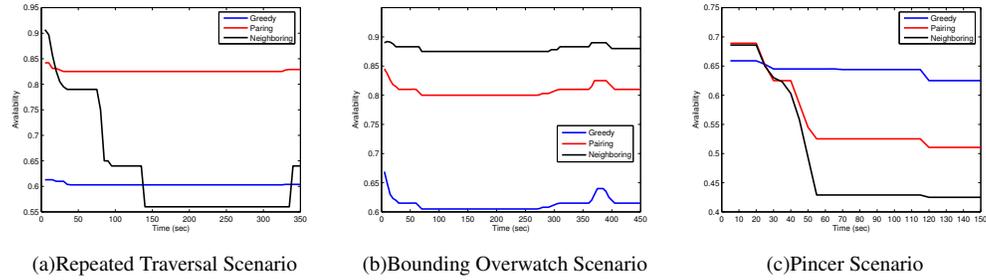


Fig. 3. Temporal results of availability

two. This phenomenon is due to the node movement pattern of Repeated Traversal. In Repeated Traversal, squads are organized into pairs and each pair consists of two squads of nodes, one leading squad and one following squad. Therefore, as we can see from Table II and Table III, the cooperative data replication will be helpful to achieve a better performance. However, we notice that because the Neighboring scheme replicates data among all neighboring squads, but some neighboring squads may not belong to the same pair, when squads start to move at different times and a network partition occurs, some nodes may not be able to access the data items which were cooperatively replicated in the other squads. Therefore, with Repeated Traversal, the cooperation of the Neighboring scheme negatively affects the performance.

Similarly, with Bounding Overwatch, several squads form a troop and the squads of the same troop move following a similar trajectory. In this case, the cooperation among squads of the same troop can also help improve data access performance. Moreover, unlike the case of Repeated Traversal, with Bounding Overwatch, each troop can have more than two squads. Therefore, the Neighboring data replication scheme works the best with Bounding Overwatch.

With Pincer, neighboring squads may not move towards the same target. Hence, considering the unreliable connectivity between neighboring squads, cooperative data replication schemes such as Pairing and Neighboring will negatively affect data access performance when network partitions occur. As a result, the non-cooperative Greedy inter-group data replication scheme performs the best with the relatively high data availability and short access delay.

C. Temporal Results

Figure 2 and Figure 3 show the temporal results of data access delay and data availability of different data replication schemes with different mobility models. Here, we apply the best-location intra-group data replication by default within each squad and thus we can compare the impact of different mobility models

based on different inter-group data replication schemes.

From Figure 2(a) and Figure 3(a) we can see that with Repeated Traversal, at the beginning of the experiments all three data replication schemes show relatively short access delay and high data availability. This is because at the initial stage, all nodes stay together. When some squads start to move, data access delay becomes longer since the distance between nodes increases and some single hop data accesses becomes multi-hop. Further, as network partition occurs, as shown in Figure 3(b), data availability of the Neighboring data replication decreases dramatically. This situation is due to the fact that the Neighboring scheme seeks to cooperatively replicate data among neighboring squads. In this case, when some squads become unreachable, many data items are not able to be accessed. The Pairing scheme, however, only cooperatively replicates data between the reliable pair of groups, which can still provide high data availability even if network partitions occur with Repeated Traversal.

Figure 2(b) and Figure 3(b) show the temporal results of different data replication schemes with Bounding Overwatch. With Bounding Overwatch, because multiple squads move together as an attacking troop, the Neighboring scheme achieves the best performance. More specifically, from the figures we can see that data availability of all schemes decreases at the start (e.g., 5th second) and around the 50th second. The decrease is due to the network partitions that occur at these two stages. For similar reasons, availability increases around the 300th second and the 380th second because in this period separate neighborhoods merge, making more data items accessible to all nodes. The corresponding increase and decrease of access delay occurs because we consider only the successful queries when calculating access delay. When network partitions occur all queries are served locally, so access delay decreases.

The temporal results of the Pincer are shown in Figure 2(c) and Figure 3(c). Because neighboring squads do not have mobility similarities, the cooperation based data replication schemes such as Pairing and Neighboring performs worse than Greedy.

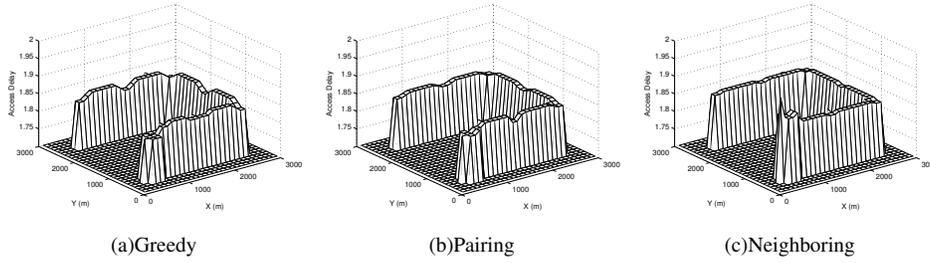


Fig. 4. Access delay of Repeated Traversal Scenario

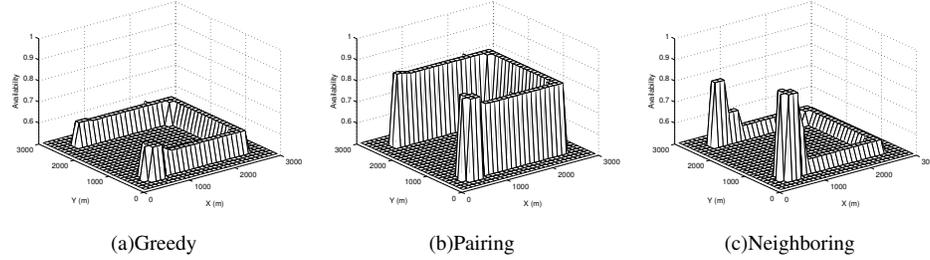


Fig. 5. Availability of Repeated Traversal Scenario

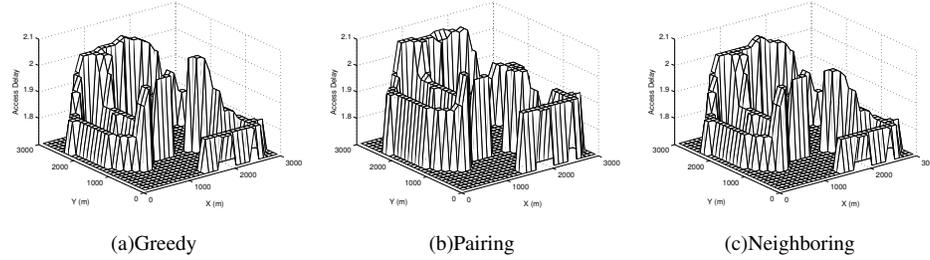


Fig. 6. Access delay of Bounding Overwatch Scenario

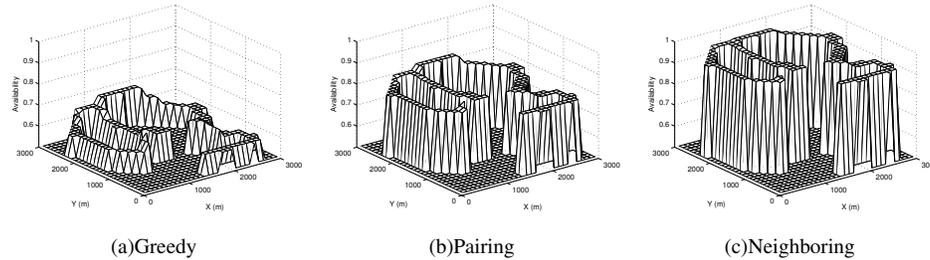


Fig. 7. Availability of Bounding Overwatch Scenario

D. Spatial Results

For a more comprehensive comparison of data replication schemes in different mobility models, we also quantify the access delay and data availability from the spatial perspective.

From Figure 4 and Figure 5, we can see that at the target areas, all schemes have relatively short access delay. This is because these areas usually have high node densities and nodes are much closer to each other. Here, we notice that the Neighboring scheme has higher access delay and data availability at the beginning area. This is due to the fact that Neighboring data replication tries to replicate data among all neighboring squads. In this case, more data items can be replicated at the beginning but some requests may be served by the nodes in the multi-hop distance. Later, as squads move and network partition happens, data availability decreases significantly. Furthermore, since only the successfully served requests are counted in the calculation of the access delay, the overall access delay of other areas de-

creases in the Neighboring scheme.

Similar results can be observed in Figure 6, Figure 7, Figure 8, and Figure 9, which demonstrate the performance of different data replication schemes with Bounding Overwatch and Pincer. Higher nodes density results in longer data access delay but higher data availability.

E. Discussions

In this section, we summarize the experimental results and identify the most suitable data replication algorithms under various mobility models.

Repeated Traversal: With Repeated Traversal, nodes are organized as squads. Each attacking squad has one leading squad to probe the target, and the connectivity between the following squad and its leading squad is reliable. Therefore, when designing a data replication scheme, it is more appropriate for squads to cooperatively replicate data with their closely connected pair-

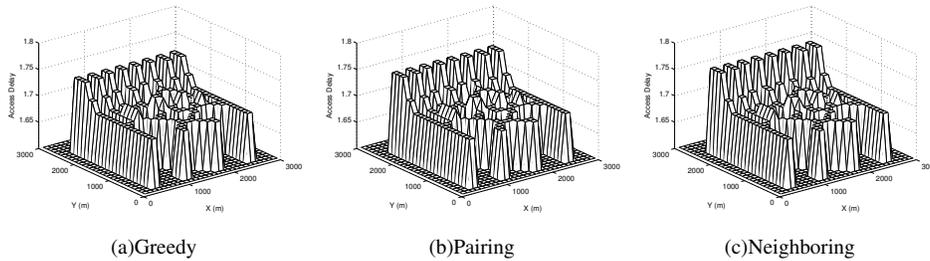


Fig. 8. Access delay of Pincer Scenario

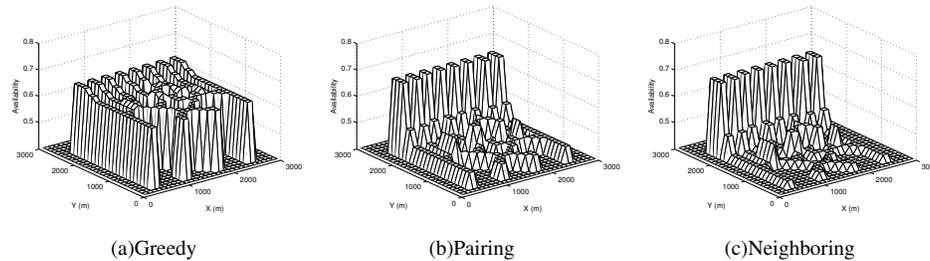


Fig. 9. Availability of Pincer Scenario

ing squads and the replication should not rely on data sharing with a large number of squads. This also explains why the Pairing scheme performs better for Repeated Traversal.

Bounding Overwatch: Bounding Overwatch has several interesting features due to its restricted mobility. First, Bounding Overwatch shows group mobility, which means several nodes forming a squad and moving together. Therefore, similar to the other two mobility models, intra-group data replication scheme that takes group mobility into consideration can help improve data access. Second, from the squad point of view, some attacking groups consist of more than two squads of nodes. For this reason, the Neighboring data replication that cooperatively replicates and shares data among all neighboring squads that have reliable connectivity shows the best performance.

Pincer: In Pincer, close squads may attack different targets. Therefore, neighboring squads do not show any reliable connections with each other, and hence the squad partition rate is high. Thus, it may not be good to share data with others and the non-cooperative Greedy scheme may be the best choice.

Furthermore, by combing all results presented above, we can observe that data access performance is also related to the node density of the network. For example, in an area where node density is high and nodes are very close to each other, data access delay is usually short and data availability is relatively high. However, in an area where nodes are sparsely connected, data access delay is long and data availability is low. This is because in a dense network, nodes have more directly connected neighbors and have easier access to their data of interest, but when node density decreases, the query node may have to search other nodes at a multihop distance to get the data or may not even be able to find the data, which negatively affects the data access performance.

VI. CONCLUSION

In this paper, we thoroughly investigated how to efficiently replicate data in a tactical network. We proposed a best-location data replication scheme for intra-group data replication which aims to minimize the overall data redundancy and data access

delay. The study is based on several metrics which are not limited to the average access delay and data availability, but include temporal and spatial analysis of these values. Through extensive experiments, we quantify the effects of three typical mobility models on data replication schemes from various perspectives, and identify the most suitable inter-group data replication scheme under various mobility models.

To the best of our knowledge, this is the first work that explores the relationship between node mobility and data replication in mobile tactical networks. We believe that the experimental results and knowledge obtained from the results are very useful for researchers to design various solutions for data sharing and replication on these typical mobility models.

REFERENCES

- [1] M. Rollo and A. Komenda. Mobility model for tactical networks. *HoloMAS '09*, pages 254–265, 2009.
- [2] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wirel. Commun. Mob. Comp.*, 2(5):483–502, 2002.
- [3] S. Yoon, S. Lee, and Y. Ko. Reliable dual-path geocasting for tactical ad hoc networks. *MILCOM*, pages 258–264, 2009.
- [4] N. Lu and C. Nelson. Cooperative team communications. *MILCOM*, pages 1991–1996, 2009.
- [5] C. E. Fossa and T. Macdonald. Internetworking tactical manets. *MILCOM*, pages 1034–1039, 2010.
- [6] T. Hara and S. K. Madria. Data replication for improving data accessibility in ad hoc networks. *IEEE Transactions on Mobile Computing*, 5(11):1515–1532, 2006.
- [7] S. Kumar, S. C. Sharma, and B. Suman. Mobility metrics based classification and analysis of mobility model for tactical network. *International Journal of Next-Generation Networks (IJNGN)*, 2(4):39–51, 2010.
- [8] B. Adamson and J. P. Macker. Reliable messaging for tactical group communication. *MILCOM*, pages 1439–1444, 2010.
- [9] T. Hara and S. Madria. Consistency management strategies for data replication in mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 8(7):950–967, 2009.
- [10] J. Cao, Y. Zhang, G. Cao, and L. Xie. Data consistency for cooperative caching in mobile environments. *IEEE Computer*, 40(4):60–66, 2007.
- [11] Y. Zhang, J. Zhao, G. Cao, and C. Das. On interest locality in content-based routing for largescale manets. *IEEE MASS*, pages 178–187, 2009.
- [12] Y. Zhang and G. Cao. V-pada: Vehicle-platoon-aware data access in vanets. *IEEE Transactions on Vehicular Technology*, 60(5):2326–2339, 2011.
- [13] L. Yin and G. Cao. Balancing the tradeoffs between data accessibility and query delay in ad hoc networks. *IEEE International Symposium on Reliable Distributed Systems*, pages 289–298, 2004.
- [14] H. Schwetman. Csim19: a powerful tool for building system models. *The 33rd Conference on Winter Simulation*, pages 250–255, 2001.