# Comparing and Evaluating Lightweight Solutions for Replica Dissemination and Retrieval in Dense MANETs

Paolo Bellavista, Antonio Corradi, Eugenio Magistretti
*Dip. Elettronica, Informatica e Sistemistica - Università di Bologna*
*Viale Risorgimento, 2 - 40136 Bologna - ITALY*
*Phone: +39-051-2093001; Fax: +39-051-2093073*
*{pbellavista, acorradi, emagistretti}@deis.unibo.it*

## Abstract

*There is an emerging market interest in service provisioning over dense Mobile Ad-hoc NETworks (MANETs), i.e., limited spatial regions, such as shopping malls, airports, and university campuses, where a high number of mobile wireless peers can autonomously cooperate without exploiting statically deployed network infrastructures. We claim that it is possible to exploit the high node population of dense MANETs to simplify the replication of common interest resources, in order to increase availability notwithstanding unpredictable node exits from dense regions. To this purpose, we have developed the REDMAN middleware that supports the lightweight and dense MANET-specific management, dissemination and retrieval of replicas of data/service components. In particular, the paper focuses on the presentation of different solutions for replica retrieval and for dissemination of replica placement information. We have compared and quantitatively evaluated the presented solutions by considering their ability to retrieve available replicas and their communication overhead. The original SID solution has demonstrated to outperform the others in dense MANETs and has been integrated in the REDMAN prototype.*

## 1. Introduction

The mass market of Wi-Fi/Bluetooth-enabled portable devices suggests novel service deployment scenarios where there are no constraints on device mobility and distributed applications should result from the impromptu opportunistic collaboration of wireless peers. In these scenarios, not only wireless nodes should privilege the access to resources available in their proximity, without requiring any static knowledge about their execution environment, but also resources should be permanently accessible, independently of unpredictable node movements, network disconnections, and battery shortage [1].

In particular, we are interested in addressing a specific deployment scenario of increasing relevance, called dense Mobile Ad hoc NETworks (MANETs) in the following. We use the term *dense MANET* to indicate an ad-hoc infrastructure-free wireless network that:

- includes a large number of wireless devices located in a relatively small area at the same time, e.g., as it will probably happen in the near future in shopping malls, exhibition centers, conference buildings, and university campuses;
- has a node density (the average number of wireless nodes at single-hop distance from any dense MANET participant) that is roughly invariant during relatively long time intervals.

The primary idea behind our approach is to provide a lightweight middleware solution to instantiate, disseminate, and manage replicas of common-interest resources (shared data and middleware/service component code) among wireless nodes in the dense MANET, so that any client could access at least one replica in its vicinity at any moment and anywhere while in the dense region. Resource replicas should be effectively placed to enable efficient retrieval solu-

tions, while imposing limited communication overhead both for replica placement and retrieval. Resource accessibility should be maintained notwithstanding the unpredictable movements of wireless devices (with their hosted replicas) inside/outside the dense MANET and their temporary loss of network connectivity.

Due to the complete lack of a static support infrastructure, we claim that resource replica management is a very hard task to perform in an effective and lightweight way when dealing with general-purpose MANETs and when dealing with strict consistency requirements over wide-scale provisioning environments [1]. For these reasons, we have decided to specifically focus on dense MANETs, where the assumption of relatively high and constant node density permits to exclude the possibility of dynamic network partitioning and sub-network merging. However, limiting the investigation to dense MANETs is not too restrictive: it is valid in most environments where there is commercial interest in provisioning distributed services for entertainment, public utility, and advertisement.

Given the above considerations, we have worked to design and implement a middleware, called REDMAN (**RE**plication in **D**ense **MAN**ETs), that transparently disseminates, manages, and retrieves resource replicas among cooperating nodes in dense MANETs [2, 3]. Transparently from the point of view of application developers, the REDMAN middleware works to maintain, with a lightweight and lazily consistent approach, the desired resource replication degree within the dense MANET, independently of possible exits of replica-hosting nodes from the dense region. REDMAN addresses dense MANET challenging issues in a highly decentralized way via original protocols specifically designed for dense MANETs. Moreover, REDMAN does not impose the availability of any Global Positioning System (GPS) equipment installed on participating devices, thus enabling also the cooperation of terminals with very strict constraints on local resources, especially on battery power consumption.

At the moment, REDMAN addresses replica management for read-only resources (files with still/moving images, audio tracks, hypertexts) or, anyway, with no consistency requirements in the case of replica modification. Let us observe that this kind of replica dissemination is suitable not only to inject entertainment/advertisement-related data in dense MANETs, but also to dynamically distribute the code of needed support/application components, e.g., driver updates, format-specific multimedia players, and game clients.

The rest of the paper is organized as follows. After giving a general overview of the REDMAN architecture and features (Section 2), necessary for the full understanding of the following proposals, the paper focuses on a very crucial aspect of replica management in dense MANETs: how to disseminate resource replicas and information about replica placement, so to enable the rapid and lightweight retrieval of requested resources at provision time, by avoiding any point of centralization. In particular, Section 3 presents REDMAN replica distribution strategy, while Section 4 compares different original solutions for the dissemination of replica placement information and for replica retrieval, specifically designed for dense MANETs. Experimental results (Section 5) show that the original Straight Information about replica placement Dissemination (SID) solution outperforms the others by achieving good search accuracy with no points of centralization and a limited message overhead, also in wide-scale environments and in presence of mobility. Related work, conclusions, and on-going research directions end the paper.

## 2. The REDMAN Middleware

REDMAN addresses the issue of disseminating replicas of resources of common interest in dense MANETs, independently of unexpected node exit from the dense region. More formally, a dense MANET is defined as the set of MANET nodes $DM(n) = \{d_0, ..., d_{N-1}\}$, where i) $\forall j \in [0, N-1]$ $d_j$ has at least $n$ neighbors at single-hop distance, and ii) the spatial node density, in the area where $DM(n)$ nodes are, is almost constant with regards to time. Given a resource with a desired replication degree $deg$, REDMAN is in charge of instantiating and distributing $deg$ replicas of it, and of maintaining the $deg$ replication degree notwithstanding the changes in the composition of the $DM(n)$ set. Service developers can simply benefit from REDMAN replica management facilities via very little modifications on their application code: they only need to include Resource Description Framework (RDF) [4] descriptors for resources to share, by specifying the suggested replication degree, and to exploit the REDMAN API to command resource replication and retrieval.

Figure 1 depicts the REDMAN modular architecture. On top of the lower layer of mechanisms (Dense MANET Configuration - DMC) for dense MANET participant identification and decentralized election, REDMAN can dynamically assign two different roles to a subset of nodes in the dense region:

- *delegates* that host resource replicas and are in charge of Replica Distribution (RD) and Replica Retrieval (RR);

- *managers* that are dynamically elected and responsible for determining and maintaining a suitable replication degree for associated resources (Replica Degree Maintenance - RDM). They can either verify the number of replicas actually available in the dense region or be notified of inconsistencies in the desired replication degree by RDM components on delegate nodes.

In the following, the paper focuses on the RD and RR components. Additional details on DMC and RDM can be found in [2, 3].
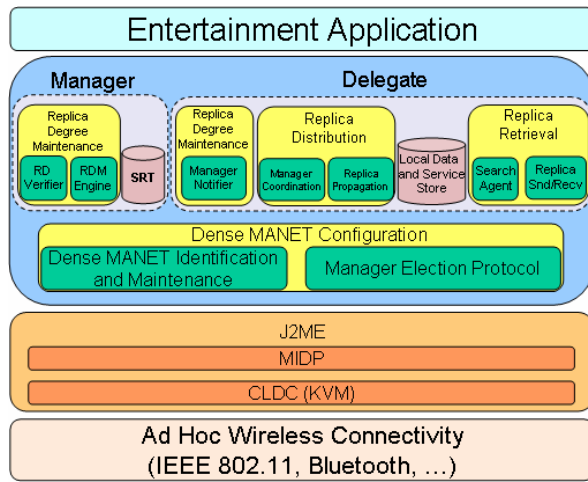


**Figure 1. The REDMAN architecture.**

## 3. Replica Distribution (RD)

RD is responsible for the lightweight dissemination of replicas on MANET nodes, also with the goal to enable the effective lightweight resource retrieval at provision time (see Section 4.2). When a delegate enters the dense MANET, it communicates the RDF description of its resources to the manager that decides the replication degree (*deg*) of each resource on the basis of the provided descriptor and of other external indicators such as the estimated number of nodes in the dense region. Then, it delegates the resource owner for the actual implementation of the replica distribution process. The main guideline of the RD protocol is to disseminate resource replicas on nodes at *r*-hop distance along a constant direction. When a delegate has to replicate one of its resources, it sends a replication packet specifying the number of replicas still required and the desired *r*-hop distance between replicas. The replication packet is propagated on nodes placed along an approximately straight line with a fixed direction.

Let us observe that REDMAN does not require dense MANET participants to be GPS-equipped and

exploits lightweight heuristic-based estimations, specific for dense MANETs, to determine constant directions. Roughly speaking, the solution guideline is that a node determines its successor by choosing, among its neighbors (the nodes at single-hop distance from it), the one sharing fewer neighbors with its predecessor. To this purpose, RD locally broadcasts the neighbor list of its predecessor to all neighbors; only the neighbors sharing with the predecessor a number of neighbors lower than a threshold reply. This determines a roughly constant direction if the node density is almost uniform in the dense MANET. When a replication packet reaches a node at *r*-hop distance from a replica, that node becomes a delegate; the new delegate reiterates the process by decreasing the number of requested replicas.

## 4. Replica Retrieval (RR)

RR aims at enabling clients to effectively find their requested resources at provision time on the basis of resource RDF descriptions, i.e., to dynamically determine the IP address of one node hosting a requested matching resource and the unique name of the resource on that node. Resource retrieval is a hard task in MANETs, where a static infrastructure is not continuously available, thus preventing the usage of a fixed centralized lookup service known by all participants [5]. The usage of a single centralized repository with replica placement information is not viable: i) a large number of requests could overwhelm the single point of centralization; ii) the repository would represent a single point of failure; iii) the repository should be updated with strict consistency requirements not to hinder resource accessibility.

In several cases, it makes sense to improve the RR performance by paying the overhead of disseminating Information about Replica Placement (IRP) to a suitably chosen subset of nodes belonging to the dense MANET. In the following, the section first analyzes main advantages and drawbacks of some common retrieval solutions; then, it presents and evaluates the original REDMAN RR strategy, called SID, designed and implemented to effectively fit the dual REDMAN replica distribution solution.

### 4.1. An Overview of Possible RR Strategies

We propose to consider three main factors to determine the effectiveness of RR strategies:
- the overhead imposed in terms of both memory required to maintain IRPs and messages exchanged to retrieve the requested resources;

- the scalability when applied to large deployment environments;
- the accuracy, i.e., the found/searched resource ratio in the finite time interval spent for retrieval.

Different RR strategies can decide different trade-offs among these factors. In particular, the memory and network overhead imposed by IRP dissemination is often traded against the overhead required at provision time for resource discovery (the growth of IRP diffusion costs usually corresponds to a decrease of runtime retrieval costs). Therefore, the choice of the optimal RR strategy depends on the characteristics of supported applications and of deployment scenarios, e.g., on the expected ratio between searches and replica instantiations, the frequency with which replicas leave/enter the dense MANET, the time requirements for resource discovery, and the size of retrieval messages and IRPs.

The most intuitive and simple RR strategies are flooding-based. A first possible solution, which we will call IRP Flooding (IF) in the following, could establish that delegates disseminate IRPs about all their hosted resources to all nodes in the dense MANET (*flooding of IRP messages*). However, "the cost of making sure that everyone knows about everything is prohibitive" in MANET environments, mainly for scalability reasons [6]. In fact, to maintain an eager-consistent up-to-date view of IRPs, any delegate with a changing set of hosted resources should overburden the network with continuous IRP update message flooding; moreover, also the memory required to locally maintain IRPs of all replicas in the dense region could be unsustainable.

A second possible flooding-based solution (Query Flooding - QF) could specify that delegates do not have to diffuse any IRP; message flooding is necessary in the search phase where all nodes are exhaustively explored to look for requested resources (*flooding of search messages*). QF makes sense only if the number of RR searches is very limited and it is not worth paying the overhead for diffusing IRPs. Moreover, if the frequency of node entrances/exits in/out the dense MANET is very high, IRPs tend to become stale very soon, and the advantages of IF is significantly reduced.

### 4.2. k-hop Distance IRP Dissemination

Our research activity has focused on investigating novel RR strategies to avoid message flooding during the search phase by distributing IRPs only to a subset of nodes in the dense MANET. An original solution investigated, called $k$-hop Distance IRP Dissemination ($k$-DID), specifies to place IRPs only on nodes positioned at fixed $k$-hop distance the ones from the others.

In other words, i) the IRP related to a specified resource should be placed at exactly $k$ hops from at least another copy of the same IRP; ii) there should not be a path shorter than $k$ hops between two copies of the same IRP; and iii) IRPs should be distributed over the whole network so that each node is at most at $k$ hops from the IRP of any replicated resource. When adopting $k$-DID, IRP retrieval (and consequently replica discovery) only requires to explore the nodes situated, on average, at ($k/2$)-hop distance from the searcher. Let us observe that $k$-DID improves the solution in [7], where placement information similar to REDMAN IRPs is duplicated on all nodes located at fixed distance from resource-hosting nodes; in [7] IRP nodes could also be at single-hop distance the one from the other.

We have carefully investigated how $k$-DID could be effectively implemented in a lightweight way. Let us preliminary note that an IRP distribution strategy based on a single network flooding is infeasible, since it cannot determine which nodes at $k$-hop distance from an IRP-owning node are also at $k$-hop distance the one from the other. To practically present our $k$-DID implementation, suppose a delegate is willing to diffuse IRPs of its resources at $k$-hop distance. $k$-DID executes the following steps:

1. the delegate prevents its neighbors distant less than $k$ hops to host an IRP copy by flooding a denial message with $TTL=k$. Nodes receiving those messages with $TTL>0$ change their state to unavailable to reflect their unavailability for IRP hosting;
2. the delegate sends an IRP copy and assigns its initial role of IRP distributor to one of the nodes, identified at step 1, that are placed exactly at $k$-hop distance and whose state is free;
3. steps 1 and 2 are reiterated until the IRP distributor cannot identify any free node at $k$-hop distance;
4. if a free potential IRP distributor is found, then IRP dissemination goes on from step 1. Otherwise, a further backtracking step 4 is done, until the initial resource delegate is reached, thus ending the protocol.

Let us say that $N$ is the number of nodes belonging to the network and $F(k)$ the average number of nodes belonging to a $k$-hop-diameter circle included in the dense MANET. $k$-DID imposes a relevant overhead in terms of messages sent for distributing IRPs, mainly depending on $N$ and $k$. However, the number of nodes with IRPs is low (about one node every $F(k/2)$ ones) and the search message overhead is very low. In fact, each client expects to find a replica of the requested resource by querying all nodes within its surrounding

*k/2* hops. However, *k*-DID hardly scales in presence of node mobility since it is difficult to preserve the validity of the three above constraints without imposing heavy communication-intensive maintenance protocols for IRP distribution.

## 4.3. REDMAN SID

We have deeply investigated the performance of the *k*-DID RR strategy (see Section 5) and found it does not well fit the addressed dense MANET deployment scenario, mainly due to its excessive cost in both IRP diffusion and IRP updates in mobile environments. Therefore, we have decided to design, thoroughly evaluate, and then integrate in REDMAN an alternative original RR solution, called Straight IRP Dissemination (SID) and described in the following. SID exploits an IRP dissemination strategy strictly integrated with the REDMAN RD one and has demonstrated to impose lower overhead than the other investigated RR solutions in most common usage scenarios.

Let us briefly recall that REDMAN RD disseminates replicas on nodes at fixed distance along an approximately constant direction. The SID IRP diffusion consists in propagating IRPs, for any replicated resource and at the time of replica distribution, on all nodes located along the almost constant direction used during resource replication, that is along the approximately rectilinear path between disseminated resource replicas. In other words, differently from *k*-DID, SID does not aim at spreading IRPs over the whole dense MANET but only along a single direction. In addition, SID respects the first two *k*-DID constraints in stationary conditions (non-mobile nodes), but not the third one. Consequently, SID permits both to store IRPs on a limited number of nodes and, at the same time, to limit IRP message overhead during IRP dissemination and resource retrieval. In fact, a REDMAN client looking for a resource exploits search messages that also propagate along approximately constant directions: the probability to rapidly determine an intersection between the direction of retrieval and that of IRP placement is high for most usual deployment scenarios, thus enabling efficient replica searches [8]. In Figure 2, delegate A disseminates a resource replica to node E; the result of replica distribution is also that B, C, and D store IRPs with the information of resource availability at nodes A and E. When node G looks for that resource, the search message propagates until it reaches node D that owns the needed IRP.

In more details, during replica dissemination all nodes forwarding replication packets maintain a reference to their sending delegate. During retrieval it is sufficient that searchers reach one of the nodes along the replica placement line to discover where a delegate is, with no need to contact the replica manager. Since search messages are locally broadcasted to all neighbors to determine suitable successors, anyone owning the IRP of the searched resources can notify the client. In the case of search failure after a timeout, REDMAN clients start exploring a different direction for retrieval exploration.

In many profitable scenarios, devices are carried by users almost fixed during most time of service delivery, e.g., spectators seated or slowly moving on terraces during a sport event. However, in general, the movements of REDMAN devices hosting replicas and/or IRPs could affect resource availability. For this reason, we have extended SID with a decentralized and completely local maintenance protocol (SID reconstruction) in charge of loosely understanding that some IRP-hosting nodes have moved and of re-distributing IRPs to suitable neighbors. When a node, notified by the DMC facility [3], loosely detects the leaving of its predecessor/successor along the straight replication path, it locally broadcasts a reconstruction message. All nodes receiving that message from both a predecessor and a successor are eligible to replace the moved node. These nodes reply to the predecessor, and it designates one of them. Only in the case the predecessor does not receive any reply (there is no suitable node or more consecutive successors have simultaneously moved), after a relatively large time interval, it re-starts a new IRP distribution. Let us point out that SID reconstruction does not aim at maintaining the strict any-time consistency of IRPs but only at lazily re-establishing IRP alignment. In addition, the implementation of SID reconstruction is optimized to enable single-message cumulative adjustments for IRPs of different resources.
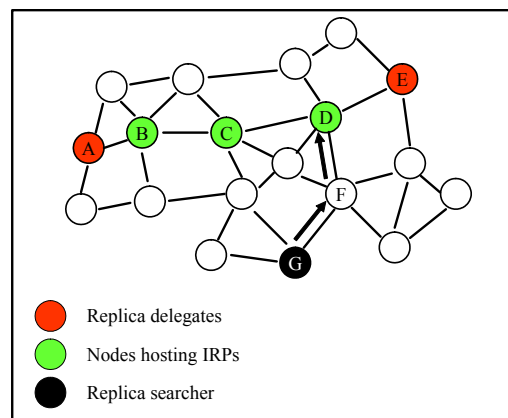


**Figure 2. The distribution of replicas/IRPs and RR exploit approximately constant directions in REDMAN.**

# 5. Experimental Results

To validate the scalability of our approach, we have extensively simulated the behavior of REDMAN protocols in challenging wide-scale provisioning environments with several hundreds of mobile nodes, on top of ns-2 [9]. We have used a square deployment area (side=1.7km) with 400 randomly positioned nodes. The dense MANET diameter, i.e., the longest minimum path between two nodes belonging to the dense region, is in most cases equal to 12. If not differently specified, we have used default ns-2 values for all simulation parameters, e.g., constant 250m circular transmission ranges, 802.11b MAC layer and bi-directional connectivity.

The simulations presented in the following have three main goals: i) to evaluate the accuracy of the SID RR strategy depending on different parameter tuning; ii) to determine SID network overhead and compare it with the other discussed RR solutions; iii) to assess the SID robustness in the case of node mobility. A wider set of experimental results, coming from both simulations and in-the-field experiments, with additional performance figures not only about RR but also related to the other REDMAN facilities, is available at `http://www.lia.deis.unibo.it/Research /REDMAN`

## 5.1. Accuracy

The first performance indicator considered for SID evaluation is accuracy, defined as the ratio between the number of successful resource searches and the total number of searches in stationary scenarios with fixed nodes. No re-iterations of the SID protocol are taken into account; searches are considered successful only if they find the needed IRPs by exploring the first retrieval direction. Mainly two tunable parameters have demonstrated to affect the SID accuracy value in stationary scenarios: the number $i$ of nodes hosting IRPs (that is the number of nodes along the straight path where replicas are positioned) and the maximum number $s$ of hops explored in the retrieval phase.

The tuning of $i$ and $s$ permits to trade the SID accuracy against its imposed message overhead. Figure 3 shows the results obtained over more than 1,000 simulations with $i$ and $s$ independently varying from 2 to 15 hops (in the case distribution/retrieval paths reach network boundaries before arriving at their maximum number of allowed hops, REDMAN automatically makes paths continuing with a new random direction back in the dense MANET). Each plotted value represents the average of 20 simulations where delegates,

dynamically determined by REDMAN RD, distribute $i$ IRPs, and randomly chosen clients look for a resource replica by exploiting an $s$-hop-limited query. The reported results show that, when $i$ and $s$ are greater than the diameter of the considered dense MANET, the accuracy overcomes 85%. In all simulations done, we have experienced that choosing $i$ and $s$ values approximately equal to the dense MANET diameter permits to achieve sufficient accuracy, with limited message overhead in both phases of SID-based IRP dissemination and RR.
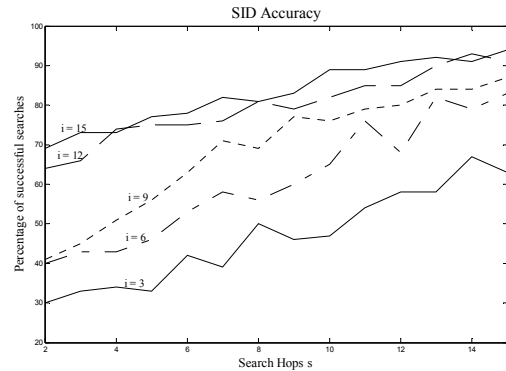


**Figure 3. SID accuracy while varying *i* and *s*.**

## 5.2. Overhead

To quantitatively compare the SID message overhead with the other presented RR solutions, we have measured the overall number of messages required to distribute and find replicas in the same challenging simulation scenario of Section 5.1. Figure 4 reports the experimental results obtained while increasing the number of searches. Each point represents the average of 20 simulations. SID parameters $i$ and $s$ are set to 12, according to what observed in the previous section. $k$-DID exploits the same number (12) of disseminated IRPs, so to realize an actual deployment scenario where it makes sense to perform a comparison between message overheads.

Figure 4 shows how QF produces a rapidly growing amount of message exchanges also for a limited number of searches. As expected, $k$-DID imposes a high overhead for IRP placement (about 4 messages per node in the dense MANET), while its search phase demonstrates to be very effective. SID exhibits linear growth in overall message overhead, by imposing a lower number of IRP distribution messages than QF and $k$-DID and only a slightly greater number of RR messages than $k$-DID. Both $k$-DID and SID require low memory occupation on IRP-hosting nodes, since they diffuse IRPs only on a limited node subset (see the concise comparison among RR solutions in Table

1). IF represents a lower bound for communication overhead, but it is a practically unviable solution because it requires all nodes in the dense MANET to store IRPs about all replicas of all available resources.
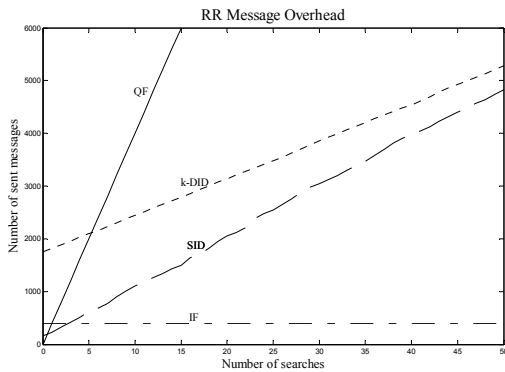


**Figure 4. Message overhead for the four presented RR solutions.**

| RR Strategy | IF | QF | *k*-DID | SID |
|---|---|---|---|---|
| Global IRP memory occupation | N * size(IRP) | size(IRP) | O(N/F(k/2)) * size(IRP) | O(i) * size(IRP) |
| Dissemination message overhead | N | 0 | O(N) | depends on i, density |
| Retrieval message overhead | 0 | N | O(F(k/2)) | depends on s, density |
| Scalability | Suits deployment scenarios with few searches | Unviable for large scenarios | Fits stationary scenarios with a very high number of searches | Fits also mobile scenarios with limited number of searches |

**Table 1. Concise comparison of IF, QF, k-DID, and SID RR strategies.**

## 5.3. Accuracy in Non-Stationary Scenarios with Mobile Nodes

To evaluate the dynamic behavior of SID in non-stationary deployment scenarios with mobile MANET nodes, we have adopted a mobility model establishing that, after a randomly chosen time interval, any node in the dense region starts moving along a rectilinear path, with randomly chosen speed direction and speed module randomly chosen in the range [1, 5] m/s.

Figure 5 presents the temporal evolution of SID accuracy by comparing it with a SID version without the local IRP maintenance reconstruction protocol.

Plotted results are average values over 20 simulated scenarios where only 10% of IRP owners remain fixed. As for the overhead evaluation in Section 5.2, *i* and *s* are set to 12. The figure shows that, after a sufficient number of nodes has moved and left the dense MANET, the reconstruction-enabled SID version outperforms the other. Reconstruction-enabled SID has demonstrated to maintain good accuracy notwithstand-

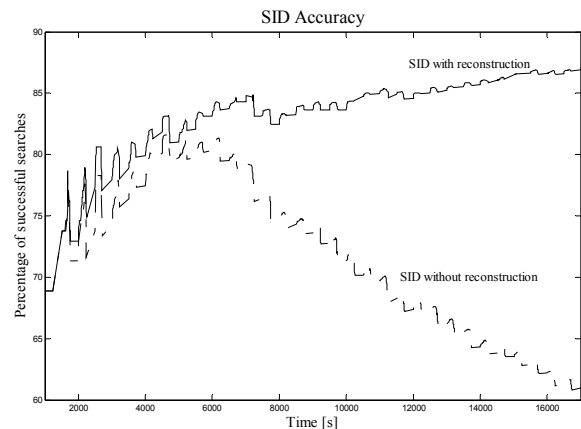ing the challenging mobility model adopted with limited message overhead.



**Figure 5. SID accuracy with/without reconstruction in mobile scenarios.**

## 6. Related Work

The idea of increasing the availability of MANET applications by replicating data/service components is still at its very beginning. So far, the research has mainly focused on replication to ensure data availability in the case of MANET partitioning. Most proposals assume that wireless nodes are aware of their physical position, e.g., by imposing all participants to be GPS-equipped [10]. Other activities mainly aim at respecting strict requirements about replica synchronization/consistency and, therefore, impose a non-negligible network overhead [11, 12]. In particular, [12] presents a secure middleware that aims at enhancing data availability by exploiting an adaptive replication protocol that also permits replica updates; however, that proposal only addresses scenarios with nodes in direct single-hop visibility and assumes that any node has complete knowledge of position of all maintained replicas and of memory/battery/mobility state of other peers. [13], instead, proposes a novel solution for read-only replicas based on the guideline to counteract network partitioning via proactive replication depending on the frequencies of resource access; the approach imposes coordination and synchronization among participants and does not scale well with growing numbers of nodes and replicated resources.

In addition, infrastructure-free and completely decentralized MANETs pose novel challenges for resource retrieval. Some activities have investigated MANET-specific broadcast-based peer-to-peer RR solutions, with limited scalability and excessive over-

head for resource-constrained clients [14]. Recent research proposals aim at limiting broadcast communications by exploiting quorum-based solutions [8, 15, 6]: the primary idea is to disseminate resource placement information on a node subset determinable without message flooding. [8] and [15] specifically address ad-hoc sensor networks where nodes are fixed. On the one hand, [15] assumes GPS-equipped nodes to spread advertisement/search packets along orthogonal directions; on the other hand, similarly to REDMAN, [8] considers provisioning environments where geographic routing cannot apply and proposes to diffuse placement data along paths with approximately constant directions, determined by choosing, as the next hop, a node that is not the neighbor of any node belonging to the already built sub-path. [6] extends the GPS-based solution in [15] for replication in mobile environments: each node only stores placement data about its nearest replica. A different approach is presented in [7]: any resource is associated with its origin place, which is in charge of distributing all replicas according to an original protocol similar to $k$-DID, as stated in Section 4.1; in that proposal, replica retrieval exploits geographical routing.

## 7. Conclusion

The challenge of supporting dense MANET applications can significantly exploit the assumption of high node population to enable lazy consistent forms of resource replication. This can increase the availability of data/service components of common interest notwithstanding unpredictable node movements at provision time. REDMAN demonstrates how it is possible to provide middleware solutions for lightweight and effective replica management under the dense MANET hypothesis. In particular, the paper presents, discusses, and compares original and completely decentralized protocols, specifically designed for dense MANETs, to effectively retrieve disseminated resource replicas, without exploiting any point of centralization and any GPS equipment. The experimental performance results obtained for the different retrieval solutions has suggested us to choose and integrate the SID RR strategy in REDMAN: SID has shown to be effective in resource retrieval and to impose a limited communication overhead for both IRP dissemination and RR, by exhibiting also good scalability in wide-scale scenarios.

The promising results obtained are encouraging further REDMAN-related research activities. First, we are working on testing and measuring the performance of the REDMAN prototype in-the-field, by deploying the middleware in actual dense MANETs consisting of about one hundred devices with IEEE 802.11b connectivity in ad-hoc mode. Secondly, we are investigating highly decentralized security mechanisms to enable the dense MANET participation (and the access to shared resources) only to authorized nodes. Finally, we are developing application prototypes for civil protection scenarios where our replication middleware is used to share gathered data and to maintain them available, where and when needed, despite of operator turnovers.

## Acknowledgements

## References

[1] I. Chlamtac, M. Conti, J. J.-N. Liu, "Mobile Ad hoc networking: imperatives and challenges", *Elsevier Ad Hoc Networks*, Jul. 2003.

[2] P. Bellavista, A. Corradi, E. Magistretti, "REDMAN: a Decentralized Middleware Solution for Cooperative Replication in Dense MANET", *3rd IEEE PERCOM Workshop*, Mar. 2005.

[3] P. Bellavista, A. Corradi, E. Magistretti, "Lightweight Replication Middleware for Data and Service Components in Dense MANETs", *1st IEEE WoWMoM*, Jun. 2005.

[4] S. Decker, P. Mitra, S. Melnik, "Framework for the semantic Web: an RDF tutorial", *IEEE Internet Computing*, Nov.-Dec. 2000.

[5] wwws.sun.com/software/jini

[6] J. Tchakarov, N. Vaidya, "Efficient Content Location in Mobile Ad hoc Networks", *IEEE Int. Conf. Mobile Data Management*, Jan. 2004.

[7] M. Tamori, S. Ishihara, T. Watanabe, T. Mizuno, "A Replica distribution method with consideration of the positions of mobile hosts on wireless ad-hoc networks", *22nd IEEE ICDCS Workshop*, Jul. 2002.

[8] D. Braginsky, D. Estrin, "Rumor Routing Algorithm for Sensor Networks", *1st ACM Int. Workshop Wireless Sensor Networks and Applications,* Sep. 2002.

[9] www.isi.edu/nsnam/ns

[10] K. Chen, K. Nahrstedt, "An Integrated Data Lookup and Replication Scheme in Mobile Ad Hoc Networks", *SPIE Int. Symp. on the Convergence of Information Technologies and Communications (ITCom 2001)*, Aug. 2001.

[11] K. Rothermel, C. Becker, J. Hahner, "Consistent Update Diffusion in Mobile Ad Hoc Networks", *Technical Report 2002/2004*, CS Dep., Univ. of Stuttgart.

[12] M. Boulkenafed, V. Issarny, "A Middleware Service for Mobile Ad Hoc Data Sharing, Enhancing Data Availability", *ACM/IFIP/USENIX International Middleware Conference*, Jun. 2003.

[13] T. Hara, "Effective Replica Allocation in Ad Hoc Networks for Improving Data Accessibility", *20$^{th}$ IEEE INFOCOM*, Apr. 2001.

[14] A. Helal, N. Desai, V. Verma, L. Choonhwa, "Konark - a Service Discovery and Delivery Protocol for Ad-Hoc Networks", *3$^{rd}$ IEEE Conf. Wireless Communications Networks,* Mar. 2003.

[15] I. Aydin, C.-C. Shen, "Facilitating Match-Making Service in Ad Hoc and Sensor Networks using Pseudo Quorum", *11$^{th}$ IEEE Int. Conf. Computer Communications and Networks*, Oct. 2002.