REDMAN: a Decentralized Middleware Solution for Cooperative Replication 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

The mass market of wireless devices is pushing towards service provisioning over dense Mobile Ad-hoc NETworks (MANETs), i.e., limited spatial regions, such as university campuses, airports and shopping malls, where many mobile wireless peers autonomously cooperate, without the need of statically deployed support infrastructures. Dense MANETs can take advantage of high node population to replicate resources of common interest to increase their availability overcoming unpredictable node movements. The paper proposes a lightweight application-level middleware, called REDMAN, to manage, retrieve and disseminate replicas of data and service components transparently from the point of view of service developers, thus facilitating the realization of scalable distributed applications for dense MANETs. REDMAN proposes novel lightweight solutions, specific for and effective in dense MANETs, to determine dense region boundaries, to perform resource cloning/distribution/retrieval, and to approximately maintain the desired resource replication degree.

1. Introduction

The increasing diffusion of wireless devices suggests novel service deployment scenarios where there are no constraints on device mobility and distributed applications are the result of impromptu collaborations among wireless peers. In these scenarios, not only wireless nodes should have location-aware resource access without requiring static knowledge about their execution environment, but also resources should be permanently available, not depending on node movement, disconnection, and battery shortage [1].

Some research activities are investigating MANET-specific solutions to bind/rebind to newly discovered distributed resources, thus enabling wireless clients to automatically redirect requests to service components also by considering their mutual location [2, 3]. On the contrary, the idea of increasing the availability of MANET applications by replicating data and service components close to their current clients is still at its very beginning. Only few state-of-the-art proposals have started to face the very challenging issue of resource replication in mobile environments, to increase probability and effectiveness of resource access [4]. Most investigations have recently addressed the issue of information availability in cellular and infrastructure-mode IEEE 802.11 networks. However, in that context, it is possible to exploit the fixed part of the network infrastructure, e.g., to store and replicate personal data at highly available servers on wired stable links. The most critical issue in infrastructure-based deployment scenarios is to properly manage user disconnections during update operations on shared data, possibly by automatically handling the reconciliation of multiple modified copies [5, 6].

Due to the complete lack of a static support infrastructure, the effective replication of data and service components in MANET dynamic environments is a hard challenge, which requires re-thinking and significantly modifying traditional approaches to resource replication. So far, the research has almost exclusively focused on replication to ensure data availability in case of network partitions. Most proposals assume that wireless nodes are aware of their physical position, e.g., by imposing the constraint of hosting Global Positioning System hardware at any participant [7]. Other research activities aim at answering strict requirements about replica synchronization and consistency, and therefore impose a heavy overhead, in terms of both network traffic and requested time to ensure the consistency of all replicas [8].

We claim that the management of data/service component replicas is a very hard task to perform in an effective and lightweight way when dealing with general-purpose MANETs and with strict consistency requirements. Therefore, we focus on a specific deployment scenario of increasing relevance for the service provisioning market, called dense MANET in the following. We use the term *dense MANET* to indicate a MANET 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, airport waiting rooms, and university campuses;
- has a node density (the average number of wireless nodes at single-hop distance from any dense MANET participant) that is almost invariant during long time intervals.

In particular, the paper proposes a middleware solution, called REDMAN (**RE**plication in **D**ense **MAN**ETs), that transparently disseminates, manages, and retrieves replicas of common interest resources among cooperating nodes in the dense MANET. REDMAN has the main goal of improving the availability of data and service components, by exploiting lightweight solutions specifically suitable for the characteristics of dense MANETs. The primary idea is of maintaining, within the dense MANET, a fixed replication degree for the needed resources, independently of possible (and unpredictable) exits of replica-hosting nodes from the dense region.

REDMAN operates at the application level because several replica management decisions, such as the suitable replication degree depending on differentiated resource criticality, are typically at this abstraction layer [9]. Working at the application level also simplifies portability over heterogeneous connectivity technologies and routing protocols. In addition, REDMAN transparently performs replica management from the point of view of service developers/administrators, who only have to indicate the criticality of the shared resources involved in service provisioning. Moreover, REDMAN has been specifically designed for battery/memory-constrained devices (PDAs, smart phones, …), which typically cannot host positioning hardware and cannot store all needed data and service components in their local memory permanently; REDMAN simplifies the mutual interactions of limited portable devices to enable collaborative service provisioning in dense MANETs.

The rest of the paper is organized as follows. Section 2 provides an overview of the REDMAN middleware, while Section 3 overviews the REDMAN original solutions for dense MANET identification and decentralized replica manager election. Section 4 focuses on high-layer REDMAN facilities for replica distribution, retrieval, and degree maintenance. Conclusions and directions of on-going research 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, e.g., due to node mobility and battery shortage. More formally, a dense MANET is defined as the set of MANET nodes $DM(n) = \{d_0, \ldots, d_{N-1}\}$, 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 *k*, REDMAN is in charge of instantiating and distributing *k* replicas of it, and of maintaining the *k* replication degree notwithstanding the changes in the composition of the DM(n) set.

In particular, to suit resource-limited nodes, REDMAN proposes dense MANET-specific lightweight solutions. REDMAN decides not to guarantee the strict any-time consistency of the replication degree of shared resources. Instead, it employs reactive strategies to counteract the reduction of replicas in DM(n) when resource delegates either fail or leave the network. In addition, to reduce the overhead and the complexity of distributed replica management, REDMAN currently manages the replication of read-only resources, thus permitting to exclude heavy and expensive operations for possible reconciliation of concurrently updated resource replicas. Dealing with read-only resources is sufficient for guaranteeing the availability of a large class of services of primary interest in MANETs (e.g., multimedia data sharing or support /application components code distribution). General-purpose protocols for replica reconciliation in traditional wired systems are not suitable, even in adapted forms, for dense MANETs, due to their relevant overhead and connectivity requirements [10].

In addition, we claim the suitability of providing REDMAN facilities at the application level to improve flexibility, configurability, and portability over different MANET communication solutions, and to hide lower layer implementation details from application developers. Developers of services for dense MANET should only provide each service component with metadata to describe the shared resource and to suggest the suitable replication degree depending on application-specific resource criticality. Clients in the dense MANET transparently perform discovery and retrieval operations, i.e., resource replication does not affect at all their application logic.

Figure 1 depicts the two-layered REDMAN middleware architecture: the lower layer **Dense MANET**

Configuration (DMC) composed by Dense MANET Identification/Maintenance and Manager Election, and the higher layer including **Replica Dissemination (RD)**, **Replication Degree Maintenance (RDM)**, and **Resource Retrieval (RR)**.

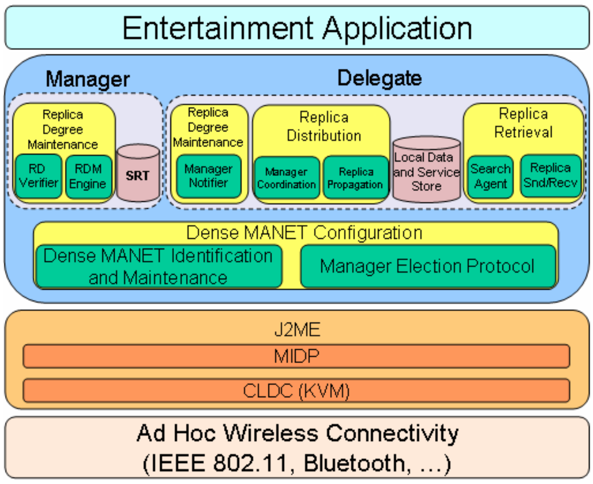

Figure 1. The REDMAN middleware architecture of facilities

3. Dense MANET Configuration

DMC has the duty of identifying the nodes in the dense MANET and electing suitable replica managers. To achieve these goals, REDMAN proposes two original lightweight protocols, specifically designed for dense MANETs, that impose very limited overhead and reach non-optimal but sufficiently accurate solutions for the addressed service scenario.

3.1. Dense MANET Participant Identification

A node belongs to a dense MANET *DM(n)* if the number of its neighbors exceeds *n* and the spatial node density in its vicinity is almost constant during long time intervals (the second condition always applies in dense MANETs given their definition).

REDMAN exploits a completely decentralized protocol to enable nodes to autonomously and locally determine the density condition fulfillment. At any time, any REDMAN node (*initiator*) can start the dense MANET identification procedure. The initiator locally broadcasts a single-hop discovery message containing the neighbor number *n*, which is the threshold to belong or not to the dense MANET. Each node receiving the message re-broadcasts it only once; since nodes receive discovery messages from all their neighbors, they can autonomously verify the belonging condition simply by counting the received messages. Due to node mobility, the accuracy of the determined set of participants tends to degrade over time; for that reason, REDMAN integrates the identification protocol with a lightweight maintenance solution that forces dense MANET participants to maintain a lazily updated list of their neighbors and to exchange low-frequency Hello packets with them.

3.2. Replica Manager Election

Dynamically chosen decentralized replica managers are responsible for deciding replication degrees of associated shared resources and for coordinating replication degree maintenance operations. This imposes a non-negligible communication overhead on manager nodes that have to periodically contact their delegates. REDMAN aims at limiting that overhead by choosing manager nodes placed in the dense MANET topological center, i.e., minimizing the hop distance from farthest dense MANET participants. To this purpose, REDMAN implements a lightweight solution for manager election that aims not at reaching optimal manager placement in any situation, but at efficiently and rapidly identifying good candidates by imposing very limited overhead.

The REDMAN election protocol explores, as replica manager candidates, only a small set of participants, called Investigated Nodes (INs). The initiator is the first IN; successive INs are chosen so to get closer to the center of the network. At each exploration step, the current IN chooses its successor, within the set of its neighbors, so to advance in the direction of farthest dense MANET nodes. In fact, by moving in that direction, the next IN is one-hop closer to the identified farthest nodes; therefore, the IN distance from farthest nodes tends to decrease and to converge toward the best solution. REDMAN exploits a heuristic-based approach to decide when to stop protocol re-iteration, i.e., when the reached solution is to be considered accurate enough. In particular, REDMAN exploits the observation that if the election protocol does not achieve any improvement in the solution quality (no decrease of farthest node distance from the current IN) while exploring a small set of consecutive INs, it is highly probable that the current solution is close enough to the optimal one.

The exhaustive description of the REDMAN DMC protocols and a large set of related performance results, coming from both simulations and in-the-field experiments, are available at the REDMAN Web site `http://lia.deis. unibo.it/Research/REDMAN/`

4. REDMAN High-Level Facilities

4.1. Replica Distribution

The **RD** facility operates to transparently distribute resource replicas in the dense MANET. REDMAN associates each resource of common interest with a metadata-based description, which includes the expected replication degree depending on the estimated resource criticality decided by service developers. When a delegate enters a dense MANET, it communicates the metadata of the shared resources to the replica manager, which decides the replication degree to enforce by considering the hint about resource criticality and the estimated number of current participants in the dense MANET. The manager maintains a Shared Resource Table (SRT) with one entry for each managed resource: each entry contains the replication degree to be enforced and weakly consistent information about replica placement (see Section 4.3). After having updated SRT, the manager delegates the resource owner for the actual implementation of replica process; the resource owner becomes the first delegate for that resource. REDMAN distributes resource replicas in a highly decentralized simple way: delegates forward their resources to randomly chosen neighbors, belonging to the dense region, by including the required replication degree. If a neighbor is willing to host a copy of the resource, it locally stores the replica, decreases the indicated replication degree, and forwards the request to another neighbor in the dense MANET; otherwise, it forwards the replica command message unchanged. In addition, when a node decides to host a replica, it notifies the replica manager; in this way, the manager can update the information in its SRT.

Since replica manager nodes have the duty of coordinating replica operations, they tend to consume more battery/network resources than other nodes. For this reason, the proposed replica distribution strategy aims at reducing manager communication load, by giving managers the only charge of sending short description messages, while delegates have the burden of actually enforcing resource replication. In addition, REDMAN periodically reallocates replica manager roles by forcing new manager elections. This also contributes to improve the suitability of nodes acting as replica managers because, during long time intervals, an elected node could lose the properties that motivated its election, e.g., its central position in the dense MANET topology.

4.2. Replica Retrieval

The **RR** facility has the goal of retrieving resource replicas effectively, by exploiting a lightweight distributed protocol. Two different reasons suggest avoiding a centralized solution: i) a large number of requests could overwhelm a single repository, and ii) that centralized repository should be updated with strict consistency requirements not to hinder the shared resource accessibility. RR exploits a simple retrieval solution based on the client flooding of resource requests.

RR avoids flooding the entire network by limiting the hop-distance at which resource retrieval requests are spread. When a node receives a search message, it checks whether it owns a copy of the needed resource. In that case, it replies to the requester by stating the resource availability; otherwise, it forwards the request to all its neighbors. Each request is tagged with a unique identifier not to repeatedly forward the same search message; in addition, a node does not forward request messages corresponding to an already forwarded reply. The above retrieval solution is quite trivial, but the proposal of novel and effective RR algorithms is not the focus of the REDMAN research activity. Current work on the RR facility is addressing the integration with replica allocation strategies to minimize the time (and the generated network traffic) for resource retrieval.

4.3. Replica Degree Maintenance

Proactive replication strategies, which generally require positioning systems on client devices and significant processing resources for predicting node disconnections, are unsuitable for dense MANETs [7]. REDMAN proposes an original lightweight **RDM** facility that works to maintain unchanged the replication degree decided for each shared resource, without guaranteeing absolute consistency, i.e., it is possible to have time intervals when the requested replication degree differs from the actual number of replicas in the dense MANET. In other words, RDM only aims at maintaining consistency likelihood for the chosen resource replication degree, with the relevant pro of significantly limiting the overhead on participant nodes.

After the initial replica distribution performed by RD, RDM simply works reacting when resource delegates leave the dense MANET. A delegate can realize it is going to exit from the dense region: in this case, it autonomously replicates its shared resources on adjacent nodes belonging to the dense MANET; these nodes then notify the replica manager of the occurred change. If a delegate is not able to foresee its leaving in time, once it realizes not to be in the dense region anymore, it tries to advise the manager (probably still reachable via intermediate nodes near the dense MANET boundaries) by specifying its hosted shared resources; it is the replica manager that commands to other delegates for those resources, as indicated in

SRT, to distribute new replicas so to re-establish the required replication degrees. The above strategy does not guarantee strict consistency at any time between the actually maintained and the requested replication degree, e.g., because of message losses or abrupt failures/exits of delegates from the dense region. For this reason, at large time intervals, resource delegates periodically send to the manager the list of their locally hosted shared resources; in the case that a delegate stops sending update messages, the manager commands new replicas of its resources to other delegates and correspondingly updates SRT.

Additional details about the REDMAN high-level facilities, together with their implementation and related experimental results are available at the REDMAN Web site, where you can also find the updated state of the on-going implementation of REDMAN-based applications in the areas of entertainment computing and emergency rescue.

5. Conclusions and On-going Research

The challenge of supporting distributed services over dense MANETs can significantly exploit the assumption of high node population to enable lazy consistent forms of resource replication. This can increase the availability of resources of common interest notwithstanding the unpredictable node entrance/exit in/from the dense MANET. The REDMAN project demonstrates how it is possible to provide application-level middleware solutions for lightweight and effective replica management also in dense MANET. In particular, REDMAN proposes original and completely decentralized protocols, specifically designed for dense MANETs, to dynamically determine dense region participants and to elect replica managers by minimizing the distance from farthest nodes. To the best of our knowledge, there are no other research activities that have already proposed and evaluated solutions for dense MANET identification and non-GPS-based determination of the MANET topology center.

The experimental results obtained are encouraging further REDMAN-related research activities. First, we are working on testing the REDMAN prototype in a dense MANET obtained by deploying a large number of devices with IEEE 802.11b connectivity in ad-hoc mode. Secondly, we are investigating and experimenting highly decentralized security mechanisms to enable the dense MANET participation (and the access to shared resources) only to authorized nodes.

Acknowledgements

Work supported by the Italian MIUR FIRB WEB-MINDS and by the Italian CNR Strategic IS-MANET Projects.

References

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

[2] Y. Yuan; W. Arbaugh, “A Secure Service Discovery Protocol for MANET”, *14th IEEE Conf. on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Sep. 2003.

[3] A. Helmy, S. Garg, P. Pamu, N. Nahata, “Contact-based Architecture for Resource Discovery (CARD) in Large Scale MANETs”, *17th IEEE Int. Parallel and Distributed Processing Symp. (IPDPS)*, Apr. 2003.

[4] T. Qing, D.C. Cox, “Optimal Replication Algorithms for Hierarchical Mobility Management in PCS Networks”, *IEEE Wireless Communications and Networking Conf. (WCNC)*, Mar. 2002.

[5] J. J. Kistler, M. Satyanarayanan, “Disconnected Operations in the Coda File System”, *ACM Transactions on Computer Systems*, Feb. 1992.

[6] A. J. Demers, K. Petersen, M. J. Spreitzer, D. B. Terry, M. M. Theimer, B. B. Welch, “The Bayou Architecture: Support for Data Sharing among Mobile Users”, *1st IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, Dec. 1994.

[7] 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.

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

[9] S. Garg, Y. Huang, C.M.R. Kintala, K.S. Trivedi, S. Yajnik, “Performance and Reliability Evaluation of Passive Replication Schemes in Application-level Fault Tolerance”, *29th IEEE Int. Symp. on Fault-Tolerant Computing*, June 1999.

[10] L.P. Cox, B.D. Noble, “Fast Reconciliations in Fluid Replication”, *21st Int. Conf. on Distributed Computing Systems (ICDCS)*, Apr. 2001.