

Quorum Sensing on Mobile Ad-Hoc Networks

Maxim Peysakhov, Christopher Dugan, Pragnesh Jay Modi, William Regli

Department of Computer Science
Drexel University
Philadelphia, PA 19104

ABSTRACT

Mobile ad-hoc networks (MANETs) are an increasingly important networking paradigm that will be the backbone of important defense and first response networks. Group decision-making is key to these environments, but is made difficult when MANETs are introduced due to network disruptions, bandwidth limitations, and host mobility patterns. Results gathered using standard group decision-making algorithms can become inaccurate, time-insensitive, or computationally undecidable. This paper focuses on a group decision-making approach using agent-based quorum sensing (ABQS) on MANETs. A mobile agent collects information (e.g., votes) from each host on a network until it can make an informed decision about global preference. This agent exploits the inherent tradeoff between efficient vote collection and result accuracy in order to provide better results, when considering survivability, hosts visited, hops made, and time spent, with only a very slight drop in correctness—benefits that greatly outweigh costs. Experimental evidence from live MANETs demonstrates the effectiveness of this solution.

Categories and Subject Descriptors

I.2.11 [Artificial Intelligence]: Distributed—Intelligent agents

General Terms

Reliability, performance

Keywords

Mobile agents, robustness and dependability of agent systems

1. INTRODUCTION

Mobile ad-hoc networks (MANETs) are composed of mobile computers that communicate over a wireless network without the support of an existing, fixed infrastructure. Instead, hosts can join or leave the network as they come in and out of range and each host acts as a router, allowing message traffic meant for another host to pass through it on a multi-hop route to its destination. MANETs present significant difficulties due to their high degree of decentralization, unstable physical-layer connectivity and resource con-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

AAMAS'06 May 8–12 2006, Hakodate, Hokkaido, Japan.
Copyright 2006 ACM 1-59593-303-4/06/0005 ...\$5.00.

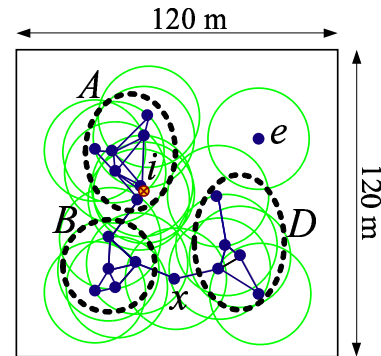


Figure 1: A simulated MANET topology.

straint limitations. As a result, group decision making is a challenge that must be addressed. Security decisions protect the network from intruders, routing decisions affect the ability to communicate, and bandwidth decisions ensure network efficiency [2].

An example group decision involving security on a MANET is highlighted in Figure 1 which depicts the *key revocation decision problem* [1]. Host x is believed to be compromised and no longer trustworthy so a decision must be made whether or not to revoke Host x 's key. This process is costly since it involves the creation and distribution of a new key plus it could possibly eliminate network connections between nodes (see cluster D); thus, the group should be certain that the host has really been compromised and that it can continue to operate functionally without Host x in the network. The obvious solution of voting is non-trivial due to the difficulties, described above, that MANETs present. Thus, elections can be incredibly inefficient or, at worst, undecidable.

In this paper, we propose an *agent-based quorum sensing* (ABQS) approach for vote collection in a MANET. In quorum sensing, vote collection is executed by a mobile agent or agents that traverses the hosts in the network. This approach promises to be more effective than conventional source/destination or broadcast message passing because the mobile agent is able to reason and adaptively choose its next destination based on certain percepts: connectivity, security, and distance to goal are examples. The agent uses a quorum sensing technique that employs a probability ratio calculation to compute a threshold. After the threshold is exceeded, the mobile agent decides that a reliable enough outcome has been determined. Furthermore, by exploiting the assumption that vote distributions are random, we can precisely calculate confidence intervals for any given threshold. This is important because it can allow principled tradeoffs between efficiency or cost of the vote collection procedure

and accuracy of the outcome decision.

2. AGENT-BASED QUORUM SENSING

There are two key components of an agent for quorum sensing:

1. An **agent migration procedure**, defining how the mobile agent navigates its environment to collect input (e.g., votes) from different hosts. In such a case, the environment is a MANET, hence agents need to sense and react to stimuli from the lower layers of the network stack (specifically the PHY, MAC and Link layers in the OSI Reference Model).
2. A **quorum evaluation procedure**, defining a statistical test to determine if the decision procedure can be terminated with high confidence of correct prediction of the outcome.

Agent Migration Procedure. Agent migration poses a unique problem on a MANETs. Most mobile agent approaches assume that communications are perfect or, at a minimum, reliable. In a MANET, migration of an agent from host a to host b may require traversal of multiple network hops that consume network resources (i.e., power, bandwidth, etc)—if it is even possible in the first place. Reliability is decreased as a consequence. Thus, this approach aims at reducing hop-count in order to increase reliability while trying to maintain a certain level of accuracy. Consider a network consisting of a set of hosts $H = \{h_1, h_2, \dots, h_{|H|}\}$. Without loss of generality, assume the quorum sensing agent starts off with a workflow that traverses these hosts in order (i.e., $h_1, h_2, \dots, h_{|H|}$). A greedy heuristic can be defined with a simple inductive argument. For the base case, assume agent a is at host h_1 :

- Let $\bar{H}$ be the hosts traversed so far, hence $\bar{H} = \emptyset$ and agent a needs to visit the hosts in the set H/h_1 ; $\text{succ}(h_1) = H/h_1$.
- Let $\text{hop-count}(h_1, h_i)$ be the number of network hops from host h_1 to host h_i . This measure can be sensed using sensors that can access the state information maintained by the network routing protocol in the MANET (i.e., OLSR or AODV).
- Select the next host h_i such that $\forall h_j \in H/\{h_1, h_i\}$, $\text{hop-count}(h_1, h_i) \leq \text{hop-count}(h_1, h_j)$; $\bar{H} = \bar{H} \cup \{h_i\}$.
- Migrate to h_i .

For the inductive case, assume agent a is at a host h_k :

- If the *quorum threshold* has been reached, terminate migration and return to the originating host with the result; else
 - $\text{succ}(h_k) = H/\bar{H}$;
 - Select the host h_i such that $\forall h_j \in H/\{h_k, h_i\}$, $\text{hop-count}(h_k, h_i) \leq \text{hop-count}(h_k, h_j)$; $\bar{H} = \bar{H} \cup \{h_i\}$.
 - Migrate to h_i .

Quorum Evaluation Procedure. In ABQS, we deploy a mobile agent to sequentially collect votes—not unlike Wald’s classic sequential probability ratio test [7]. The ABQS vote collection agent contains a quorum sensor based on those found in social insects, specifically the *Leptothorax albipennis* family of ants. These ants estimate the quality of new nesting sites by observing a *quorum number*, a rough calculation of the number and suitability of nest mates [5]. Whenever this *quorum number* goes outside of an ants indecision threshold, the nesting site can be decided upon.

Our goal is to determine a threshold which results in a statistically reliable performance for the quorum sensing agent in a vote estimation task. We assume that the samples are independent. If this is not the case, we force the agent to collect votes in a random manner thereby removing the dependency. The probability density function (PDF) f is then computed for the fair binomial distribution for the corresponding number of tests as each vote is collected:

$$f(x = k) = \binom{n}{k} p^k (1 - p)^{n-k}$$

Let n be the number of votes collected, k be the number of positive votes and $p = 0.5$. The value of the function f is the probability that a sample with exactly k positive votes was generated by a binomial distribution with mean of 0.5. The narrowest confidence interval C determined around the mean of 0.5 such that the combined probabilities of all its members $F(C)$ are at least 90%:

$$F(C) = \sum_{k \in C} f(x = k) \geq 0.9$$

Note that the 0.9 can be changed in order to take advantage of the tradeoff between accuracy and time mentioned earlier. Elections are too close to call and the process of collecting samples continues when the collected sample belongs to the interval C . If, however, the collected sample is outside of the interval C , then this would indicate a strong bias toward one of the two options and the election can be decided with certainty $\frac{1+F(C)}{2}$. More confidence in the predicted result can be achieved by broadening the interval C .

3. EMPIRICAL VALIDATION

We analyze the performance of ABQS briefly in simulation and focus on embodied experiments. Votes can be collected in two ways: sequential and parallel. Sequential vote collection sends out a single agent which tours the network and collects votes. ABQS has an obvious implementation in such a case because the agent determines when the election is decidable. Parallel vote collection sends one agent to each host and returns with the result. The Host collecting the votes implements ABQS as votes are received and stops collecting ballots once the threshold is surpassed.

Simulated Experiments. A MANET simulator [6] is employed to test these hypotheses for large numbers of hosts under a variety of networking scenarios. The simulator enables the creation and mobility of agents which can perform various tasks—voting, in this case. The transmission time for each agent over each link is inversely proportional to the square of the physical distance between the hosts if they are in radio range of each other. Links are random and continuously changing. Agents cannot travel between hosts that have not established direct radio contact.

The network simulator was initialized with 30 mobile hosts. Each host was given a random vote with [60 : 40] bias towards the “yes” decision. Agents were given 500 iterations of the simulator to collect the votes and report their decision. The experiment was repeated 500 times during a single run of the simulator. We also performed 10 independent runs of the simulator to reduce the dependency on the random number generator used in the experiments. As a result, 5000 data points were collected for each agent. Out of these experiments, agent $\mathcal{A}$, a sequential collecting ABQS agent, was able to produce a result 83% of the times while agent $\mathcal{T}$, a sequential collecting agent which requires all votes, produced the result only in 77% of all cases. Considering only the experiments where election results were obtained or estimated, we computed measures for correctness and performance improvement. Table 1 gives the comparison of both agents based on their average performances. Agent $\mathcal{A}$ is 6% more survivable and about 19% more efficient (in terms of visited hosts, network hops and time spent) than traditional agent $\mathcal{T}$ with only 0.5% sacrifices in correctness.

Embodied Experiments. We present experiments using Drexel University’s live MANET testbed [4, 3]. It consists of dozens of mobile computing hosts—PDAs and tablets—on an 802.11b wireless network with ad-hoc routing. In particular, five (5) Compaq TC-1000 tablet PCs were configured with the OLSR MANET routing protocol operating over CISCO Aironet 802.11b network cards. These hosts were placed in an actual outdoor urban environment.

Agent	No ABQS	ABQS
Correctness (%)	100.0	99.5
Survivability (%)	77.0	83.0
Hosts Visited (%)	100.0	81.1
Hops Made (hops)	131.1	107.5
Time Spent (sec)	407.6	335.3

Table 1: Performance comparison for simulated experiments.

Method	Sequential		Parallel	
	No ABQS	ABQS	No ABQS	ABQS
Correctness (%)	100.0	100.0	100.0	100.0
Survivability (%)	90.0	100.0	93.0	100.0
Hops Made (hops)	29.4	12.6	61.2	22.5
Hops Made (<i>st.dev.</i>)	3.7	4.4	5.7	11.7
Time Spent (sec)	14.4	7.6	5.9	3.5
Time Spent (<i>st.dev.</i>)	10.3	7.5	5.7	1.6

Table 2: Performance comparison for embodied experiments.

Note that this is an uncontrolled environment that includes multi-path interference, other networks as well as disruptions from foliage, automobiles and physical structures.

Hosts 1 through 3 are static, positioned linearly and approximately 40 meters apart. Hosts 4 and 5 are mobile and randomly change position. Each physical device runs five virtual hosts. A slight modification to the routing protocol makes hosts on the same physical device appear two hops away. This forces agents to treat virtual hosts just like they would physical hosts—simulating a larger network with fewer physical nodes. Host 4 begins all election procedures and we compare sequential and parallel vote collection with and without ABQS.

The votes of individual group members were pre-determined and randomly distributed prior to the vote collection in such a way that there was an exact 74% bias in each election towards a “yes” vote. Rogue hosts are uncommon in practice so a 74% bias conservatively approximates real-life situations. The threshold for ABQS was set to 95% accuracy in all of the performed experiments. All agents are compared on completion time, bandwidth utilization, whether or not it completed, and accuracy. The experiments were repeated 30 times. Average values and standard deviations of measured parameters are presented in Table 2.

In each of the experiments, all of the elections were tabulated correctly. The size of our network (25 hosts) and the number of experiments did not appear to be large enough to allow any mistakes in ABQS to manifest themselves. Both parallel and sequential vote collectors utilizing ABQS were always able to call the elections before the deadline. Vote collection without ABQS had a failure rate of 10% and 7% for sequential and parallel vote collection.

As expected, a sequential process of vote collection results in less network traffic and longer times than a parallel process. However, applying ABQS allows 57% less traffic for sequential vote collection. We recognize that the number of hops traversed prior to the making a decision for parallel ABQS does not represent accurately the savings in the bandwidth usage, since reaching a decision does not automatically remove all the vote collection agents that are still in transit. Although time costs have much higher standard deviations, we still saw a speed up of 47% for sequential and by 40% for parallel—levels that are statistically significant.

The numbers from comparison Table 2 and the histograms in Figure 2 show that both sequential and parallel methods of vote collection can gain in reliability as well as significantly increase their performance by utilizing the ABQS technique.

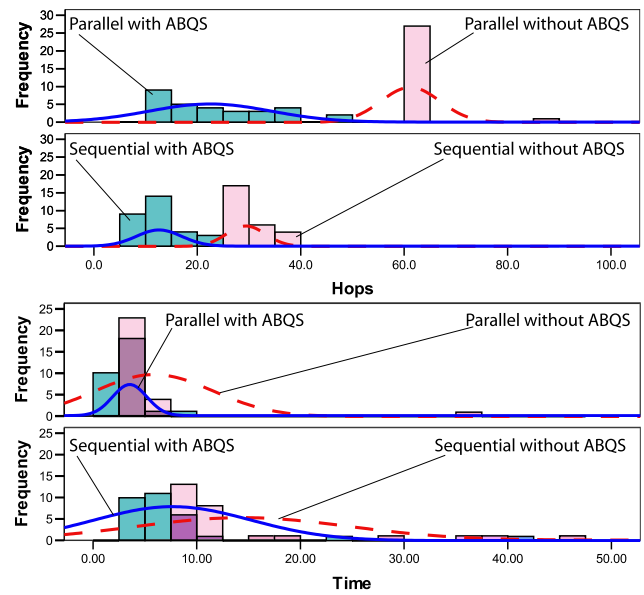


Figure 2: Histogram comparison of the number of hops and the time elapsed for all vote collection types.

4. CONCLUSIONS AND FUTURE WORK

This paper introduced a technique for *agent based quorum sensing* and provided a viable solution to distributed decision-making on a MANET. Greater efficiency was obtained in terms of significant time and bandwidth improvements at the cost of only a slight drop in decision accuracy. These results are not limited to a binary plurality vote. Preliminary results show that these techniques can be applied to other methods of vote collection: Borda count, approval vote, and plurality with multiple outcomes. The authors hope to explore that matter further, along with analyzing the exact role that vote collection order plays in efficiency.

5. REFERENCES

- [1] D. Artz, M. Peysakhov, and W. Regli. Network meta-reasoning for information assurance in mobile agent systems. In Proc. *IJCAI*, pp. 1455–57, 2003.
- [2] F. Bai, N. Sadagopan, and A. Helmy. Important: a framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks. In Proc. *IEEE Comp. and Communications Soc.*, pp. 825–835, 2003.
- [3] V. Cicirello, et. al. Designing dependable agent systems for mobile wireless networks. *IEEE Intell. Sys.*, 19:39–45, 2004.
- [4] J. Kopena, et. al. Service-based computing on manets: Enabling dynamic interoperability of first responders. *IEEE Intelligent Sys.*, 20(5):17–25, 2005.
- [5] S. Pratt, E. Mallon, D. Sumpter, and N. Franks. Quorum sensing, recruitment, and collective decision-making during colony emigration by the ant *leptothorax albigipennis*. *Behavioral Ecology and Sociobiology*, 52:117–127, 2002.
- [6] E. Sultanik, M. Peysakhov, and W. Regli. Multi-agent-based simulation iv. In Proc. *MABS*, 2005.
- [7] A. Wald. *Sequential Analysis*. John Wiley & Son, 1947.