A Distributed Reasoning Approach to Mobile Mesh Network Optimization

Marcos A. M. Vieira, Matthew E. Taylor, Prateek Tandon
Manish Jain, Ramesh Govindan, Gaurav S. Sukhatme, Milind Tambe
Department of Computer Science
University of Southern California
Los Angeles, CA
Email: mvieira, taylorm, tandonp, manishja, ramesh, gaurav, tambe@usc.edu

Abstract—**By using robots as routers, a team of networked robots can provide a communication substratum establishing a wireless mesh network. The mobile mesh network can autonomously optimize its configuration, increasing performance. One of the main sources of radio signal fading in such a network is multi-path propagation, which can be mitigated by moving the receivers distances on the order of a wavelength. In this paper, we measure the performance gain when robots are allowed to make such small movements and find that it may be as much as 300%. Our main contribution is the design of a system that allows robots to cooperate and improve the real-world network throughput via a practical solution. We model the problem of which robots to move as a distributed constraint optimization problem (DCOP). Our study includes using four local metrics to estimate global throughput.**

I. INTRODUCTION

By using robots as routers, a team of networked robots can provide a communication substratum, establishing a wireless mesh network. The mobile mesh network can autonomously optimize its configuration, increasing performance.

Robots might mitigate many of the communications problems present in urban settings, such as relaying signals into shadows and making small adjustments to reduce multi-path effects. One of the main sources of radio signal fading in such a network is multi-path propagation, which can be mitigated by moving the receivers distances on the order of a wavelength. Small movements might increase receive signal power, improving more than any coding scheme [1].

Robots as routers have applications in various settings. As described above, in a infrastructure setting, it can improve performance of wireless mesh network by mitigating multi-path fading. In an infrastructure-less settings, it can be used to form a connection backbone such as in the LANdroids project [2], where the goal is to keep soldiers covered with communications in urban settings. Another interesting application is for Search and Rescue. Small robots could venture where humans can not go to search for survivors of earthquakes, collapsed mines and other disasters. Finally, automatization of finding weak wireless spots.

Our work emphasis on practical real-world solution regards the robots and the network. The paper focuses on practical solutions to realistic scenarios by running experiments with our testbed.

Our paper hypotheses are: is it possible to improve overall network throughput by using robotics router which can only make small movements? if possible, how would you design such a system?

Here we summarize the main contributions of the paper. We verify if indeed small movements can improve network performance and quantify them. We present the design of a practical system which uses the distributed constraint optimization framework to improve communication. A detail study of four local metric to estimate global throughput is also presented.

We consider the scenario where the robots do not need any knowledge of the environment. Robots do not have a map of the physical layout or known obstacle points. There is no topological information about WiFi connectivity or interference maps a priori. It is often difficult to predict signal propagation characteristics within an urban environment. Many factors can impact signal strength such as the angle of incidence, where an emitter is located, and even the building materials.

Robots are not constantly in motion, but are generally static. Robots only execute small movements relative to their neighbors using their local sensing. The problem of accurate robot localization is non-trivial, such as indoors or GPS-denied environments. Autonomous localization is often a challenge, robots are constrained to small movements. Robots only need local localization hardware such as shift encoder and 802.11 commodity radio.

In the next section, we start by explaining multi-path fading and the need for cooperating between robots (Section II). Then, we describe the distributed constraint optimization framework(Section III) and how we map to our problem. We follow by describing our platform, our experimental methodology and our main results on Section IV. Our results are encouraging because, even with relatively simple local algorithms, we can get average performance improvements of 40%.

II. BACKGROUND

Radio signal fading can be attributed to two mutually independent phenomena: multi-path propagation and path loss. We briefly discuss these phenomena and further details can be found elsewhere [3].

Multi-path propagation is a small-scale effect where the distance scales involved are on order of a wavelength.Multipath occurs when a transmitted signal takes more than one path to a receiver, causing the signals to interfere. Interference has either a constructive or destructive effect on the main component depending on whether it arrives in or out of phase.

Path loss is a large-scale effect of propagation in any medium (e.g., air or water), defined by the way in which radio energy is transmitted in the medium of propagation and its resulting loss. This property is also called *slow fading*.

As described previously in [4], if all signal components that reach the receiver are of equal strength, the multi-path fading is called Rayleigh fading, while if there is a line-of-sight (LoS) component that is significantly stronger, we have Ricean fading.

Small movements of a network's radios can help to optimize the network and negate the effect of deep fades (strong destructive interference).It has been suggested [2] that moving $\frac{1}{4}$ to $\frac{1}{2}$ of a wavelength (λ) is sufficient to escape a deep fade. Others [4] use a Rayleigh fading model with data correlated until 0.38 wavelength. In this work, we will move our radios on the order of half wavelength so signals in different locations are uncorrelated and we can (hopefully) escape from deep fades.

Fading for one radio is defined with respect to a single neighbor: a local movement may allow the radio to escape one deep fade, but at the same time introduce a new fad with respect to a different neighbor. Thus, it is critical to coordinate movements to improve the overall throughput. In this paper, we address a series of questions: (1) Is there a sequence of coordinated movements that improves the throughput? (2) How well will using only local information allow us to optimize the network (relative to the global optimum configuration)? To address these questions, we use the framework of the *distributed constraint optimization problems* (DCOPs), to reason about how radios should move to best improve the network.

III. DISTRIBUTED REASONING APPROACH

In this section, we describe the distributed constraint optimization framework and how we map to optimize system throughput.

A. DCOP and D-CEE

A DCOP consists of a set V of n variables, $\{x_1, x_2, \ldots, x_n\}$, assigned to a set of agents (e.g., independent reasoning entities), where each agent controls one variable's assignment. Variable x_i can take on any value from the discrete finite domain D_i. The goal is to choose values for the variables such that the sum over a set of binary constraints and associated payoff or reward functions, $f_{ij} : D_i \times D_j \rightarrow N$, is maximized. More specifically, find an assignment, A, s.t. F(A) is maximized: $F(A) = \sum_{x_i, x_j \in V} f_{ij}(d_i, d_j)$, where $d_i \in D_i, d_j \in D_j$ and $x_i \leftarrow d_i, x_j \leftarrow d_j \in A$. For example, in figure 1, x_1, x_2, and x_3 are variables, each with a domain of $\{0,1\}$ and the reward function as shown. If agents 2 and 3 choose the value 1, the agent pair gets a reward of 9. If agent 1 now chooses value 1 as well, the total solution quality of this complete assignment is 12, which is locally optimal as no single agent can change its value to improve its own reward (and that of the entire DCOP). $F((x_1 \leftarrow 0), (x_2 \leftarrow 0), (x_3 \leftarrow 0)) = 22$ and is globally optimal.

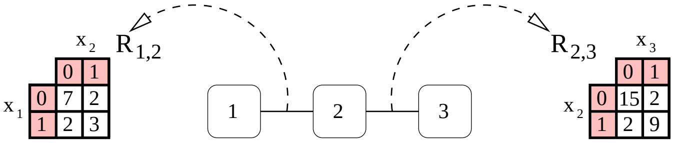


Fig. 1. This figure depicts a three agent DCOP.

In this problem, we model each mobile radio as an agent. Every value an agent can take is one possible physical position for the mobile radio. Constraints are between neighbors in the wireless network. Rewards on the constraints are defined by a local metric, such as the packet reception rate.

We implemented MGM [5] DCOP method and artificially provide agents with the reward for each possible value. Provided such a matrix, this *MGM-Omniscient* algorithm will find a locally optimal assignment of values for all agents, and this gives an upper bound. MGM-Omniscient defines a round as a period in which every agent: (1) communicates its current value to all its neighbors, (2) calculates and communicates its *bid* (the maximum gain in its local reward if it is allowed to change values) to all its neighbors, and (3) changes its value (if allowed). An agent is allowed to change its value if its *bid* is larger than all the bids it receives from its neighbors. At quiescence, no one agent can deviate from the proposed assignment and increase the net reward.

The agents in a DCOP are traditionally assumed to have *a priori* knowledge of the corresponding reward functions. In order to more flexibly model a class of real world domains, we previously introduced *Distributed Cooperative Exploration and Exploitation* (D-CEE) [6] problems, which do not make this assumption. Thus, D-CEE problems appear similar to DCOPs, but with the following features absent from DCOPs: (1) agents initially know the constraint graph but only discover rewards when a pair of agents set their values to explicitly discover a reward value, (2) problems last a set amount of time, and (3) the agents' seek to maximize the on-line global reward over this time horizon T.

The mapping from our network optimization problem onto a D-CEE is similar to that of a DCOP, but now agents must explore different locations to determine the value of local (point-to-point) metrics, we provide a time horizon after which the agents must stop optimizing (to ensure that the network is optimized quickly, and to reduce batter consumption), and the on-line reward is maximized (ensuring that the network will quickly improve, and that it will be performing as well as possible during the optimization).

SE-Optimistic [6], a D-CEE algorithm used in this paper, assumes the maximum reward on each constraint for all unexplored values for agents. Thus, in our domain, it assumes that when moving to a new location, the reward (based on a particular metric) between it and every neighbor is maximized. On every round, each agent bids its expected gain: $\mathit{NumberLinks} \times \mathit{MaximumReward} - R_c$ where R_c is the current reward. The algorithm then proceeds as in MGM-Omniscient. This is similar to a 1-step greedy approach where agents with the lowest rewards have the highest bid. Agents typically explore on every round for the entire experiment.

B. Problem Definition

We model the problem of maximizing throughput as a DCOP/D-CEE problem, where robots must coordinate their movements in a decentralized fashion.

The network workload W consists of a set of multi-hop flows w_{ij} between nodes. The overall global system goal is to maximize the summation of all flows w. Unfortunately, it is not feasible to direct measure the throughput of each flow since it brings large overhead. Thus, it is necessary to estimate the throughput using just local metrics. In this paper, we study four local metrics, each of which defines the reward of a particular agent (and thus how likely it will want to change its position): *minimum SNR* (Signal-to-noise ratio) is the minimum SNR on an agent's links, the *summation of SNR* is the sum of all SNRs on an agent's links, the *minimum PRR* (Packet Reception Rate) is the minimum PRR on an agent's links, and the *summation of PRR* is the sum of all PRRs on an agent's links.

IV. EXPERIMENTAL EVALUATION

To investigate the efficacy of small movements in improving mesh network throughput, we have used physical robots and conducted experiments in an office building. This section describes our platform and our experimental methodology, and then presents results.

A. The Robot Platform

We use a commododized robotics platform and made minimal modifications to it using commercial off-the-shelf products. Our platform consists of an iRobot Create and a small embedded computer mounted on top of it (Figure 2).

Fig. 2. This picture shows a team of iRobot Creates with an Ebox, the platform used for experiments in this work.

The Create, a differential drive robot, has a round chassis with a diameter of 33 centimeters. The robot has two kinds of sensors. First, it has a pair of tactile sensors that, together with a bumper, can help determine if the robot hits an obstacle and the angle at which it does so. Second, it has a suite of infrared (IR) sensors: the bumper contains an IR wall sensor on the right and an omnidirectional IR receiver in the top, and four additional IR sensors mounted underneath the bumper facing down. We do not add additional sensing hardware to the Create.

The embedded computer, the Ebox 3854, is an 800 MHz embedded PC with 256MB shared DDR memory, and supports a 1280x1024 VGA interface, one 10/100 LAN, and USB, mini PCI and compact flash sockets. The embedded computer runs Ubuntu (Linux Kernel 2.6.22) as the operating system. For sensing and control, we developed a Create driver for Player [7].

B. Configuration

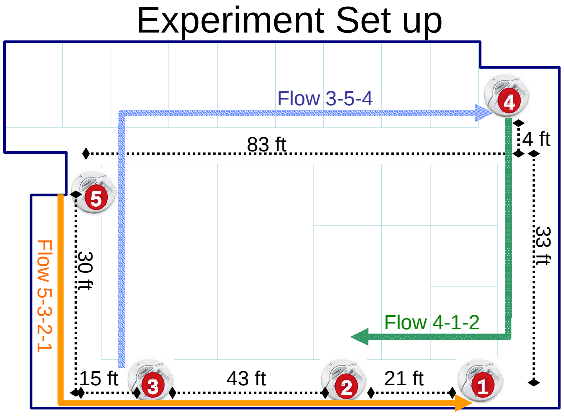


Fig. 3. Initial Configuration of Team of Creates

Our experiments use five robots distributed throughout an indoor office environment as shown in Figure 3. Robots 1, 2 and 3 are within line-of-sight of each other, and the other robots are each not within line-of-sight of any robot. The robots form a ring topology. The simple network optimization solution, move all robots to the center of the ring, is impossible due to the topology of the building.

The robots were configured to use 802.11b, with a 11 Mbits/s data rate (maximum data rate), using 1 dBm transmission power in ad hoc mode. We use channel 14 (which is unused by commercial cards in the US), ensuring that we do not observe external interference.The network was configured with static routing to avoid routing flapping (a router forwards packets via one route then changes to another router) interfering with the measurements.

The network has three multi-hop flows, represented by arrows in figure 3. The flows go through nodes 3-5-4, 4-1-2 and 5-3-2-1. The flows use all the links in the network. Each flow takes 10 seconds for each sampling. *We avoid interference between flows* by creating/measuring the flows sequentially. Flows were created with the *iperf* [8] tool and SNR values were measured (per link) using the *iwspy* Linux command.

C. Using Local Metrics to Improve Throughput

The first set of experiments quantifies the throughput improvement obtainable from small movements, where the robots reason over only local metrics. The total number of possible configurations is exponential with the number of possible robot locations, we constrain robots to only two positions for tractability. The following section will relax this assumption, allowing for more positions per robot.

The five robots experiments yield a total of 32 possible configurations and each configuration is sampled at least three times. Figure 4(a) shows the noise for each agent for the 32 configurations. The noise is constant at -98 dBm, which implies there are no external time-varying radio sources contributing signal interference. Figure 4(b–f) show the SNR per link for each agent for all configurations, as well as the variance of SNR for each configuration. As expected, nodes with non line-of-sight connectivity have lower SNR. Figure 4(g–k) show the Packet Loss Rate (PLR) per link for each agent for all configurations. Links with non line-of-sight connectivity have higher packet lost.

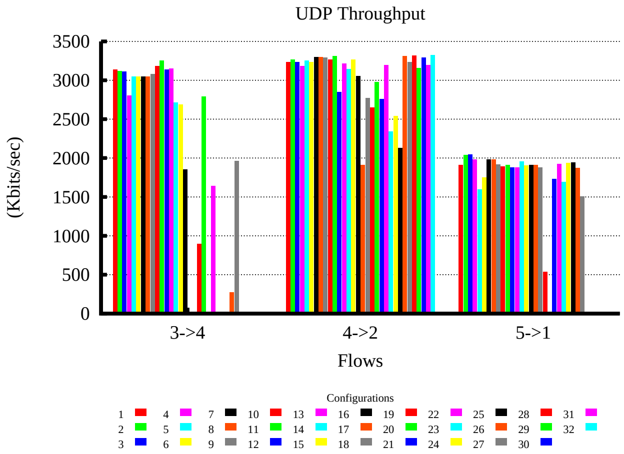


Fig. 5. UDP Flows

Figures 5 and 6 show the multi-hop TCP and UDP flow, respectively, per configuration. Indeed, there is high variability between the flows per configuration in both TCP and UDP flows. Flow 3-4- is the flow with highest variance. Flow 5–1 is the flow with the longest hops and has lowest throughput on average.

In Figure 8 (and 7), we sort the sum of the throughputs of all the TCP (respectively UDP) flows in each configuration. This shows that there is a significant difference across configurations (recall that each configuration can be attained from a starting configuration by small movements of a subset of the 5 robots). There is almost a *2.5x difference* in total throughputs between the best configuration and the worst. Note that our

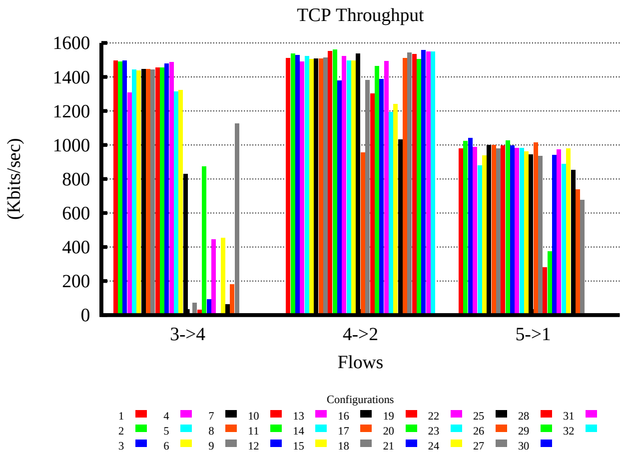


Fig. 6. TCP Flows

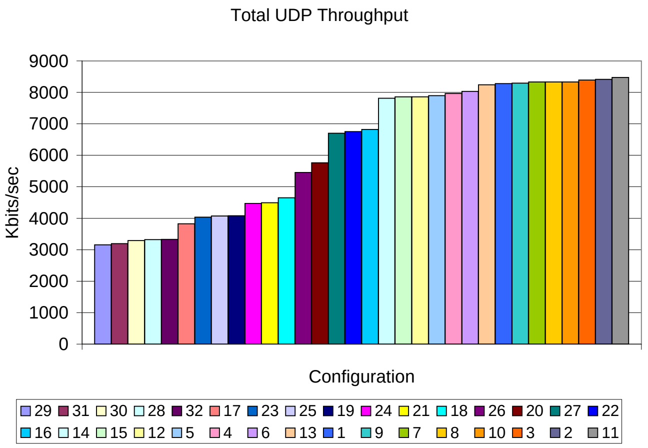


Fig. 7. Sorted Total UDP Flows

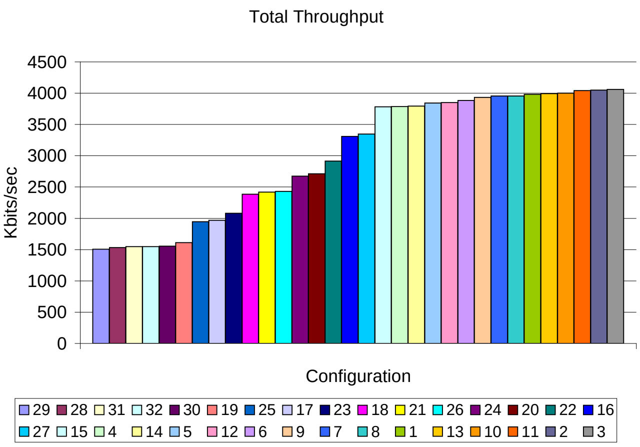
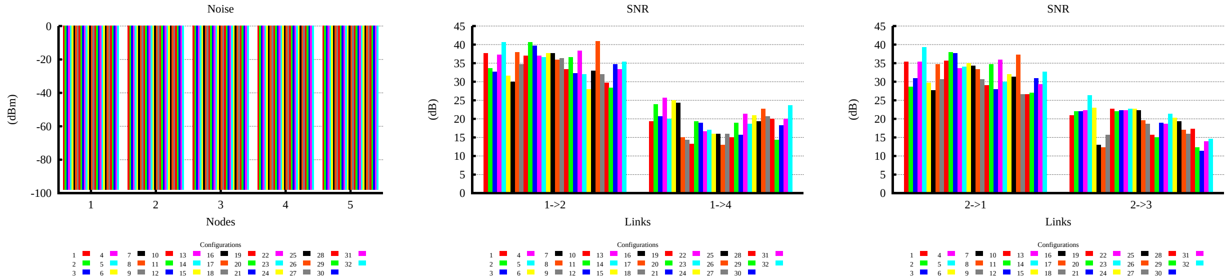


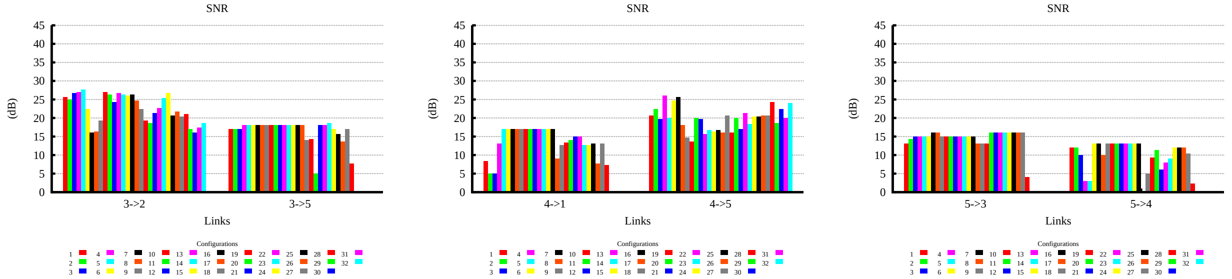
Fig. 8. Sorted Total TCP Flows



(a) Noise

(b) SNR from Robot 1

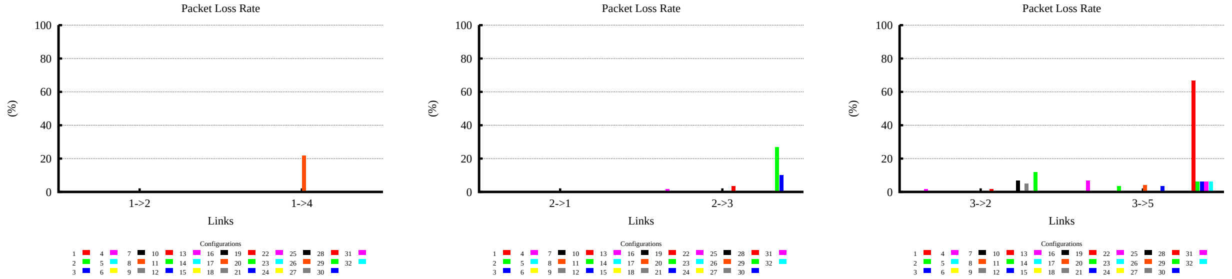
(c) SNR from Robot 2



(d) SNR from Robot 3

(e) SNR from Robot 4

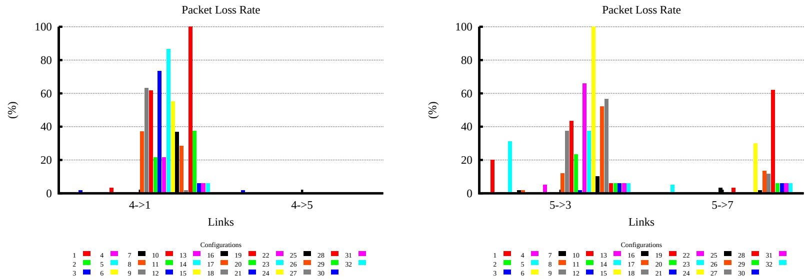
(f) SNR from Robot 5



(g) PRR from Robot 1

(h) PLR from Robot 2

(i) PLR from Robot 3



(j) PLR from Robot 4

(k) PLR from Robot 5

Fig. 4. Local-Metric: Noise, SNR, PLR,for all configurations.

topology has not been especially engineered to achieve this result, which leads us to believe that in other topologies we are likely to see similar performance improvements. If true, it is encouraging, and suggests that a mechanism for coordinated small movements can improve performance significantly.

From this section, we conclude that there is a significant difference in SNR, PRR, UDP and TCP throughput by just doing small movements.

D. Local Metrics

This section evaluates the possible local metrics to be used as a local reward to the coordination algorithms so that the system can achieve a global reward improvement. These results will prove key to understanding the system performance, described in the following section.

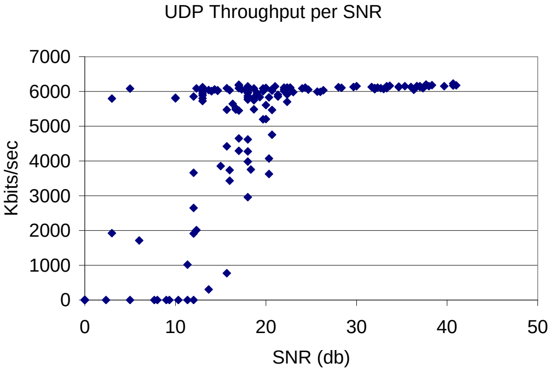


Fig. 9. Relation between SNR and Throughput

Figure 9 shows the 1-hop UDP throughput per SNR value. We can divide the graphics into three areas. Below 10 dB, it is usually not possible to decode packets and the throughput is zero. Between 10 db and 20 dB, it is possible to decode the packets, but the throughput fluctuates significantly. When the SNR is above 22 dB, the throughput is always high (6Mbps). The correlation between throughput and SNR is 0.63. This results are very similar to Zhao [9], which studied 802.15.4 radios and showed the same three behaviors per SNR.

Figure 10 depicts the 1-hop UDP throughput per PRR value. When the PRR is below 0.9, the throughput is low (less than 3Mbps). But, even when the PRR is 1, the throughput may be low. The correlation between throughput and PRR is 0.61.

We can conclude that SNR and PRR do not have weak correlation with throughput (correlation is greater than 0.5) and thus can be used to estimate overall throughput.

In the general case, each node in a network may have many links. Here we propose two metrics to evaluate: *min* and *sum*. The min consists of choosing the minimum value among the links. The sum consists of the summation of the links per node.

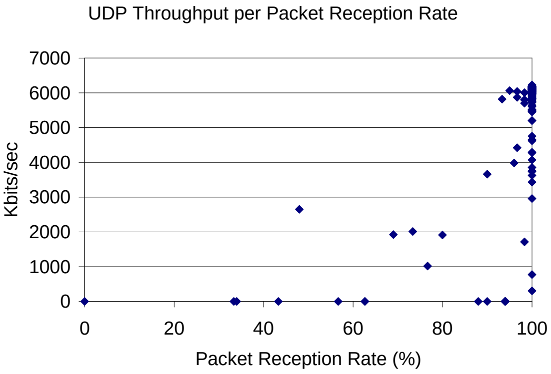


Fig. 10. Relation between PRR and Throughput

Figure 11 illustrates the relation between the multi-hop TCP throughput and the four described local metrics: min PRR, min SNR, sum PRR, sum SNR. In all cases, as the local metrics increase, the total throughput usually increases, but there do exist outliers. In some cases, there are also local minima. For instance, in Figure 11(a), with 40% PRR, the total throughput is 3.5 Mbps. However, as the PRR increases, the throughput temporarily decreases. Such local minima may pose significant difficulty to optimization algorithms.

	PRR	SNR
Min	0.562902083	0.607282598
Sum	0.574420026	0.712466205

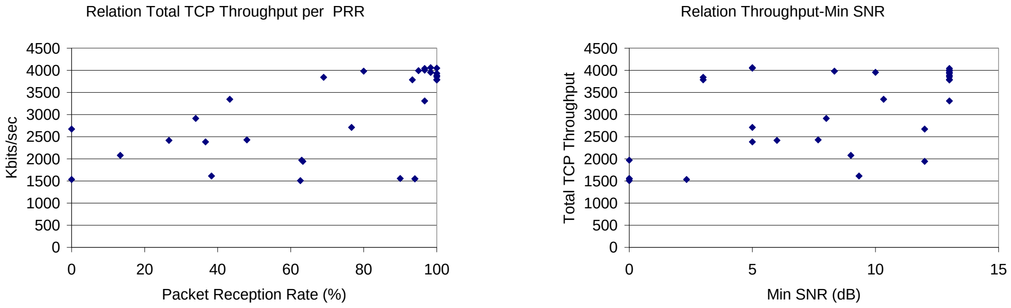
TABLE I
CORRELATION BETWEEN TOTAL TCP THROUGHPUT AND LOCAL METRICS

Table I shows the correlation between the four local metrics and total TCP Throughput. The summation metric has higher correlation than minimum metric. We now focus on the summation metric since it has higher correlation, leading us to believe it is a better metric to use as an estimator.

E. System Improvement

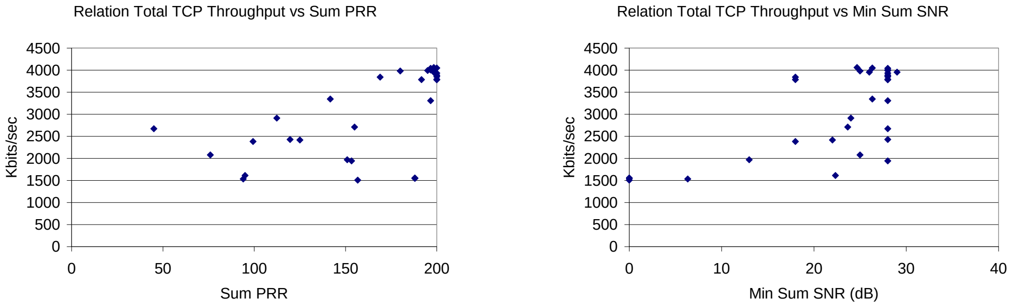
This section evaluates the overall system improvement obtained by using MGM-Omniscient.We focus on the local summation metric with SNR and PRR. Recall that each agent will work to maximize its local reward (in this case, the sum of the SNR or PRR on its links), which will ideally maximize the global metric. Although the agents work to maximize SNR and PRR, this section shows that the corresponding network flows are also maximized, *even though they are not directly measured* by the agents for optimization purposes.

We also extend MGM-Omniscient to support dynamic local metric. In MGM, on every round, all agents bid and the agents who won their bid (per neighborhood) are allowed to move.



(a) a) Minimum PRR

(b) b) Minimum SNR



(c) c) Summation of PRR

(d) d) Summation of SNR

Fig. 11. Relation between total Throughput and local metrics

Initially, MGM assumes that all rewards are known and static. We added the following extension: after agents move, they re-sample their local metrics (because their local metrics could have change even for the agents that did not move). In this way, the reward matrices are updated on every round. We call it Dynamic-DCOP.In our case, all the rewards are known. In our system, on each round all robots sample before bidding. However, such an approach may lead to cycling, requiring agents to timeout, stopping the optimization .

For all possible initial configuration, we execute MGM-Omniscient and dynamic-DCOP and compare the total TCP throughput between initial and final configuration.

Figure 12 shows the average overall improvement for all configuration using static min sum SNR, static min sum PRR, dynamic min sum SNR, dynamic min sum PRR. Allowing agents to sample does not improve the overall system performance. The min-sum-PRR improves the total throughput on average almost 45%. The improvement is not higher because we are using local metric to maximize a global metric and the system might get on a local minima.

Figure 13 shows the total throughput improvement using dynamic min sum PRR for each configuration. In some configurations, there is nearly a 300% gain. Unfortunately, in some configurations, the total throughput could not improve. This is due to the approximation of using local metrics to maximize a global metric, which may result in the system being stuck in a local minima.

F. D-CEE: Optimizing with More Agents

Here we discuss running our system with higher number of positions. We allowed each robot to have five possible positions as illustrated in figure 14.

We use the SNR metric since this does not include any overhead to the network.

We use the formula:

$$MAX - SIGNAL - STR * NumNeighbors - SumOfSNR$$

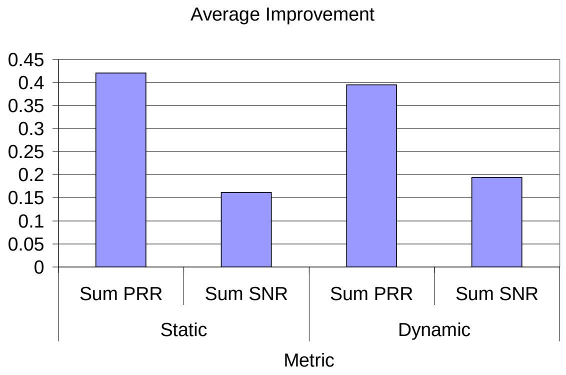


Fig. 12. Improvement per configuration using sum PRR metric

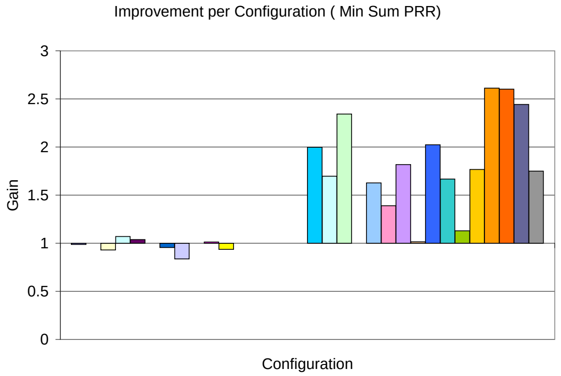


Fig. 13. Improvement per configuration using sum PRR metric

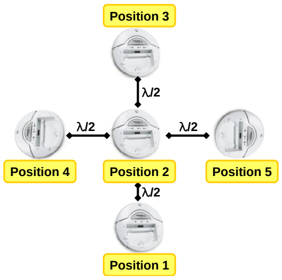


Fig. 14. Positions

Since the number of agents is constant for everyone, MAX-SIGNAL-STR×(NumNeighbors) is also constant. We are maximizing the bid but its effectively making the robot with the worst SumOFSNR move.

Figure 15 shows the average total TCP Throughput improvement per round, where robots were allowed to move 5 positions. This results are encouraging because, even with relatively simple local algorithms, and using small movements, we could improvement performance by almost 40%.

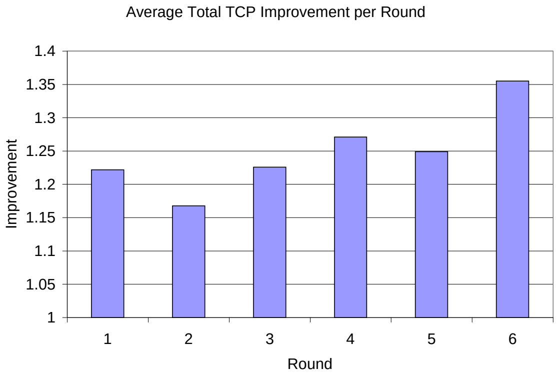


Fig. 15. Improvement per round with Total TCP Flows

V. RELATED WORK

Using small movements to combat the multi-path fading effects in complex environments has promise and this paper is not the first to examine such effects. In [10] [1], they showed that small movements can increase receive signal power, improving more than any coding scheme. Their focus was on a pair of nodes. We, on the other, focus a high-level view to improve global network throughput, where coordination is necessary. We also evaluate different metrics, SNR and PRR and how it affects global throughput.

Existing theoretical work[11] shows that mobility increases capacity with random source-destination pairs with loose delay constraints. Other work includes [4], where the authors propose a methodology for exploiting multi-path fading by controlling the robot according to radio signal strength. They solve the problem of how many samples are needed for given communications performance and how they should be spaced. They provide lower bounds on the number of samples for a single robot. Using 802.15.4 radio, they also show there is room for improvement (as much as 20 dB in RSSI).

To overcome environment interference, the authors of [12] consider the problem of controlling a team of robots to ensure end-to-end communication. They propose two different metrics, point-to-point signal strength and data throughput, to monitor the network connectivity of the system. Even ad-hoc communication protocols pose difficult challenges during multi-robot experimentation, as shown by the authors of [13]. Their focus is not on small movements, they need a map of the environment and their goal is not to optimize network throughput.

Complementary to our work, [14] discusses a game theory dynamic programming algorithm to guarantee connectivity of a mobile user to an access point by moving a a team of robotic routers. The problem is modelled as a pursuit-evasion game, with the goal of finding the shortest escape trajectory since user could move in an adversarial trajectory.

Avoiding deep fade can be combined with other modern radio techniques such as multiple-input multiple-output (MIMO) [15].

In [6], the D-CEE framework is presented to study the problem of how to coordinate mobile nodes to maximize the cumulative RSSI. The paper focus on algorithms to study the trade-off between exploration and exploitation. We, on the other hand, focus on different local metrics (SNR, PRR) and how it affects the overall network.

In addition to the DCOP work discussed in earlier sections, previous work in distributed constraint reasoning in sensor networks [16], [17] uses a precursor method to the DCOP formulation which does not handle unknown reward matrices. Marder et al. [18] formulate dynamic sensor coverage as a "potential game," which is similar to a DCOP. However, like other DCOP work, the reward matrix is known, there is no time limit, and only final reward is considered. Cheng et al. [19] suggest an approach for coordinating a set of robots based on swarm intelligence, however the objective of the work is to disperse the robots evenly within a specified shape, and not to optimize the signal strengths across the network.

Correll et al. [20] look at optimizing a wireless network of mobile robots using a distributed swarm optimization, but are concerned with changing the topology (i.e., neighbors) of the network rather than optimizing signal strength. Gerkey et al. [21] address a similar problem, but use auction mechanism and the goals of agents are significantly different (agents modify the topology of the network and on-line reward is not emphasized). Farinelli et al [22] perform decentralized coordination on physical hardware using factor graphs, however, rewards are known and cumulative reward is not considered.

VI. CONCLUSION

In this paper, we demonstrate that mobile robots can be used successfully in a mesh network. With robotic routers forming a network, nodes can avoid deep face caused by multi-path fading. Our study shows that small movements can improve network performance and that the total network throughput could vary as much as 300% when the robots moved on the order of half a wavelength. Avoiding deep fades is a pairwise problem between sender and receiver, which requires coordination. Thus, we designed a practical system which uses the distributed constraint optimization framework to improve communication. We studied four local metric (min SNR, min PRR, sum SNR, sum PRR) to estimate the global throughput. Our results are encouraging because we can achieve an average global performance improvement of 40% while maximizing only local metrics.

REFERENCES

[1] J. Smith, M. Olivieri, A. Lackpour, and N. Hinnerschitz, "RF-Mobility Gain: Concept, Measurement Campaign, and Exploitation," *IEEE Wireless Communications*, vol. 16, no. 1, pp. 38–44, February 2009.

[2] "Broad Agency Announcement 07-46 LANdroids." [Online]. Available: http://www.darpa.mil/ipto/programs/ld/ld.asp

[3] J. S. Seybold., *Introduction to RF Propagation*. Wiley - Interscience, 2005.

[4] M. Lindhé, K. H. Johansson, and A. Bicchi, "An Experimental Study of Exploiting Multipath Fading for Robot Communications," in *Robotics: Science and Systems*, Atlanta, GA, USA, 2007.

[5] J. Pearce and M. Tambe, "Quality guarantees on k-optimal solutions for distributed constraint optimization," in *IJCAI*, 2007.

[6] M. Jain, M. E. Taylor, M. Yokoo, and M. Tambe, "DCOPs Meet the Real World: Exploring Unknown Reward Matrices with Applications to Mobile Sensor Networks," in *Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI)*, July 2009.

[7] B. P. Gerkey, R. T. Vaughan, K. Støy, A. Howard, G. S. Sukhatme, and M. J. Mataric, "Most Valuable Player: a Robot Device Server for Distributed Control," in *Intl. Conf. On Intelligent Robots and Systems*, vol. 3, Maui, HI, USA, 2001, pp. 1226–1231.

[8] A. Tirumala, F. Qin, J. Dugan, J. Ferguson, and K. Gibbs, "Iperf," http://dast.nlanr.net/Projects/Iperf/.

[9] J. Zhao and R. Govindan, "Understanding Packet Delivery Performance In Dense Wireless Sensor Networks," in *Proceedings of the ACM Conference on Embedded Networked Sensor Systems*, Los Angeles, CA, November 2003.

[10] M. Lindhé and K. H. Johansson, "Using robot mobility to exploit multipath fading - wireless communications in networked robotics," *IEEE Wireless Communications*, vol. 16, no. 1, pp. 30–37, February 2009.

[11] M. Grossglauser and D. N. C. Tse, "Mobility Increases the Capacity of Ad Hoc Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 10, no. 4, pp. 477–486, August 2002.

[12] M. A. Hsieh, A. Cowley, V. Kumar, and C. J. Taylor, "Towards the deployment of a mobile robot network with end-to-end performance guarantees," in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, 2006, pp. 2085–2091.

[13] F. Zeiger, N. Kraemer, and K. Schilling, "Commanding Mobile Robots via Wireless Ad-hoc Networks - A Comparison of Four Ad-hoc Routing Protocol Implementations," in *ICRA*, 2008, pp. 590–595.

[14] O. Tekdas and V. Isler, "Robotic Routers," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2008.

[15] A. J. Paulraj, D. Gore, R. U. Nabar, and H. Blcskei, "An Overview of MIMO Communications: A Key to Gigabit Wireless," in *Proc. IEEE*, 2004, pp. 198–218.

[16] V. Lesser, C. Ortiz, and M. Tambe, *Distributed sensor nets: A multiagent perspective*. Kluwer Academic Publishers, 2003.

[17] W. Zhang, Z. Xing, G. Wang, and L. Wittenburg, "An analysis and application of distributed constraint satisfaction and optimization algorithms in sensor networks," in *AAMAS*, 2003.

[18] J. Marden, G. Arslan, and J. Shamma, "Connections between cooperative control and potential games illustrated on the consensus problem," in *European Control Conference*, 2007.

[19] J. Cheng, W. Cheng, and R. Nagpal, "Robust and self-repairing formation control for swarms of mobile agents," *AAAI*, 2005.

[20] N. Correll, J. Bachrach, D. Vickery, and D. Rus, "Ad-hoc wireless network coverage with networked robots that cannot localize," in *ICRA*, 2009.

[21] B. P. Gerkey, R. Mailler, and B. Morisset, "Commbots: Distributed control of mobile communication relays," in *AAAI Workshop on Auction Mechanisms for Robot Coordination*, 2006.

[22] A. Farinelli, A. Rogers, A. Petcu, and N. Jennings, "Decentralized coordination of low-power embedded devices using the max-sum algorithm," in *AAMAS*, 2008.