Proceedings of the 2005 IEEE
International Conference on Robotics and Automation
Barcelona, Spain, April 2005

An Efficient Group Communication Protocol for Mobile Robots

Saumitra M. Das, Y. Charlie Hu, C.S. George Lee, and Yung-Hsiang Lu
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907-2035
{smdas, ychu, csglee, yunglu}@purdue.edu

Abstract— **Mobile robot teams have many useful applications such as search and rescue, exploration and hazard detection and analysis. Communication between the robots of a team as well as between the robots and a human operator or controller are useful for many applications. Many applications of mobile robots involve scenarios in which no communication infrastructure such as base stations exist (e.g. demining in battlefields) or the existing infrastructure is damaged (e.g. search and rescue after an earthquake). In such scenarios, it is necessary for mobile robots to form an ad hoc network to enable communication by forwarding each other's packets.**

In many applications, group communication can be used for flexible control, organization, and management of the mobile robots. Multicast provides a bandwidth efficient communication method between a source and a group of robots. In this paper, we propose an efficient multicast protocol MRMM (Mobile Robot Mesh Multicast) for deployment in mobile robot networks. MRMM exploits the fact that mobile robots know what velocity they are instructed to move at and for what distance in building a long lifetime sparse mesh for group communication that is more efficient. Our results show that MRMM provides an efficient group communication mechanism that can potentially be used in many mobile robot application scenarios.

I. INTRODUCTION

Communication between mobile robots is useful and even critical in many applications. In many mobile robot applications, communication infrastructure may be damaged or not present requiring the mobile robots to form an ad hoc network using each other as forwarding nodes to enable communication. Group communication among the mobile robots thus requires protocols that can operate without central control and handle dynamic topology changes due to the mobility of the mobile robots. Multicast is the most important group communication primitive and is critical in applications where close collaboration of teams (e.g. rescue teams, search teams) are needed. It is very useful when audio, video, images, and other such data need to be shared among team members. Multicast provides an efficient means of sending the same data to multiple recipients. Compared to multiple unicasts, multicast minimizes the link bandwidth consumption, sender and router processing, and delivery delay. Since multicast involves minimizing the number of transmissions in order to reach a group of receivers, it is effective in minimizing the energy consumption of group communication in a network of mobile robot teams.

The use of multicast as a group communication primitive in mobile robot networks could be envisioned in the following three scenarios: Firstly, multicast messages in mobile robot networks could be from one mobile robot to a set of other mobile robots. For example, in a large mobile robot team, only some subset of the nodes may be equipped with GPS or other localization equipment. These nodes could be organized into a multicast group so as to periodically notify each other of their positions and act as landmarks and navigation guides for other robots that do not know their positions. As another example, robots that have video cameras could communicate with other robots in the 'cam' group to build a global map. Multicast groups created based on capabilities can also be used by a robot to search for other robots with a specific capability or resource. In summary, all robots with similar capabilities could organize themselves into a multicast group to enable efficient cooperation and coordination.

Secondly, multicast messages may also need to be sent from human operators to a group of mobile robots in order to efficiently manage them. For example, the human operator may want to communicate with and command only the group of robots that have a specific chemical sensor. As another example, the human operator may want to organize the robots whose remaining energy is above a certain threshold into a high availability group in order to manage them. Thus, a team of robots could be partitioned into subteams using multicast groups for ease and flexibility of management and control. These subteams can also be grouped together, split apart, or hierarchically organized depending on the scenario and application.

Finally, multicast messages may also be sent from a mobile robot to multiple controllers or sinks similarly as in a sensor network. For example, a team of demining robots may need to communicate its findings to a group of platoon commanders outside the minefield. Thus multicast can also be used for data dissemination.

Many multicast protocols have been designed for use in mobile ad hoc networks (e.g. MAODV [10], ODMRP [4], MCEDAR [11], DDM [3]). However, the multicast pro-

---

[1]This work was supported in part by the National Science Foundation under Grant IIS-0329061.

tocols proposed for the general mobile ad hoc networks (MANETs) do not exploit two unique characteristics of mobile robot applications which do not exist in typical MANETs: (1) Mobile robots are typically mission-oriented and have predictable and planned mobility. This is in contrast to a human's movement which are fairly random. (2) Mobile robots frequently pause at various points and often the duration of the pause can be estimated or is known beforehand. For example, if a robot pauses to measure some environmental parameter, the amount of time taken for the task can typically be estimated or known in advance.

In this paper, we propose exploiting the stationarity characteristics of mobile robot applications to improve multicast group communication. Specifically, we propose and design the Mobile Robot Mesh Multicast (MRMM) protocol for efficient multicast in mobile robot applications. MRMM is based on the ODMRP [4] protocol designed for MANETs. MRMM exploits the fact that mobile robots know what velocity they are instructed to move at and for what distance in order to build a long-lifetime sparse mesh for group communication. The sparseness of the mesh reduces the overhead significantly while the long lifetimes of nodes that are part of the mesh allow reliable packet delivery and increased fault tolerance. We evaluate the performance of MRMM and compare it to ODMRP under a mobile robot application scenario.

II. Network Model and Scenario

In this paper, we consider an application scenario where many robots are used to form a mobile robot network. Each robot has a simple sensory ability and limited computational power. This makes it practical to build a large number of such robots. The communication among the robots is based on wireless ad hoc communication (IEEE 802.11) in which robots forward each other's packets when the source and destination nodes of a packet are not within direct reach. This reflects our mobile robot testbed in which Pioneer mobile robots [1] are equipped with laptops and PDAs and use Orinoco Wavelan wireless cards to communicate with each other. MRMM does not require GPS equipment or other localization protocols since it operates without position information.

Mobile robot networks have the following unique mobility characteristics different from traditional MANETs. Consider a paused (stationary) robot. This robot could be waiting for commands or busy doing a task, e.g., detecting chemicals. The pause duration (T) is thus composed of a period of time t to complete the current task followed by a random period of time $T - t$ after which the robot moves again. We assume a set of k tasks that the robot can undertake each with a different task duration. Consider a robot currently in motion. In most applications, the robot is issued instructions on which direction to move and at what speed. For example, these values could be determined by an energy-efficient motion planning algorithm as proposed in [6]. Thus the robot knows a distance D to travel in a particular direction. Also, without GPS support a robot can measure with some error the distance d already traveled and thus the distance $D - d$ which we denote as the distance-to-rest (d_{rest}).

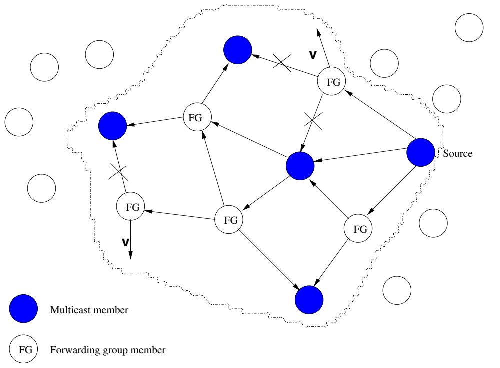


Fig. 1. MRMM mesh construction and pruning. The source first builds the mesh to reach all the group members (shaded) using the forwarding group (marked FG), the FG nodes are then pruned and outgoing edges from pruned FG nodes are removed.

Thus we assume that the robot can estimate its d_{rest} when mobile, the time remaining to complete a task t, and its current set velocity v.

III. Mobile Robot Mesh Multicast

MRMM (Mobile Robot Mesh Multicast) is based on the ODMRP (On Demand Multicast Routing Protocol) [4] multicast protocol developed for MANETs. MRMM is an extension of the ODMRP protocol with specific features for efficient operation in mobile robot applications. Like ODMRP, MRMM also uses a mesh to enable redundancy and consequently more reliable delivery as well as to avoid the drawbacks of tree maintenance in mobile networks. Similar to ODMRP, MRMM also consists of two major phases: (1) Mesh construction and maintenance: A mesh is created using a subset of the mobile robots that are a part of the network. The mesh is a structure in the network such that all group members are part of the mesh and certain number of non-members are recruited to forward packets so that no disconnections occur and some redundancy is present. This mesh has to be dynamically reconfigured and adaptable to disconnections due to mobility. (2) Data delivery: Data packets are broadcast by robots that are part of the mesh so as to be received by all the group members at that point in time.

A. *Mesh Construction and Pruning*

Similar to ODMRP, when a multicast source in MRMM has packets to send, but has no mesh set up for the multicast group, it floods a JOIN QUERY control packet through the entire network. When an intermediate node receives the JOIN QUERY packet, it stores the source ID and the sequence number in a data structure to detect any potential duplicates. The routing table is updated with the appropriate node ID (i.e. backward learning) from which the message was received for the reverse path back to the source node. If the message is not a duplicate and the TTL is greater than zero, it is rebroadcast. When the

JOIN QUERY packet reaches multicast group members, it creates and broadcasts a JOIN REPLY to its neighbors. When the JOIN REPLY is received by a node, it checks if the next hop ID of one of the entries in the JOIN REPLY packet matches its own ID. If this is true, the node realizes that it is on the path to the source and becomes part of the forwarding group by setting the Forwarding Group (FG) flag. In this way, each FG member propagates the JOIN REPLY until it reaches the multicast source via the selected path (shortest). This whole process constructs (or updates) the routes from sources to receivers and builds a mesh of nodes, i.e., the forwarding group. Because the nodes maintain soft state, the JOIN QUERY packets have to be periodically flooded to refresh membership, maintain the mesh and rebuild the parts of the mesh that have disconnections due to node mobility. This is also done on-demand, i.e., when the source has data to send. Thus if we denote the set of nodes (robots) that make up the entire mobile robot network as G, then a JOIN QUERY selects from among these G nodes a set of nodes $F \subseteq G$ that are on the path from the source to all the group members. The set of group members is denoted as M. Figure 1 depicts the set of nodes F which is comprised of the nodes labeled FG and the set of group members. Note that $F \cap M \neq \phi$.

In contrast to ODMRP, MRMM exploits the mobility knowledge present in mobile robot networks, i.e., the knowledge of d_{rest}, v and t as mentioned in Section II in order to run a mesh pruning algorithm. The objective of the pruning algorithm is to select a new set of nodes $P \subseteq F$ that maximizes the lifetime of the mesh without greatly affecting the redundancy and path lengths of the resulting mesh formed by the set of nodes P. Note that $M \subseteq (F \cap P)$. This new set of nodes P are the newly selected, smaller set of forwarding group nodes. The nodes that are removed from F to get P are selected to be the nodes that are most likely to break apart from the mesh. Since P is generally smaller than F, the number of rebroadcasts and consequently the overall control overhead will be reduced in MRMM. For example, in Figure 1, the nodes depicted with a velocity vector are moving away from the mesh and the mesh pruning algorithm will select these nodes to be removed from F. Thus as shown in Figure 1, those nodes will not rebroadcast the JOIN QUERY, reducing the number of JOIN QUERY transmissions. Another important consequence of this is that the data packets will travel over a sparser mesh resulting in lower number of data transmissions required to deliver all the data packets. Thus, MRMM will have an improved forwarding efficiency.

However, the goal is to reduce the set of forwarding group members without affecting other factors like reliability and packet delivery performance. Making the mesh too sparse may affect the reliability of the mesh resulting in lower packet delivery performance. Thus, MRMM uses ω , the mesh sparseness factor to control how aggressively nodes are removed from the set F. In general, a sparser mesh would be more unreliable and have lower packet delivery performance. However, since MRMM constructs the mesh out of the least volatile nodes, even a sparser mesh provides similar packet delivery performance as will be demonstrated by the results.

In order to enable mesh pruning in a distributed manner, each node, upon receiving a JOIN QUERY packet, forwards the packet (and thus includes itself in the forwarding group) probabilistically. A node i will uniformly choose a random number between 0 and 1. If this value is less than P_i, it will rebroadcast the packet. Otherwise it will silently absorb the packet and not become part of the mesh.

$$P_i = (\frac{1}{v})^{min(\omega, \frac{d_{rest}}{R})}$$

where, R is the radio range and d_{rest} is the distance to rest as specified in Section II. Thus, ω controls how sparse the mesh will be. If it is large the set P will be smaller and the mesh sparser. In the simulation, we use a value of 2.5 as a heuristic for ω. The factor $\frac{d_{rest}}{R}$ increases as the distance to rest normalized by the radio range increases. Note that ω is used to restrict the amount by which the mesh is thinned due to the value of d_{rest}.

B. Data Delivery

Data is delivered in a similar manner as in ODMRP. When receiving a multicast data packet, a node that is a member of the FG forwards only non-duplicate packets. Nodes in the forwarding group are demoted to non-forwarding nodes if not refreshed before they timeout. Thus, over time the mesh will disappear if not used.

In contrast to ODMRP, each node checks if it has moved a threshold distance (500m) away from the time the mesh state was set up for that data source. It then stops broadcasting the data packets for that source even if they arrive from some other point in the mesh. This feature also helps to improve the forwarding efficiency of MRMM.

C. Incorporating Mobility Prediction

If we assume that the mobile robots are GPS enabled, each robot can easily calculate how soon it will move out of range of its neighbors. Thus, if we model the mesh as a graph, we will know the lifetimes of each edge in the graph. The source can learn the minimum of these lifetime values during the mesh construction phase. The source can then set the JOIN QUERY periodic interval to the minimum timeout value of the mesh. This has been previously proposed in [5]. However, in our network we do not assume GPS availability since the robots may be indoors and GPS equipment may not be economical to put on all robots. Thus, in our scenario timeouts need to be estimated by assuming that each of a node's neighbors is exactly some distance away (say 125 m or half the radio range).

Inclusion of such timeouts can potentially further reduce the overhead of MRMM. However, with a sparse mesh there are subtle tradeoffs involved. For example, if even one node is moving fast, the JOIN QUERY packets will have to be sent at a higher rate than before, resulting in higher overhead. Another issue is the amount of error that may occur in timeout estimation without GPS. However, this technique is orthogonal to the main focus of this paper. Inclusion of such a prediction technique into MRMM and its evaluation is part of our future work.

IV. EVALUATION METHODOLOGY

We use the Glomosim simulator [12] to evaluate the performance of MRMM. We implemented MRMM in Glomosim and compare it to ODMRP. We simulated a network of 100 robots randomly distributed in an area of $1000m$x$1000m$. A simulation duration of 500 seconds is chosen.

Mobility pattern: The mobility model used is as follows. As the simulation starts, each robot chooses a random destination in the given area and starts moving towards the chosen destination with a speed chosen uniformly between 1 and 5 meters/second. Note that the speed chosen is lower than those considered for MANET multicast protocol comparison studies (typically 10-20 m/s) since they assume faster participants (e.g. cars). Once the robot reaches the destination, it chooses a random task from a set of tasks with a maximum task duration of p (pause time) seconds and pauses for the task's duration. After pausing it chooses the next random destination. This effectively models the movement of a group of robots performing tasks. The robot moves towards a particular area and spends some time at the location performing a task. The robot then moves to the next position. Although the mobility of robots in the simulation is random, the robots know their velocity, task duration and d_{rest}. We have considered pause times ranging from 900s to 0s and a simulation duration of 900s. Note that the random task selection is not done for p=900s and 0s. A pause time of 900s in a simulation of duration 900s signifies that no robots move. A pause time of 0 second signifies that the robots are in constant motion and do not stop to perform any tasks.

Traffic pattern: Each robot that is a multicast source sends multicast data packets to a group. Each packet sent is 512 bytes in size and the packet rate is 1 packet every second per source.

Performance Metrics: The following metrics are measured in comparing the protocols: (1) Control overhead: Number of control packets transmitted by the multicast protocol. In both protocols these consist of JOIN QUERY, JOIN REPLY, and JOIN ACK packets. (2) Multicast delivery ratio (MDR): Fraction of multicast data packets originated by the source that are received by the receivers. For example, in a multicast group with K members, each originated packet needs to be received by K members in order for the MDR to be 100%. (3) Average delivery latency (Delay): Packet delivery latency averaged over all of the multicast packets delivered to all receivers. (4) Forwarding efficiency (FE): Average number of data packet transmissions per delivered data packet. This is an indicator of the bandwidth efficiency of the protocol since the lower the FE, the lesser is the amount of bandwidth used to deliver a set of data packets.

V. EVALUATION RESULTS

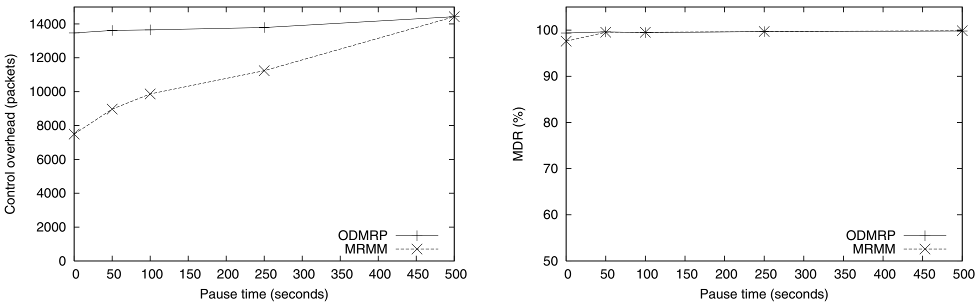
In this section, we first compare MRMM with ODMRP using the four metrics discussed above for a network of 100 robots. We first examine the effect of varying mobility on both protocols. We then investigate the effect of increasing the size of the multicast group on the performance of the two protocols.

A. *Effects of Mobility*

In this section, we vary the pause time from 0 to 500 seconds to simulate the conditions of a continuously mobile to a completely static mobile robot network. We assume a single multicast group in the network with 10 members. An example of such a scenario is a human operator or a controller sending control messages to all 10 robots with video cameras which form a multicast group.

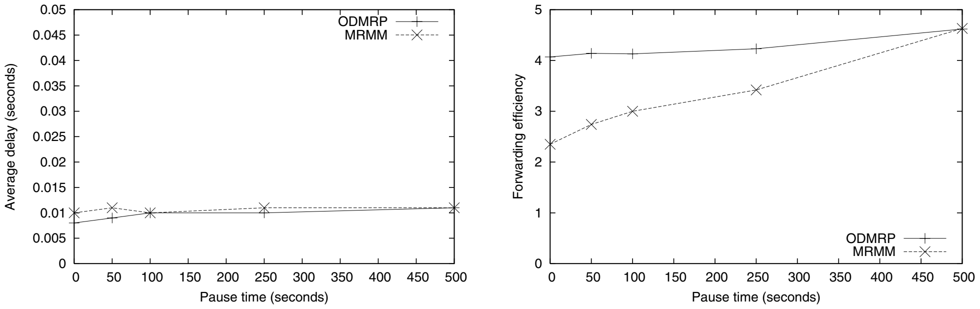
The performance results are depicted in Figure 2. The first observation is that the control overhead of MRMM is significantly lower than that of ODMRP for a wide range of pause times. As seen in Figure 2(a), for the continuous movement scenario, the overhead of MRMM is 44% lower than that of ODMRP. The control overhead remains lower for a wide range of pause times and finally is equal when the network is completely static. The control overhead reduction in MRMM is primarily due to a reduction in the propagation of JOIN QUERY packets. Since nodes that are likely to break the mesh are less likely to rebroadcast the JOIN QUERY, the number of total transmissions of JOIN QUERYs is reduced, which leads to reduced control overhead. In addition, since the mesh is composed of a smaller set of nodes, the number of JOIN ACKS transmitted is also slightly reduced. For a static network with pause time 500 seconds, MRMM cannot discriminate which nodes are likely to break the mesh and consequently all nodes rebroadcast JOIN QUERYs resulting in the performance being similar to ODMRP. Even in this scenario, schemes could be designed to reduce the size of the mesh. However, that is orthogonal to the basic motivation of MRMM which is to find the set of nodes that maximize the lifetime of the forwarding mesh.

More importantly, Figure 2(b) shows that despite having low overhead, MRMM delivers as many multicast packets as ODMRP. This is interesting since the mesh formed by MRMM is sparse and consequently more susceptible to failures due to mobility. However, by finding the set of long-lifetime nodes, the MRMM mesh is more resilient to failures from mobility which balances out the lower redundancy in the mesh. This results in the MDR of MRMM being close to that of ODMRP. Additionally, Figure 2(c) shows that the delay performance of both protocols are similar. ODMRP delivers packets with slightly



(a) Control overhead

(b) MDR



(c) Average delay

(d) Forwarding efficiency

Fig. 2. Performance comparison with varying mobility.

lower delay since with a larger mesh, in many cases, it is able to deliver packets using shorter transmission paths. Another significant benefit of MRMM is the improved forwarding efficiency. The forwarding efficiency measures the bandwidth usage of a multicast protocol by indicating how many transmissions of a data packet are required on average to deliver it. Since MRMM builds a sparser mesh, the number of times a data packet is transmitted is lower thus improving the forwarding efficiency. For example, at a pause time of 100 seconds, MRMM requires 27% fewer transmissions per successfully delivered data packet than ODMRP while at 0 second the FE is lower by 42%.

In summary, MRMM reduces the control overhead and improves forwarding efficiency for a wide range of mobility scenarios in comparison to ODMRP.

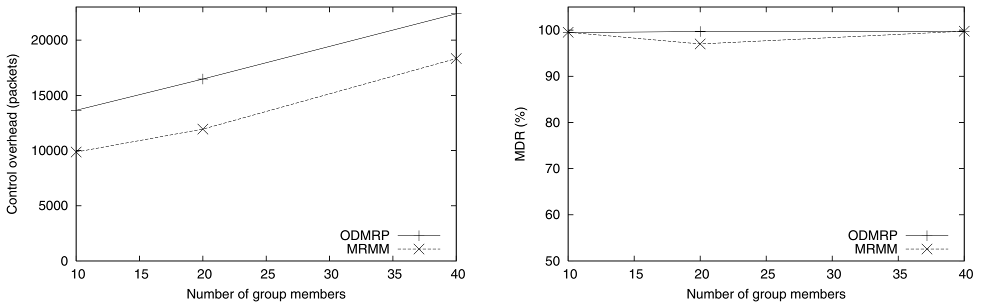
B. Effects of Group Size

In mobile robot networks, it is likely that different multicast groups may have different sizes. Some groups may be large, e.g., the group consisting of all robots with sonars, whereas some groups with more costly equipment may be small, e.g., the robots equipped with video cameras. Also the group size may vary over time. For example, if a group consisting of high energy robots is created, its size is likely to vary over time. Thus, it is important to study the performance of multicast protocols with varying group sizes.

In this section, the pause time considered is 100 seconds. Only one group exists in the mobile robot network while the group size is varied from 10 to 40.

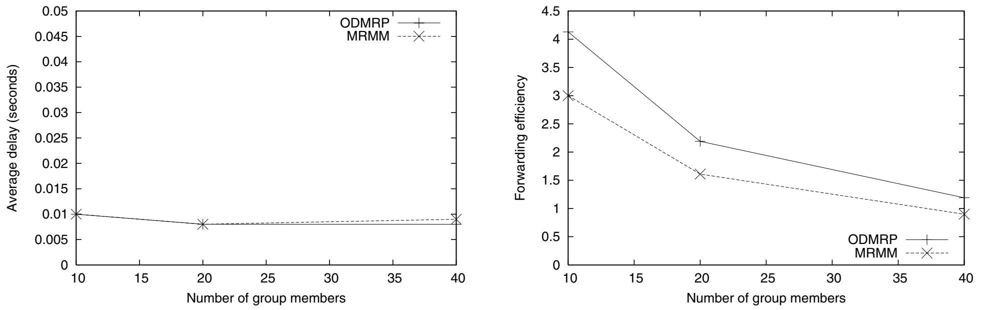
The results are depicted in Figure 3. The results show that the overhead of both protocols grow with the group size. This is because an increase in the number of members results in an increase in transmissions of JOIN QUERY and JOIN REPLY packets. MRMM has consistently lower control overhead than ODMRP even as the group size increases. Figure 3(b) shows that the MDR of MRMM is comparable to that of ODMRP despite its lower overhead. This is significant since as the group size increases, it is possible that using a sparser mesh in MRMM may result in a lower rate of successful delivery. However, since the nodes in the MRMM mesh are on average more reliable than the ones in the ODMRP mesh, the packet delivery ratio remains similar.

The delay performance of both protocols are also similar as the group size is increased. The forwarding efficiency of both protocols improves (decreases in value) as the group size is increased. This is expected since as the number of members increases, more and more members are likely to be reached with the same number of transmissions. This is because MRMM, like ODMRP is a broadcast based protocol and incorporates the use of wireless multicast advantage. Wireless multicast advantage allows multiple members to receive packets using a single broadcast transmission in the mesh. Thus as the number of members increases, the number of data transmissions required per successfully delivered data packet decreases. Note that the forwarding efficiency of MRMM is superior to that of ODMRP although the gap reduces as the group size is increased. MRMM actually achieves a forwarding efficiency below 1 for large group sizes which indicates that less



(a) Control overhead

(b) Multicast Delivery Ratio



(c) Average delay

(d) Forwarding efficiency

Fig. 3. Performance comparison with varying group size.

than 1 data packet transmission is required per successfully delivered data packet.

In summary, MRMM consistently has lower control overhead and higher forwarding efficiency than ODMRP for various group sizes while maintaining similar delivery ratios and delay performance.

VI. RELATED WORK

In [9], the authors propose the use of autonomous communication relays for tactical robots. They further discuss the use of such relays for allowing tactical ground robots to communicate with a base [7] and also operate in hazardous environments [8]. In [2], the authors propose and evaluate the use of ad hoc networking techniques and the corresponding algorithms for communication in mobile robot networks for both coordination and sensing oriented communication.

VII. CONCLUSION

In this paper, we studied multicast group communication in mobile robot networks. Multicast is useful for mobile robot team applications that involve coordination among team members, sensing, and data collection. Multicast also allows for flexible control and operation of the multi robot team by supporting organization of the mobile robots into many structures such as sub-teams, hierarchies, etc. Finally, multicast also allows for efficient discovery and usage of distributed resources available in the network due to heterogeneous capabilities and resources of each robot.

We proposed and evaluated the MRMM protocol for multicast which reduces the overhead and bandwidth usage and consequently reduces the energy for group communication in mobile robot networks. In our future work, we will look at how multicast can be enhanced by localization. We also plan to study the use of multicast for specific application scenarios in mobile robot networks.

REFERENCES

[1] Activmedia robotics. http://www.activrobots.com/.

[2] S. M. Das, Y. C. Hu, C. Lee, and Y.-H. Lu. Performance comparison of communication protocols for mobile robotic sensors. In *Proc. of ICRA*, 2004.

[3] L. Ji and S. Corson. Differential Destination Multicast–A MANET Multicast Routing Protocol for Small Groups . In *Proc. of IEEE INFOCOM*, April 2001.

[4] S.-J. Lee, M. Gerla, and C.-C. Chiang. On-Demand Multicast Routing Protocol. In *Proc. of IEEE WCNC*, September 1999.

[5] S.-J. Lee, W. Su, and M. Gerla. On-demand multicast routing protocol in multihop wireless mobile networks. *MONET*, 7(6), 2002.

[6] Y. Mei, Y.-H. Lu, C. Lee, and Y. C. Hu. Energy-efficient motion planning for mobile robots. In *Proc. of ICRA*, 2004.

[7] H. Nyugen, N. Farrington, and N. Pezeshkian. Maintaining communication link for tactical ground robots. In *Proc. of AUVSI Unmanned Systems North America*, August 2004.

[8] H. Nyugen, N. Pezeshkian, A. Gupta, and N. Farrington. Maintaining communication link for a robot operating in a hazardous environment. In *Proc. of ANS 10th Int. Conf. on Robotics and Remote Systems for Hazardous Environments*, March 2004.

[9] H. Nyugen, N. Pezeshkian, M. Raymond, A. Gupta, and J. Spector. Autonomous communication relays for tactical robots. In *Proc. of IEEE ICAR*, July 2003.

[10] E. M. Royer and C. E. Perkins. Multicast operation of the ad-hoc on-demand distance vector routing protocol. In *Proc. of MobiCom*, August 1999.

[11] P. Sinha, R. Sivakumar, and V. Bharghavan. MCEDAR: Multicast Core-Extraction Distributed Ad hoc Routing. In *Proc. of IEEE WCNC*, September 1999.

[12] X. Zeng, R. Bagrodia, and M. Gerla. Glomosim: A library for parallel simulation of large-scale wireless networks. In *Proc. of PADS Workshop*, May 1998.