An Optimization Approach for Cooperative Communication in Ad Hoc Networks

Carlos A.S. Oliveira and Panos M. Pardalos
University of Florida *

Abstract. Mobile ad hoc networks (MANETs) are a useful organizational technique for providing communication infrastructure to wireless devices. They consist of loosely coupled units, that communicate locally only to accessible neighbors. Routing of data among non accessible units in a MANET is made possible through multi-hop retransmission. There are many applications of MANETs, including coordination of rescue groups and other military applications such as UAVs (Unmanned Air Vehicles). We study the problem of determining an optimal route for a group of ad hoc users, such that the total connection time among nodes in the resulting MANET is maximized, subject to a limit on the maximum distance traveled by unit. This problem is called the cooperative communication problem in ad hoc networks (CCPMANET). A model for the problem using integer linear programming is developed. We give also a formulation based on graph theory. A heuristic local search algorithm is used to find solutions with good quality for the CCPMANET. Results of this algorithm, in comparison with standard techniques to solve integer programming models, are shown. The results suggest that the proposed IP model is effective in solving instances of moderate size.

Key Words. Optimization, cooperative control, integer programming, ad hoc networks, unit graphs.

1 Introduction

MANETs (Mobile Ad hoc networks) are a new paradigm for distributed communication that simplifies the deployment of wireless services. It has been used increasingly in situations where it is difficult or expensive to provide coverage using dedicated servers (the standard technique used in cellular systems, for example). An ad hoc network is based on wireless units that can serve as clients, as well as servers. The distribution/routing technique used depends on multi-hop protocols, in order to deliver messages between nodes connected to the network. We are concerned with the study of connectedness of a MANET from the point of view of a group of users that must cooperate to accomplish assigned tasks. With this goal in mind we propose a mathematical model and an algorithm to maximize the total connectivity in this situation.

1.1 Applications

Ad hoc networks have important applications, since they can be used to model real situations where mobile devices are not controlled by a central authority [13]. They are usually applied

*Center for Applied Optimization, Dept. of Industrial and Systems Engineering, University of Florida. 303 Weil Hall, Gainesville, FL. 32611. Emails: {oliveira,pardalos}@ufl.edu.

in very dynamic scenarios, where hosts change their positions with time, and communication is not guaranteed to reach a specific destination.

Applications of MANETs range from commercial wireless access on remote locations, to rescue forces or other operations on military battlefield [3]. These applications have in common the role of users in the dynamical state of the network.

Group communication and synchronization is required in many cases where group members have specific tasks to accomplish. In military situations, for example, a group of users may have different but related targets. Thus, communication between group elements is an important issue, and must be maintained for as long as possible.

1.2 Related Work

Much work has been done in the development of MANETs, with special attention to routing strategies for wireless devices on ad hoc networks [6, 10]. In terms of optimization, the majority of the work for MANETs has concentrated on minimizing the number of nodes required by a backbone subset, with the objective of maintaining connectivity to this backbone in one hop [2, 3, 7, 11]. This problem can be formulated as the classical connected dominated set problem, which is known to be NP-hard [9].

A mathematical model of user mobility in a MANET is important in order to make high level decisions based on actual usage patterns. Various mobility models have being proposed to account for changes in position of hosts in an ad hoc network [5]. We give here a special case of a mobility model described by discrete steps, and formalized as a planar graph. A previous discussion of a graph-based mobility model is given in [12].

1.3 Contributions

Although ad hoc networks have been recognized as important in cooperative control applications, there has been no proposals with the objective of maximize its usage in such situations. Thus, the study in this work can be viewed as a novel application of optimization to the integration of cooperative control and MANETs.

We present a mathematical formulation for the problem of maximizing the communication time for a group of clients with specified targets, given as destinations in a network. The constraints require that the total distance traveled do not exceed some specified quantity. This is necessary, for example, to model physical constraints on mission targets, such as total completion time, or maximum distance without refueling. The problem is modeled using integer mathematical programming and computational results from this effort are reported.

1.4 Organization

This paper is organized as follows. In the next section, a formal definition of the cooperative communications problem on ad hoc networks is given. Section 3, an integer programming model for the problem is introduced, with other useful properties of the CCPMANET. This model will be useful as a formalization technique and also to provide upper bounds on the optimal solution of some instances of the CCPMANET. The results of computational experiments with the algorithm are given in 4. Finally, concluding remarks are given on Section 5.

2 The Cooperative MANET Problem

We model the problem of maximizing connectivity for a set of objects traveling from a set of sources to a set of destinations. Initially we give some concepts needed from graph theory. Then, we discuss a set of simplifying assumptions that will be made about the problem. A formal definition for the problem is given at the end of the session.

2.1 Unit Graphs

A *graph* is a pair $G = (V, E)$, where $V = \{v_1, \ldots, v_n\}$ is a set of nodes and E is a set of pairs of nodes (v_i, v_j), with $v_i, v_j \in V$. The graphs used in this paper are planar, that is, the set of nodes represent points in the plane. Let $d : V \times V \rightarrow \mathbb{R}$ be a function returning the *distance* between a pair of adjacent nodes. Then, a *unit graph* is a graph which has $d(v_i, v_j) \leq 1$, for $v_i, v_j \in V$.

A unit graph is useful to model ad hoc networks. In this case, nodes represent wireless units, and edges represent possible links between nodes. Assuming that the radius of transmission has distance equal to one unit, then a unit graph is a representation of the set of links between wireless units in the network.

The standard notation for graphs is used in this paper. In particular, $N(v)$ is the set of nodes adjacent to v, and the degree $d(v)$ of node v is $d(v) = |N(v)|$, for $v \in V$.

2.2 Problem Assumptions

We discuss some simplifying assumptions that will be made about the cooperative communication problem. The importance of this assumptions is that they will result in a problem that is tractable using combinatorial optimization techniques.

The main simplifying assumption used in this paper is that the area covered by the set of objects is modeled as a planar graph $G = (V, E)$. Edges of the graph have the associated distance function $d : E \rightarrow \mathbb{R}_+$. If the graph represents a two dimensional surface, then clearly it will be a *planar graph*, and many useful properties of planarity, such as a bounds on connectedness, can be used. Another interesting characteristic of planar graphs that makes them useful in our problem is the *triangular inequality* property.

Definition 1. *A graph* $G = (V, E)$ *has the triangular inequality property if, for any* $x, y, z \in V$*, if* $(x, y), (y, z), (x, z) \in E$ *then* $d(x, y) + d(y, z) \geq d(x, z)$.

This condition is easy to ensure in our problem, since it follows from the fact that edge lengths represent the Euclidean distance between adjacent nodes. The next assumption is that in the graph G given as input, each pair of adjacent nodes represents the possible positions of an object after one unit of time passed. This means, in particular, that all objects travel at the same speed on a particular edge. The justification for this assumption is not too hard to find. Suppose for example, that the graph represents a road system, and that objects are cars traveling at the maximum allowed speed. In this case, clearly the distance covered in one unit of time is the same for all cars on that edge.

Additionally, we assume that the positions of objects in the system are discretized by units of time, each node in the graph representing a possible position. Also, it is assumed that at each of the nodes, there is just a limited number of possible directions that can be taken.

Going back again to the road system example, there is just a small number of directions available at each point. This is represented in the model graph by the *degree*, i.e., the number of neighbors at each node.

With these assumptions about the graph, the following data is provided for each instance of the problem. It is given a set of mobile units $U = \{u_1, \ldots, u_N\}$, together with an origin s_i and a destination d_i for each unit $u_i \in U$. This means that unit u_i must start its trajectory at node s_i of the graph and travel until node d_i is reached. The time in our system is discretized into time instants $t_0, \ldots, t_T$. If at a particular instant t_i the unit u_j is in node v_k, then in the next step u_i can be in any of the nodes adjacent to v_k or again in v. That is, if we defined $\sigma(u_i)$ to be the next position assumed by unit u_i, then $\sigma(u_i) \in N(u_i) \cup \{v\}$.

A *graph configuration* C is a graph where the positions (nodes) assumed by each of the objects $u_1, \ldots, u_n$ are marked. If the total time is given by the instants $t_0, \ldots, t_T$, then the graph configurations represent the sequence of positions assumed by units $u \in U$ during the interval of time $[0, T]$. A *feasible sequence of graph configurations* $C_1, \ldots, C_c$, or for short *feasible sequence*, is one in which object u_i is in position j of configuration k if and only if object u_i is in position j' of configuration $k - 1$ and $j \in \sigma(j')$, for $1 < k \leq c$.

2.3 Measures of Connectivity

Before continuing with the description of the problem, we need to to make clear how to measure connectivity in the graphs resulting from the trajectories of the considered objects. To do this, we define some metrics of connectivity of communication in a graph. Clearly, the best situation in terms of connectivity happens when all nodes are linked. However, in a network of units with limited power this situation will not be true in most of the cases. In the other extreme, when all n components are unable to communicate, we have the worst possible scenario. In between, a series of alternatives exist for determining the connectivity among units in the network.

Initially, a good indication of how connected is a network is given by the number of components generated by the unit graph. However, this is not a completely objective measure, since different situations can generate the same number of components. For example, given a network with n nodes, it is possible to have a decomposition in two components, where the first component has $n - 1$ nodes and the second only one node. Another possibility has components with sizes $\lfloor n/2 \rfloor$ and $\lceil n/2 \rceil$.

A more objective measure of connectivity is given as follows. Suppose that each pair of nodes is consider separately when computing the total connectivity of a graph configuration. Thus, the connectivity of a configuration C would be given by

$$\gamma = \sum_{i=1}^{N} \sum_{j=1}^{N} y_{ij},$$

where y_{ij} is 1 if and only if i is connected to j in the configuration graph. This γ is a measure of the connectivity of the configuration. The definition can still be further generalized by letting $w : V \times V \to R$ be a function such that $w(a, b)$ gives the weight (or importance) of the connectivity between nodes a and b, $a, b \in V$. Thus, we have a general definition of connectivity among the objects $u_1, \ldots, u_n$, given by

$$\gamma_w = \sum_{i=1}^{N} \sum_{j=1}^{N} w(v_i, v_j) y_{ij}. \tag{1}$$

This is the definition that will be used in the remaining of the work.

Then, the objective of the CCPMANET problem is to maximize connectivity, as defined in equation (1), over all feasible sequences of configurations, for time instants $t_0, \ldots, t_T$. Also, we require that the total distance traveled by unit i be at most D_i, due to physical constraints, such as, for example, total amount of fuel available. (Of course, we must have $D_i \geq \delta(s_i, d_i)$, where $\delta(s_i, d_i)$ represents the cost of the shortest path from s_i to d_i.)

3 A Mathematical Programming Model

The problem as stated in the previous section can be defined using mathematical programming techniques. This will be useful as a way of formalizing the problem, and also can be used to give results using standard IP software.

Let x_{ijt} be a binary variable, defined as

$$x_{ijt} = \begin{cases} 1 & \text{if object } i \text{ is in node } j \text{ at time } t \\ 0 & \text{otherwise,} \end{cases}$$

for $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots, N\}$, and $t \in \{1 \ldots, T\}$. Also, consider the variable y_{ijt} defined as

$$y_{ijt} = \begin{cases} 1 & \text{if nodes } i \text{ and } j \text{ are connected at time } t \\ 0 & \text{otherwise,} \end{cases}$$

for $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots, N\}$, and $t \in \{1 \ldots, T\}$.

The connectivity of graph G at instant of time t is given by

$$\gamma(t) = \sum_{i=1}^{N} \sum_{j=1}^{N} w(i, j) y_{ijt},$$

where w is a weight function on the edges, as explained in the previous section. Thus, our objective function is

$$\max \sum_{t=1}^{T} \gamma(t)$$

subject to constraints that can be described as follows. The first set of constraints define the binary value of the variable y_{ijt}. This value must be equal to 1 if and only if both nodes v_i and v_j hold objects. The can be described with the following constraint:

$$y_{ijt} \leq (x_{iut} + x_{ju't})/2, \tag{2}$$

for all $i, j \in \{1, \ldots, n\}$, $u, u' \in \{1, \ldots, N\}$, $t \in \{1, \ldots, T\}$. The next restrictions account for the simple rules of movement for objects in the system: they must travel to an adjacent node or continue in the same. This is represented by a disjunctive constraint, which means that, for each node i, at least one of the adjacent nodes in the previous graph configuration must be selected.

$$x_{ijt} \leq x_{i',j,t-1} + (1 - z_{i',j,t}), \tag{3}$$

for all $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots, N\}$, $t \in \{2, \ldots, T\}$, and i' such that $v_{i'} \in N(v_i) \cup \{v_i\}$. To make at least one of the above conditions true, we must have the following constraint:

$$\sum z_{i,j,t} \geq 1, \tag{4}$$

for $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots, N\}$, and $t \in \{1, \ldots, T\}$. The following constraints give initial and final conditions that must be true for all units in the group

$$x_{ij1} = 1, \qquad \text{for } j \in \{1, \ldots, N\} \text{ and } i = s_j \tag{5}$$

$$x_{ijT} = 1, \qquad \text{for } j \in \{1, \ldots, N\} \text{ and } i = d_j. \tag{6}$$

These constraints enforce each unit j to start in node s_j in the first configuration C_1, and finish in node d_j in the last configuration C_T.

The next constraints are related to the limit on distance traveled by objects in the system. They can be developed using well known flow inequalities [1]. These constraints use a variable r_{abj} that defines a multicomodity flow in the network for each unit j and among each pair of nodes $a, b \in V$.

$$\sum_{(a,b) \in V \times V} d(a,b) r_{abj} \leq D_j \qquad \text{for } j \in \{1, .., N\} \tag{7}$$

and

$$N r_j = b_j \qquad \text{for } j \in \{1, \ldots, N\}, \tag{8}$$

where N is the node-arc incidence matrix, b_j is a vector such that $b_j(s_j) = 1$, $b_j(d_j) = -1$, $b_j(k) = 0$, for $k \neq s_j$, $k \neq d_j$, and r_j is a vector of variables r_{abj}, for a fixed value j.

Finally, the remaining constraints determine the type of values allowed for the variables in the problem

$$x_{i,j,t} \in \{0,1\}, \quad y_{i,i',t} \in \{0,1\}, \quad z_{i,j,t} \in \{0,1\}, \quad r_{i,i',j} \in \mathbb{R}_+, \tag{9}$$

for all $i, i' \in \{1, \ldots, n\}$, $j \in \{1, \ldots, N\}$, and $t \in \{1, \ldots, T\}$.

4 Computational Results

Computational experiments were performed with the proposed IP model to determine its effectiveness. In our experiments, a set of instances were generated by using parameters based on real values. For example, the instances tested bellow were generated for areas ranging from 10 to 20 miles. The generator is based on previous work done for the minimum size backbone problem [3, 4].

The software used for IP modeling was the Mosel XPress package, which has an integrated modeler and LP solver. Integer programs can be solved by calling an embedded branch-and-bound routine, which also applies standard valid inequalities, such as Gomory cuts (see [8]). The experiments were run on a PC equipped with a 2.8GHz processor and 504MB of main memory.

instance	num. of nodes	IP opt	time (s)	LS opt.	time (s)
1	20	517	12	517	1
2	30	721	71	721	4
3	40	769	59	758	5
4	60	811	129	806	12
5	70	876	178	875	16
6	80	974	291	969	19
7	90	992	482	990	23
8	100	1184.7	7200	1153	34
9	110	1523.4	7200	1432	39
10	120	1589.6	7200	1496	46

Table 1: Results of computational experiments.

The IP model was generated for instances with 20 to 120 nodes and each group with 5 units. The sources and destinations were chosen randomly from available nodes. The integer programming model was solved, yielding the results summarized in Table 1. The last three instances were stopped after 2 hours of computation, but the integrality gap reported was less than 5%.

To make possible a comparison, we created a simple local search algorithm for the problem. It starts from an initial solution which is equal to the union of shortest paths between sources and destinations. This solution is improved using a 2-opt method for increasing connectivity: if two units cannot communicate, we try to move then to a closer position, provided that the resulting configuration remains feasible. The algorithm was coded in C language, and compiled with the gcc compiler. It was executed in the same instances, and the resulting values reported in the last columns of Table 1.

From the results in Table 1 we see that the integer model, processed through standard methods such as branch-and-bound, is useful in solving instances of moderate size. We could not, however, achieve optimality for some larger instances. This suggests that other techniques must be applied to improve the performance in solving the CCPMANET. One possible method that we intend to explore in the future is to generate cuts based on polyhedral properties of the problem.

5 Concluding Remarks

In this paper, a new problem related to mobile ad hoc networks on cooperative environments has been discussed. We defined the cooperative communications problem in ad hoc network (CCPMANET). The CCPMANET has been modeled in the context of discrete network optimization, and the assumptions made allowed the problem to be more easily tractable with the well known tools of combinatorial optimization. Based on this model, an IP formulation for the problem was proposed, as well as a heuristic algorithm.

Many questions about cooperative communication in MANETs remain open. For example, it may be interesting to consider that the problem is in fact continuous, and therefore to propose continuous analytical and numerical methods to solve it.

Another of the features of mobile ad hoc networks that are not addressed here is the non symmetry that characterizes most existing networks. Asymmetry can occur as a result of physical limitations as, for example, different equipment used, interference, etc., or as a deliberate result of power control decisions. It would be interesting to develop models that consider these characteristics using asymmetric links.

Another important extension of the proposed model occurs when the group of objects considered are part of different sub-unities. The interesting feature here is that some of these sub-unities can have a non-empty intersection. This makes more complicated to define the maximum connectivity in a suitable way for the whole network.

References

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, 1993.

[2] K. M. Alzoubi, P.-J. Wan, and O. Frieder. Distributed heuristics for connected dominating set in wireless ad hoc networks. *IEEE ComSoc/KICS Journal on Communication Networks*, 4(1):22–29, 2002.

[3] S. Butenko, X. Cheng, D.-Z. Du, and P. M. Pardalos. On the construction of virtual backbone for ad hoc wireless network. In S. Butenko, R. Murphey, and P. M. Pardalos, editors, *Cooperative Control: Models, Applications and Algorithms*, pages 43–54. Kluwer Academic Publishers, 2002.

[4] Sergiy Butenko, Xiuzhen Cheng, Carlos A.S. Oliveira, and Panos M. Pardalos. A new algorithm for connected dominating sets on ad hoc networks. In S. Butenko, R. Murphey, and P. Pardalos, editors, *Recent Developments in Cooperative Control and Optimization*, pages 61–73. Kluwer Academic Publishers, 2003.

[5] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, 2(5):483–502, 2002.

[6] Ching-Chuan Chiang and Mario Gerla. Routing and multicast in multihop, mobile wireless networks. In *Proceedings of ICUPC '97*, 1997.

[7] B. Das and V. Bharghavan. Routing in ad-hoc networks using minimum connected dominating sets. In *International Conference on Communications*, pages 376–380, 1997.

[8] Dash Optimization Inc. *Xpress-Optimizer Reference Manual*, 2003.

[9] M. R. Garey and D. S. Johnson. *Computers and Intractability - A Guide to the Theory of NP-Completeness*. W. H. Freeman, San Francisco CA, 1979.

[10] David B. Johnson. Routing in ad hoc networks of mobile hosts. In *Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications*, 1994.

[11] I. Stojmenovic, M. Seddigh, and J. Zunic. Dominating sets and neighbor elimination based broadcasting algorithms in wireless networks. In *Proc. IEEE Hawaii Int. Conf. on System Sciences*, 2001.

[12] J. Tian, J. Hhner, C. Becker, I. Stepanov, and K. Rothermel. Graph-based mobility model for mobile ad hoc network simulation. In *Proceedings of 35th Annual Simulation Symposium*, San Diego, California, 2002.

[13] P.-J. Wan, K. Alzoubi, and O. Frieder. Distributed construction of connected dominating set in wireless ad hoc networks. In *IEEE INFOCOM'02*, 2002.