Resource Discovery in Disconnected Mobile Ad-hoc Networks

(Extended Abstract)

O. Wolfson, A. Ouksel and Bo. Xu

The University of Illinois at Chicago, e:mail {wolfson@cs.uic.edu, aris@uic.edu}

Consider an urban area with hundreds of thousands of vehicles. Drivers and passengers in these vehicles are interested in information relevant to their trip. For example, a driver would like his/her vehicle to continuously display on a map, at any time, the available parking spaces around the current location of the vehicle. Or, the driver may be interested in the traffic condition one mile ahead. Such information is important for drivers to optimize their travel, to alleviate traffic congestion, or to avoid wasteful driving. The challenge is processing queries in this highly mobile environment, with an acceptable delay, overhead and accuracy. One approach to solving this problem is maintaining a distributed database stored at fixed sites that is updated and queried by the moving vehicles via the infrastructure wireless networks. Potential drawbacks of this approach are (i) the responses to queries may be outdated, (ii) the response time may not meet the real-time requirements, and (iii) access to infrastructure communication service is costly. In this proposal we explore a new paradigm that is based on inter-vehicle communications.

We assume that each vehicle has the capability of communicating with its neighbors. This communication can be enabled by a local area wireless protocol such as IEEE 802.11, Ultra Wide Band (UWB), Bluetooth, or CALM. These protocols provide broadband (typically tens of Mbps) but short-range (typically 50-100 meters) peer-to-peer communication. With inter-vehicle communication, a mobile user discovers the desired information from its neighbor vehicles, or from remote vehicles by multi-hop transmission relayed by intermediate moving vehicles. Thus, resource discovery is performed in an *inter-vehicle ad hoc network*. Compared to the traditional fixed-station based information query, this paradigm has the following advantages:

1. It provides better information authenticity, accuracy, and reliability, especially for real-time information. Consider for example parking space availability. Information collected from a vehicle that is leaving a parking lot tends to be more reliable than that from the fixed site.
2. It is free of charge, assuming that vehicles are willing to relay messages for free (in exchange for their messages being relayed).

Here, we propose a message routing technique that tries to route a message from one host to another even if the source and destination are never connected either directly or transitively. The basic idea is, when a moving object *O* receives a message for which it is not the destination and there are no other objects (except the one from which *O* receives the message) within the transmission range of *O*, *O* carries the message. When *O* encounters an object that is expected to be able to move the message toward the destination, it forwards the message to that object. Since the routing is aided by motion of objects, we call this paradigm *mobility-aided routing*, or *MAR*.

In MAR, when *O* decides whether to forward a message to an object it encounters, there are three factors coming into play. First, the bandwidth consumption; second, the reachability of the destination (in an ad-hoc network, a destination is not necessarily reachable); and third, the delay of delivery. While flooding the message to every object encountered maximizes reachability and minimizes delay, it increases bandwidth consumption. The decision of whether to forward a message to a neighbor object relies on how each of the above factors is weighted. And it also relies on the knowledge possessed by an object regarding the future motion of other objects. We distinguish between the following two cases, in terms of what a moving object knows about the motion plans of others:

1. An object knows the motion plans of every other object. For example, in a fleet management scenario, each object in the fleet is aware of the motion plan of every other one in the same fleet. And objects in the fleet only communicate via objects inside the fleet.
2. An object does not know the motion plans of other object.

We have devised resource discovery algorithms for both cases, but this paper will be limited to the first one. In this abstract , we formally set up the problem, discuss one polynomial time optimal message routing algorithm and then indicate the additional issues that will be addressed in the final paper.

Moving objects use short-range (typically 50-100 meters) wireless technologies for inter-object communication. We denote by *r* the transmission range of inter-object communication. We assume that an object is capable of detecting the objects that are within its transmission range. More over, if multiple objects enter into the transmission range simultaneously, they are detected sequentially. We say that two objects *encounter* each other when their distance is smaller than *r*. Communication between objects in the ad hoc network is free of charge.

A *resource* is a piece of information about an event (e.g. the availability of a parking space), or the status of a set of decision variables (e.g. traffic speed of a highway segment during a particular time period) specific to a certain location or area. The *home* of a resource is the location of the event or the area corresponding to the status report. For example, the home of an available parking space is the location of the parking space, and the home of a resource about the traffic status is a highway segment or city blocks.

For each resource, there is a pre-determined area where the resource is available for access. This area is referred to as the *distribution area* of the resource. Various ways may be used to ensure the resource available in the area. For example, a head object may be designated to collect resources in the area and provide answers to queries about them. Another possibility is that two objects exchange the resources available to them when encountering other objects in the area. Thus a resource may be replicated over several objects in the area.

We assume that resources are represented by some resource description language (e.g. RDF or DAML+OIL), and a resource query is represented by a resource query language (e.g. RDF Query Specification).

There are basically two steps in the processing of a resource query. In the first step, the query is disseminated to the distribution area of the queried resource. If any object in the distribution area receives the query and it has the queried resource, then in the second step the answer is returned to the requesting object. We refer to the first step as the *query dissemination* step and the second the *answer delivery* step. In this section we introduce the structures of a query message and an answer message; and then discuss the procedure of resource discovery. A *query number* is associated with each request. A possible scheme of the query number is as follows. The query number consists of two components. The first component is a unique identification of the requesting object. The second component distinguishes the query from the other queries issued from the same object. For example, it can be a number that is increased by one for each query issued from the requesting object. Now we describe the structures of a query message and an answer message respectively. A query message contains the following fields: query, trajectory of the requester, destination area, expire time, Hop To Live (HTL),Query number. An answer message contains the following fields: Answer, destination trajectory, HTL and query number (all three last fields are copied from the corresponding query message). At any point in time, each object keeps a queue that contains the latest *h* different query numbers included in the messages that it has already previously received and corresponding to the specific queries it has already answered. This queue is called the *query number queue*.

Procedure Description

The procedure of resource discovery starts by the requesting object constructing a query message as described above. Then the query dissemination step follows up, in which the query message is delivered to the destination area, possibly in multi-hops. At any hop, if the expire time is reached or HTL is zero, the query message is discarded. When an object in the destination area receives the query message, it first checks whether the query number of the query message appears in the query number queue. If so, the query message is simply discarded. Otherwise, the object evaluates the query. If it has the answer, an answer message is constructed, the query number is added to the query number queue, and the answer delivery step starts. In the answer delivery step, the answer message is delivered to the requester, possibly in multi-hops. At any hop, if the expire time is reached or HTL is zero, the query message is discarded.

The most important issue in the above resource discovery procedure is how to route a message from a source to a destination. The sketch of our solution to this issue is as follows. When an object *O* receives a message for which *O* is not the destination and there are no other objects (except the one from which *O* receives the message) within the transmission range of *O*, *O* carries the message. When *O* encounters an object that is expected to be able to move the message toward the destination, it forwards the message to that object. This procedure is called *mobility aided routing*, or *MAR*. We distinguish between two cases, in terms of what an object knows about the motion plans of other objects. In section 4.1 we discuss MAR in the case where each object knows the trajectories of each other object. In subsection 4.2 we discuss MAR in the case where an object does not know the motion plans of others.

Suppose that each object knows the trajectories of each other object, namely each object knows how the other objects will move. Given the length of a message, the source and the destination of the message, how do we know if the message can reach the destination before the expire time via the MAR procedure? In order to attack this issue, let us first discuss a more fundamental issue, namely how do we know if a message can reach the destination.

First let us introduce several notions that will be used to decide reachability.

Definition 1: Let $d(p,q)$ denote the distance between location p and location q. A *contact interval* between two objects i and j, denoted $L_{ij}(s,e)$, is a time interval $[s,e]$ such that

1. for any time point $t \in [s,e]$, $d(Tr_i(t),Tr_j(t)) \leq r$. Recall that $Tr_i(t)$ and $Tr_j(t)$ are the expected locations of i and j at t interpolated by Tr_i and Tr_j respectively;
2. if there exists a time point $t<s$ (or $t>e$) such that $d(Tr_i(t),Tr_j(t)) \leq r$, then there exists a point t' such that $t<t'<s$ (or $e<t'<t$) and $d(Tr_i(t'),Tr_j(t'))>r$. □

Intuitively, a contact interval is a continuous time interval where the objects represented by the two trajectories are within the transmission range of each other.

Definition 2: The *contact graph* is an undirected graph, where each node i represents a moving object i, and each link $L_{ij}(s,e)$ represents a contact interval between i and j. □

Note that there can be multiple links between i and j, each corresponding to one contact interval.

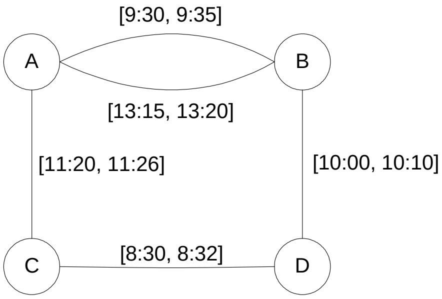

Figure 1: An example contact graph

Example 1: Figure 1 gives an example contact graph of four objects A, B, C, and D. There are two contact intervals between A and B, indicating that they encounter each other from 9:30 to 9:35 and from 13:15 to 13:20 respectively. Similarly, the contact graph gives the contact intervals between A and C, C and D, and B and D. Let us assume that the one-hop transmission time is less than one minute. From Figure 1 we can see that D is reachable from A at any time before 9:30. A possible routing scheme is that A forwards the message to B when A encounters B at 9:30, and then B forwards to D when B encounters D at 10:00. However, D is not reachable from A at any time after 9:35, via either B or C. D is not reachable via B because after 9:35 A encounters B before B encounters D. Similarly D is not reachable via C because A encounters C after C encounters B. □

1. **Shortest Path Algorithm, here. We assume at most single link between two objects in a contact graph. In this algorithm we assume that the cost in time of message passing between vehicles is negligible.**
2. **Modification of the above algorithm to include the case where the transmission of a message incurs a cost. This will have a consequence both on the feasibility as some previously feasible paths will no longer be, and the probability of delivering an answer to requester within the deadline.**
3. **General algorithm, including possibility of multiple contacts between vehicles, to find a feasible schedule with deadlines. We call this problem, sequencing to minimize completion time with precedence constraints.**