Opportunistic Data Dissemination in Mobile Peer-to-Peer Networks*

A. Prasad Sistla[1], Ouri Wolfson[1], Bo Xu[1]

[1] Department of Computer Science, University of Illinois at Chicago
University of Illinois at Chicago, USA
{sistla, wolfson, boxu}@cs.uic.edu

Abstract. In this paper we examine the dissemination of availability reports about resources in mobile peer-to-peer networks, where moving objects communicate with each other via short-range wireless transmission. Each disseminated report represents an observed spatial-temporal event, and the relevance of the report to a moving object decays as the age of the reported resource and the distance from its location increase. We propose an opportunistic approach, in which an object propagates the reports it carries (namely the information that it has about these resources) to encountered objects and obtains new reports in exchange. Least relevant reports are discarded after each exchange so as to limit the communication data volume of future exchanges. Our theoretical and experimental analysis indicates that the opportunistic dissemination algorithm automatically limits the global distribution of a report to a bounded spatial area and to the duration for which it is of interest. We propose two variants of the opportunistic dissemination algorithm and compare them with the traditional client-server architecture in terms of data accuracy. The proposed system has the potential to create a completely new information marketplace.

1 Introduction

A mobile peer-to-peer network is a set of moving objects that communicate via short-range wireless technologies such as IEEE 802.11 [1], Bluetooth [2], or Ultra Wide Band (UWB) [3]. With such communication mechanisms, a moving object receives information from its neighbors, or from remote objects by multi-hop transmission relayed by intermediate moving objects. A killer application of mobile peer-to-peer networks is resource discovery in transportation. For example, the mobile peer-to-peer approach can be used to disseminate the information of available parking slots, which enables a vehicle to continuously display on a map to the driver, at any time, the available parking spaces around the current location of the vehicle. Or, the driver may use this approach to get the traffic conditions (e.g. average speed) one mile ahead. Similarly, a cab driver may use this approach to find a cab customer, or vice versa.

A mobile peer-to-peer network can also be used in matching resource producers and consumers among pedestrians. For example, an individual wishing to sell a pair

* Research supported by NSF Grants 0326284, 0330342, ITR-0086144, and 0209190.

of tickets for an event (e.g. ball game, concert), may use this approach right before the event, at the event site, to propagate the resource information. For another example, a passenger who arrives at an airport may use this approach to find another passenger for cab-sharing from the airport to downtown, so as to split the cost of the cab. Furthermore, the approach can be used in social networks; when two singles whose profiles match are in close geographic proximity, then one can call the other's cell phone and suggest a short face-to-face meeting.

The approach can also be used for emergency response and disaster recovery, in order to match specific needs with expertise (e.g. burn victim and dermatologist) or to locate victims. For example, scientists are developing cockroach-sized robots or sensors that are carried by real cockroaches, which are able to search victims in exploded or earthquake-damaged buildings [4]. These robots or sensors are equipped with radio transmitters. When a robot discovers a victim, it can use the data dissemination among mobile sensors to propagate the information to human rescuers. Sensors can also be installed on wild animals for endangered species animal assistance. A sensor monitors its carrier's health condition, and it disseminates a report when an emergency symptom is detected. Thus we use the term moving objects to refer to all, vehicles, pedestrians, robots, and animals.

In this paper we propose to examine an *opportunistic* approach to dissemination of reports regarding availability of resources (parking slot, taxi-cab customer, dermatologist, etc.). In this approach, a moving object propagates the reports it carries to encountered objects, and obtains new reports in exchange. For example, a vehicle finds out about available parking spaces from other vehicles. These spaces may either have been vacated by these encountered vehicles or these vehicles have obtained this information from other previously encountered ones. Thus the parking space information transitively spreads out across vehicles. Similarly, information about an accident or a taxi cab customer is propagated transitively. In this paper we explore this information propagation paradigm, which we call *opportunistic peer-to-peer* (or OP2P).

With OP2P, a moving object constantly receives availability reports from the peers it encounters. If not controlled, the number of reports saved and communicated by a peer may continuously increase. In order to limit the data exchange volume, we employ a relevance function that prioritizes the availability reports. The relevance of a report to a moving object o is clearly spatio-temporal, namely the relevance decreases the older the report gets, and the farther the reported resource is from m. In this paper, we introduce a simple spatio-temporal relevance function, and assume that each moving object saves only the M most relevant reports. We call this method *Opportunistic Report Dissemination* (ORD). In this paper we study ORD from three aspects.

First, we examine the pattern of report propagation with ORD. Mathematical modeling and analysis of the report-distribution is intractable in general. However, it can be solved for the case in which the relevance is purely temporal, i.e., no spatial component. This is the case for hotspots that broadcast to neighboring moving objects, for example, the current stock-market average. Thus we first devise differential equations that model object distribution in case relevance of a report to a moving object is temporal. We show that a report R generated at a certain time disappears from the system after a limited period of time, t. Using the maximum speed of a moving object, t can be easily translated to a limited geographic area where the information about R

spreads. This area is a circle around the point in space where the report was generated, namely its home location.

Then, using simulations we analyze the more general case, where relevance to a moving object is spatio-temporal. We show that again, a report only spreads within a limited geographic area around its home location (e.g. the location of a parking space, or the location of a cab customer). Second, within this limited area, the replication-density of a report varies with time, in a way which will be explained. Finally, the report starts disappearing from the system, until a time threshold beyond which there is no copy of the report in the system.

Then we compare ORD with the client/server model. In the client/server model, a sensor senses the availability of the resource, and sends a report to a central database when the resource becomes available. The moving objects access the server through a cellular network. There are several drawbacks of the client/server model. First, it is difficult for the model to scale to a large number of moving objects. One possible solution to increase the scalability is to divide a geographic area into service regions (similar to cells in a cellular infrastructure). There is a server in each service region that handles resources and moving objects within that region. However, this solution introduces the complexity of hand-over, which occurs when a moving object crosses the border between two service regions. Second, the client/server model is vulnerable to the failure of the central server. Finally, in the client/server mode, a moving object user has to pay for the cellular communication and the information service. In the peer-to-peer model a user only needs to pay for the initial installation of the communication module. The operation of the communication module is virtually free. A back of the envelope calculation reveals that the cost (in terms of fuel) of communicating with encountered vehicles is less than a cent per day, even if the communication is continuous throughout the day.

With all the above extra cost, what does client/server buy us? We compare the quality of data received with ORD and client/server. Observe that at any time instance an availability report in the local database of a moving object o may be incorrect in the sense that a resource that shows as available, actually becomes unavailable before o reaches it. So we propose a set of data quality measures that mimic the precision/recall measures used in information retrieval. Our simulations show that with very reasonable object density and wireless transmission range, the quality of the ORD method reaches that of client/server. This indicates that OP2P could serve well as an alternative to the client/server model but with much less operational cost. We also study the performance of ORD when there are failures in peer-to-peer interactions. In an OP2P environment, interaction failures may be caused by packet loss, communication module sleeping for power reservation, or the limited connection time between highly mobile objects.

Finally, we study a variant of ORD, in which invalidation messages are propagated when a resource becomes unavailable. This algorithm is called Opportunistic Report Dissemination with Invalidation (ORDI). We compare ORDI with ORD and client/server with invalidation. It would appear that clearly the invalidation algorithm will have a higher data quality, but this is misleading since we compare the algorithms on the same size of local database; therefore, the invalidation messages may occupy space of other correct reports. However, our experimental analysis shows that ORDI is

indeed superior to ORD.

In summary, this paper makes the following contributions. First, we devise a mathematical model for dissemination of information about purely temporal resources and experimentally analyze the dissemination of information about spatio-temporal resources. Then we propose an algorithm for opportunistic dissemination of information about spatio-temporal resources, and we compare it with the tradition client/server model. Finally we analyze the performance of the algorithm when using invalidation messages.

Although some concepts employed in this paper including gossiping and invalidation have been analyzed in the past, this paper applies these concepts to a mobile peer-to-peer environment. Furthermore, it combines these concepts with a novel aspect, which is filtering and ranking of spatio-temporal information based on a relevance function.

The rest of the paper is organized as follows. Section 2 introduces the model and the ORD algorithm. Section 3 analyzes the report propagation pattern. Section 4 compares ORD with the client/server model. Section 5 describes ORDI and compares it with ORD and the client/server model. Section 6 discusses relevant work and section 7 concludes the paper.

2 The Model

2.1 Resource Model

In our system, resources may be spatial, temporal, or spatio-temporal. Information about the location of a gas station is a spatial resource. Information about the price of a stock on 11/12/03 at 2pm is temporal. There are various types of spatio-temporal resources, including parking slots, car accidents (reports about such resources provide traffic-jam information), taxi-cab requests, ride-sharing invitations, demands of expertise in disaster situations, and so on. These resources are spatial in the sense that they are tied to a location, and are temporal in the sense that they are valid or available only for a limited time-duration.

Formally, in our model there are N *resource types* T_1, T_2, ..., T_N . At any point in time there are M *resources* R_1, R_2, ..., R_M, where each resource belongs to a resource type. We assume that resources are located at points in two-dimensional geospace. The location of the resource is referred to as the *home* of the resource. This is the spatial aspect of resources. For example, the home of an available parking space is the location of the space, and the home of a cab request or a cab-sharing invitation is the location of the customer. The state of each resource alternates between *valid* (i.e. available) and *invalid*. The period of time during which the resource is valid is called the *valid duration*. This is the temporal aspect of resources. For example, the valid duration of the cab request resource is the time period since the request is issued, until the request is satisfied or canceled.

2.2 Peers and Validity Reports

The system consists of two types of peers, namely fixed hotspots and moving objects. Each peer *o* that senses the validity of resources produces *validity reports*. Denote by *a*(*R*) a report for a resource *R*. For each resource *R* there is a single peer *o* that produces validity reports, called the *report producer* for *R*. A peer may be the report producer for multiple resources. Each report *a*(*R*) contains four attributes, namely *resource-type*, *resource-id*, *timestamp*, and *home-location*. Attribute resource-type indicates the resource type of *R*. Resource-id is the identification of *R* that is unique among all the resources in the system. In our model time is a sequence of discrete atomic time units, 1, 2, 3, …, and timestamp is a natural number indicating the time at which *a*(*R*) is transmitted to a peer by its producer.

For each resource type *T*, a peer *o* has a *validity reports database*, or *reports database*. Denote by $DB_o(T)$ the reports database of *o* for the resource type *T*.

2.3 Relevance Model

In order to limit the data exchange volume, for each resource type *T*, a moving object keeps in the reports database the top *M* relevant reports of type *T* that the object knows at that time. *M* is referred to as the *interest threshold*. For example, a user who is looking for a resource type *T* has the reports database that keeps the top 10 relevant validity reports of *T*. In other words, the user wants only the 10 most relevant reports to be saved and displayed. In this paper we use the following relevance function:

$$\mathrm{Rel}(a(R)) = e^{-(\alpha \cdot t + \beta \cdot d)} \quad (\alpha, \beta \geq 0) \tag{1}$$

where *t* is the age of *a*(*R*), namely the number of time units since *a*(*R*) is transmitted by its producer, and *d* is the travel distance from the home-location of *R* to the moving object. α and β are constants that represent the decay of relevance with respect to time and distance respectively. α and β may vary per resource type. Observe that this function is always positive, indicating that each report always has some relevance, and it decreases as *t* and *d* increase. We assume that each moving object is equipped with a GPS system so that (i) the object knows its location at any point in time and (ii) the clock is synchronized among all the objects. Thus both the age *t* and the distance *d* can be computed by the moving object.

Let us consider the resources that require a moving object to physically reach them ahead of other objects in order to occupy or possess them (e.g. parking slots, cab requests, or highway assistance requests). In our prior work ([22]) we have shown that for such a resource *R*, under some conditions the relevance of a report *a*(*R*) equals to the probability that *R* is valid when the moving object reaches it.

Theorem 1: Assume that the length of the valid duration (see subsection 2.1) of *R* is a random variable with an exponential distribution having mean *u*. Let the speed of the moving object be *v*. If $\alpha = 1/u$ and $\beta = 1/(u \cdot v)$, then the relevance of a report *a*(*R*) is the probability (at report acquisition time) that the resource *R* is valid when the moving object reaches *R*.

For the proof of Theorem 1 a reader is referred to [22]. The theorem motivates our definition of the relevance function (at least for resources with exponentially distributed valid-duration).

2.4 The Opportunistic Report Dissemination (ORD) Algorithm

We assume that each peer is capable of communicating with the neighboring peers within a maximum of a few hundred meters. One example is an 802.11 hotspot or a PDA with Bluetooth support. The underlying communication module provides a mechanism to resolve interference and collisions. Each peer is also capable of discovering peers that enter into or leave out of its transmission range (see e.g. [13]).

Recall that each moving object keeps a database of size M for each resource type. When two moving objects A and B encounter each other (i.e. they come within transmission range)[1], for each resource type T, A and B exchange their local databases, i.e. $DB_A(T)$ and $DB_B(T)$, and each one keeps the M most relevant reports. When A encounters a hotspot C, C transmits to A the reports it produces. Again A keeps the M most relevance reports.

In the rest of this paper we will assume that there is a single resource type in the system. However, most of our results also apply when there are multiple resource types, and we will specify when they do not.

3 Pattern of Report Propagation

In subsection 3.1 we study the propagation pattern of reports for temporal resources and in 3.2 we study the propagation pattern of reports for spatio-temporal resources.

3.1 Propagation Pattern of Reports for Temporal Resources

In this subsection we theoretically analyze how a report is propagated per time and per distance with the ORD algorithm. We consider a special case for the relevance function, where the decay factor of distance β is zero. Thus the relevance is purely a function of the age. This relevance function models the decay of the reports that are only specific to time, e.g. the Dow Jones Industrial Average at a particular time. In this section we first introduce the parameters and our assumptions, and then we develop a mathematical model that describes the propagation of a report. Finally we use the mathematical model to analyze the propagation of reports.

[1] If A uses broadcast, then the broadcast is used to select a peer to interact with. After the peer is selected, the interaction follows the one-to-one exchange procedure. Extension of report exchange to take the advantage of broadcast is a subject of our future work.

3.1.1 Parameters and Assumptions

Let N be the total number of peers in the system. We assume that the value of M is 1 for all the peers although our result can be extended to the general case where M is more than 1. Each peer interacts with other peers by a Poisson process with intensity λ. We assume that the length of the valid duration of a resource is 0, and the wireless transmission range is small enough such that at most one peer can receive the report when it is produced. Observe that with this assumption, the age of a report is always 0 when it is acquired. Further observe that the report may still be relevant even after it is invalid. For example, although a Dow Jones Industrial Average report is invalid right after it is generated, it is still of interest for a period of time. For another example, even after occupied, a parking slot report is relevant because peers do not always know whether it is occupied. We consider only the reports that are received by a peer when they are produced. Such reports are generated within the system by a Poisson process with intensity μ. Peers are randomly distributed in the space at any point in time, and therefore each peer is equally probable to receive each produced report. A newly generated report is sent to exactly one peer. Thus each peer receives newly generated reports according a Poisson process with the rate μ/N. Finally we assume that each report exchange is finished instantaneously.

3.1.2 A Mathematical Model for Report Propagation

Let us define two variables: $\boldsymbol{q(t)}$**:** The conditional probability that a peer has a report for a resource R at time t (t>0) given that R is created at time 0. $\boldsymbol{g(t)}$**:** The probability that at time t (t>0) a report that is created after 0 is in the reports database of a peer. Now consider $q(t+\Delta t)$ which is the probability that at time $t+\Delta t$ a report R that is created at time 0 is in the reports database of a peer o. Let Δt be small enough such that at most one report is generated in the system between t and $t+\Delta t$ and o can interact with at most one peer during the same time interval. $q(t+\Delta t)$ is the probability that one of the following mutually exclusive events happens:

1. o has R at time t, and it does not acquire any new report between t and $t+\Delta t$, and it does not interact with any peer between t and $t+\Delta t$. The probability for this to happen is $q(t)\cdot(1-\lambda\cdot\Delta t)\cdot(1-\frac{\mu}{N}\cdot\Delta t)$.

2. o has R at time t, and it does not acquire any new report between t and $t+\Delta t$, and it interacts with one peer o' between t and $t+\Delta t$, and o' does not have a report that is created after 0. The probability for this to happen is $q(t)\cdot(1-\frac{\mu}{N}\cdot\Delta t)\cdot\lambda\cdot\Delta t\cdot(1-g(t))$.

3. At time t o has no reports, or has a report that is created before 0, and it does not acquire any new report between t and $t+\Delta t$, and it interacts with one peer m' between t and $t+\Delta t$, and m' has R. The probability for this to happen is $(1-q(t)-g(t))\cdot(1-\frac{\mu}{N}\cdot\Delta t)\cdot\lambda\cdot\Delta t\cdot q(t)$.

Thus

$$q(t+\Delta t) = q(t)\cdot(1-\lambda\cdot\Delta t)\cdot(1-\frac{\mu}{N}\cdot\Delta t) + q(t)\cdot(1-\frac{\mu}{N}\cdot\Delta t)\cdot\lambda\cdot\Delta t\cdot(1-g(t)) +$$
$$(1-q(t)-g(t))\cdot(1-\frac{\mu}{N}\cdot\Delta t)\cdot\lambda\cdot\Delta t\cdot q(t)$$

By similar analysis, we obtain the following equation:

$$g(t+\Delta t) = \frac{\mu}{N}\cdot\Delta t + (1-\frac{\mu}{N}\cdot\Delta t)\cdot g(t) + (1-\frac{\mu}{N}\cdot\Delta t)\cdot(1-g(t))\cdot\lambda\cdot\Delta t\cdot g(t)$$

After simplification of the above difference equations, we get the following differential equations:

$$\begin{cases} \frac{dq(t)}{dt} = -(\lambda+\frac{\mu}{N})\cdot q(t) + 2\cdot\lambda\cdot q(t)\cdot(1-g(t)) - \lambda\cdot q(t)^2 \\ \frac{dg(t)}{dt} = \frac{\mu}{N} - \frac{\mu}{N}\cdot g(t) + \lambda\cdot g(t) - \lambda\cdot g(t)^2 \end{cases} \quad (2)$$

Since each peer is equally probable to acquire the report, $q(0)=1/N$. Finally, $g(0)=0$. Let $C(t)$ be the number of copies of a report t time units after its creation. We have the following theorem.

Theorem 2: $C(t)$ is a random variable with expected value $q(t)\cdot N$ where $q(t)$ is given by the equation group (2).

We used Theorem 2 to compute the expected number of copies as a function of time. We used the following set of parameter values: N=2500, λ=0.12, μ=10. The solid line in Figure 1 shows the result. Observe that the number first increases until a maximum value is reached. And then it decreases until disappearing from the system. From this figure we can estimate how far away a report can be propagated. Take the cut-off age beyond which the expected number of copies is below 1, which is about 60 seconds. Assume that the wireless transmission range is zero. Multiplying this cut-off age by the maximum speed of moving objects gives the maximum distance the report can be propagated to. For example, if the maximum speed is 60 miles/hour, then the maximum distance is 1 mile.

3.1.3 Validation of the Mathematical Model

We conducted a simulation to validate the analytical model. In this simulation, 2500 objects are initially uniformly distributed within a 5mile×5mile square area and they randomly move with a constant speed 40 miles/hour. The transmission range is 50 meters. This setup gives on average 0.12 interactions per each object per each second. Reports are generated with intensity 10 and each report is randomly assigned to an object. Figure 1 shows the results. The dashed line represents the experimental result. It can be seen that Theorem 1 accurately describes the behavior of the system.

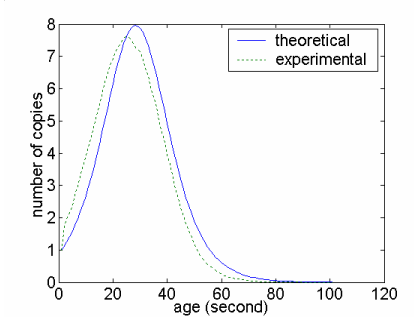

Figure 1: Number of copies of a report as a function of its age

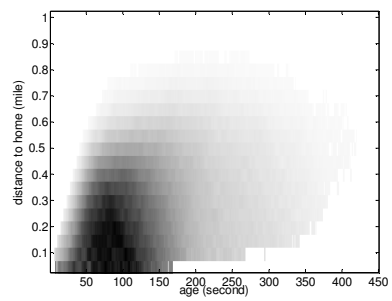

Figure 2: Propagation pattern of reports

3.2 Propagation Pattern of Reports for Spatio-temporal Resources

In this subsection we study by simulations the propagation pattern of reports for spatio-temporal resources. First we describe the simulation method and then we present the simulation results.

3.2.1 Simulation Method

We synthetically generated and moved objects within a 10mile×10mile square area. The objects move in a random-walk model. Specifically, for each object i, we randomly chose two points within the square area, and assigned them as the start point and the first stop of i respectively. i moves along line segment between the two points at a constant speed. When the first stop is reached, another random point is chosen as the second stop of i, and i moves from the first stop to the second stop at the same constant speed. And so on. The motion speed of i is randomly picked up from the interval $[v\text{-}5, v\text{+}5]$ where v is a parameter.

Hotspots are randomly distributed in the square area with density 500 hotspots per square mile. Resources are generated only at hotspots. At each hotspot, the length of the valid duration of a resource follows an exponential distribution with mean u seconds, and the time length of the invalid duration follows an exponential distribution with mean 360 seconds. The home of all the reports announced by a hotspot is the location of the hotspot. All the hotspots and the moving objects have the same wireless transmission range. All the moving objects have the same interest threshold. The value of the time decay factor α is $1/u$ and the distance decay factor β is $1/(u \cdot s)$ where s is the motion speed of a moving object.

There are five parameters for each simulation run, namely the interest threshold M, the wireless transmission range r, the constant speed v, the objects density g (i.e. the number of objects per square mile), and the mean of valid duration u. M is fixed to be 10, v is fixed to be 40 miles/hour, and u is fixed to be 120 seconds. The time unit is second. All the parameters and their values are listed in Table 1.

Table 1: Simulation parameters and their values.

Parameter	Symbol	Unit	Value
Mean of valid duration	u	second	120
Interest threshold	M		10
Transmission range	r	meter	50
Motion speed	v	miles/hour	40
Object density	g	objects/mile^2	100

Each simulation run is executed as follows. At the beginning of the simulation run, $10\times10\times g$ objects are generated and they start to move at the same time (time 0). Resources are generated and the status of each resource alternates between valid and invalid as described earlier. When the distance between two peers is smaller than r in a time unit, they exchange their reports, re-evaluate the relevance, and purge the least relevant reports if needed. Each exchange is finished instantaneously. The length of each run is 10 simulated hours.

During a simulation run, we trace the distribution of each report $a(R)$ at each time unit. For this purpose, we generate 50 rings centered at the home of R, each with the width of 0.05 mile. At each time unit we calculate the density of the copies of $a(R)$ at each ring, and average among all the time units of a simulation run.

3.2.2 Simulation Results

Figure 2 shows the average density as a function of the age and the distance to home for ORD. The density is coded by the gray-level, such that a deeper gray-level represents a higher density. The lowest gray-level (white) represents zero density. We make the following observations.

Observation 1: At any point in time, there is a spatial boundary for the distribution of the report, beyond which the density is zero. This boundary first expands, until a maximum value (about 0.9 mile) is reached. Then the boundary starts to shrink until finally the report disappears from the system (at the age of 450 seconds or so). The boundary expands at beginning because of the propagation of the report caused by opportunistic exchanges. However, as time passes, the relevance decreases, causing two effects: (i) more objects purge the report out; and (ii) less objects save it. These two effects make the number of copies start to decrease, and thus the boundary starts to shrink. After some time, the relevance becomes so low that all the objects that have carried it have purged it out, and no objects save it upon exchange. The report thus disappears from the system.

Observation 2: The gray-level tends to be deep when the distance to the home is small and it fades as the distance to home increases. In other words, the copies are more densely distributed in the areas close to the home than in the areas farther away. This is a useful behavior, because it means that the report has a higher availability in the area to which it is of interest.

The above propagation pattern shows that, by very simple local decisions made at each moving object, the opportunistic dissemination algorithm automatically limits the global distribution of a report to a bounded spatial area, which is a circle around the home location of the resource. The algorithm also limits the distribution to the time-

duration for which the report is of interest. We conducted experiments with more parameter configurations. These experiments show that the spatial and temporal boundaries automatically adapt depending on the number of resources in the system, the traffic density and speed, and other parameters that dictate the amount of storage, processing power, and bandwidth that should be allocated to each resource. For example, if resources are generated less frequently, then each report will stay in the system longer, and spread farther.

4 Comparison with Client/Server Model

With the ORD algorithm, a moving object *o* may have validity reports that are incorrect, i.e. the resources these reports refer to become invalid before *o* reaches them. When the validity reports in an object are used for decision making, an important measure is how many out of them are correct and how many are incorrect. In this section we compare ORD with the client/server model with this regard. In the client/server model, the validity reports are stored in a central database, and transmitted to the moving objects as query answers.

In subsection 4.1 we describe the client/server model. In 4.2 we define the performance measure. In 4.3 we describe the simulation setup, and in 4.4 we present the simulation results.

4.1 Client/Server Access (CS)

In the client/server model, there is a centralized database that stores validity reports. The database is updated by report producers and is queried by moving objects. A report producer inserts a validity report $a(R)$ to the database when a resource R is sensed. Each moving object *o* issues a continuous query to the centralized database. For example, when approaching the destination, *o* issues query ″acquire the top 10 relevant validity reports about parking slots″. At each time unit the centralized database evaluates each query and transmits *o* the top M reports (top with regard to *o*) where M is the interest threshold. The timestamp of each transmitted report is set to the transmission time. *o* replaces the current reports in its local reports database with the received reports. The centralized database knows the location of *o* at any point in time.

4.2 Performance Measure

Definition 1: A resource R is *correct* for a moving object *o* at time t if R remains valid when *o* reaches it under the condition that *o* goes to R at t. Otherwise R is *incorrect* for *o* at t.

According to the above definition, if R is invalid at t, then it is definitely incorrect for *o* at t. However, R may be incorrect for *o* at t even if R is valid at t. What matters is whether R is still valid when *o* reaches it.

Definition 2: A validity report $a(R)$ is *correct* for a moving object o at time t if R is correct for o at t. Otherwise $a(R)$ is *incorrect* for o at t.

Notice that in reality, at the time when o receives $a(R)$, it usually does not know whether $a(R)$ is correct or not, because it does not know whether R will remain valid when o reaches R. In this paper "correctness" and "incorrectness" of reports are defined solely for the purpose of performance evaluation.

Definition 3: Let K be the sum of the relevance values of the validity reports in o's reports database at time t. Let K' be the sum of the relevance values of validity reports in o's reports database that are correct for o at t. The *precision* of o's reports database at time t is K'/K.

We call the above measure "precision" because it indicates how many of the reports that o know are correct. This mimics the precision measure that is used in the information retrieval area. However, in our definition, precision is not the fraction of the correct reports known by o out of all the reports known by o. Instead, it is the ratio between the total relevance of the two sets. The reason for this is that the reports are unequal in relevance and therefore are different in importance for decision making. Thus when each report is counted for precision, it should be weighted by the relevance. With the same fraction of the correct reports, the higher relevance the correct reports occupy, the better data quality of the database. Next we define the notion of recall.

Definition 4: Let R be a resource that is correct for o at time t. The *relevance* of R to o at t is the relevance of the report $a(R)$ assuming that its timestamp is t.

Notice that above we define the relevance of a resource, as opposed to the relevance of a report defined in section 2.3. Intuitively the relevance of R to o at t is the relevance of the report $a(R)$ that has age 0. In other words, the relevance of R to o depends only on the distance of o from the resource.

Definition 5: Let C be the sum of the relevance values of the correct validity reports in o's reports database at time t. Let C' be the sum of the relevance values of the M correct resources in the system that are most relevant to o (i.e. the M closest correct resources). The *recall* of m's reports database at time t is C/C'.

We call the above measure "recall" because it indicates how many of the correct resources that o is interested in knowing (i.e. the correct top M in the system), are actually known by it. However, notice that in our definition, recall is not the fraction of resources correctly known by o out of the top M resources in the system. Instead, it is the ratio between the total relevance of the two sets.

Definition 6: Let P be the precision of o's reports database at time t, and Q the recall of the database at t. The *precision-recall product* of the database at t is $P \cdot Q$.

In this paper the precision-recall product is used as the performance measure.

4.3 Simulation Method

The simulation setup is similar to the one used in 3.2, except for the following. First, the CS algorithm is implemented. With the CS algorithm, at each simulated time unit the system sorts all the resources in the order of their relevance to o and puts the top

M to *o*'s database. Second, in ORD an exchange between two moving objects succeeds with probability *p*. *p* is called the *successful interaction probability*.

Let us emphasize that in the simulations for the CS model we assume that the bandwidth available for the communication between the server and the clients is infinite, and we ignore the contention/collisions that may cause transmission failures. In the simulations for the ORD algorithm, on the other hand, we take into considerations the bandwidth constraint and contention/collisions, in the following way. First, we calculate the bandwidth required by the ORD algorithm with the most bandwidth-consuming parameter configuration in Table 1. This configuration is object density g = 2500 per square mile and transmission range r = 200 meters. With this configuration, the bandwidth consumption of the ORD algorithm is 40KBytes per second per object[2]. On the other hand, we estimated, for the above network configuration, with contention and collisions, the effective bandwidth available for each object is 56KBytes per second when 802.11g ([6]) is used. The estimation is extrapolated from the empirical results of [7][3]. This suggests that even with the most bandwidth-consuming parameter configuration, the bandwidth consumption of the ORD algorithm does not exceeds the network capacity. Therefore in all the experiments with ORD, the successful interaction probability p is set to be 1, except for the group that study the impact of p.

The above justification assumes a single resource type. If there are multiple resource types, then with the above network configuration, the bandwidth consumption of ORD may exceeds the network capacity. In that case the successful interaction probability p is lower than 1, which we will also study.

Table 2: Variable simulation parameters.

Parameter	Symbol	Unit	Value
Transmission range	r	meter	50, 100, 150, 200
Object density	g	objects/mile^2	100, 500, 1000, 1500, 2000, 2500
Successful interaction probability	p		0.05, 0.1, 0.15, 0.2, 0.25, 0.5, 0.75, 1

Finally, let us mention that we repeated the experiments for motion in a grid network rather than Euclidean space. The reason is that in our traffic applications vehicles move along a road network. We determined that the network results are very similar to those reported here for Euclidean space. Due to space limitations these network results are omitted. The parameters are listed in Table 2.

4.3 Simulation Results

Impact of object density. Figures 3, 4, and 5 show the precision-recall product of ORD as functions of the object density and compare these measures with those of CS.

---

[2] Notice that the moving object does not transmit any report to a hotspot.

[3] The reference analyzes the effective bandwidth available per object for a short-range wireless technology, with contention and collisions taken into considerations.

For ORD, the performance increases as the object density increases. Intuitively, as the object density increases, the interactions among objects become more frequent, and thus the newly generated reports get propagated more quickly. These reports purge the old reports out of the databases of moving objects. Since the new reports are more likely to be correct than the old ones, both the precision and the recall increase. Notice that the precision-recall product of CS is not 1. This is because the central database does not know when a resource R will become invalid, and therefore the report $a(R)$ returned by CS to a moving object o may be incorrect (i.e. R is invalid when o reaches R).

The figures show that the performance of ORD approaches to CS when the transmission range is 50 meters and the object density is 2500 objects per square mile. In this situation, the average distance between two neighboring moving objects is about 32 meters, smaller than the transmission range. In other words, most of the time the network formed by moving objects is connected. In this case, with ORD the propagation of a report reaches its maximum spatial boundary almost instantaneously. However, this is different than a simple flooding in a connected network. In our model each moving object only transmits top M reports and the spreading of each report is automatically restricted within a small portion of the whole network (see the propagation analysis in section 3).

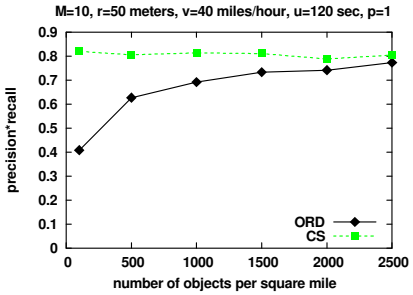

Figure 3

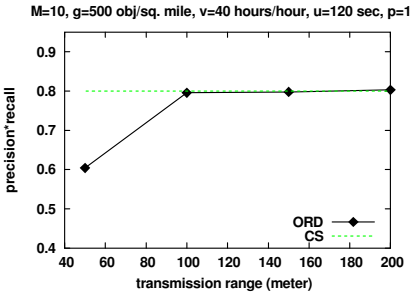

Figure 4

Impact of transmission range. Figure 4 shows the precision-recall of ORD as functions of the transmission range and compare with those of CS. Intuitively, as the object density increases, the interactions among objects become more frequent, which generates similar effects as when the object density increases.

Impact of successful interaction probability. Figure 5 shows the performance measures of ORD as functions of the successful interaction probability. Intuitively, as the successful interaction probability increases, the effective interactions among objects become more frequent, which generates similar effects as when the object density increases. In fact, comparing Figure 5 and Figure 3, we notice that reducing the successful interaction probability is the same as simply reducing the object density. For example, the effect of a successful interaction probability of 0.25 is the same as reducing the object density from 2000 to 1000.

Observe that we tested with a long range of probability values, from 0.05 to 1. This is because the successful interaction probability is used to model not only communica-

tion reliability, but also the ORD implementations in which not every encounter generates an interaction. For example, if a moving object broadcasts for every 20 time units, or the communication module is awake for 5% of time, then the successful interaction probability is 0.05.

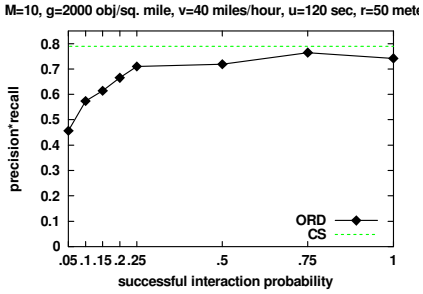

Figure 5

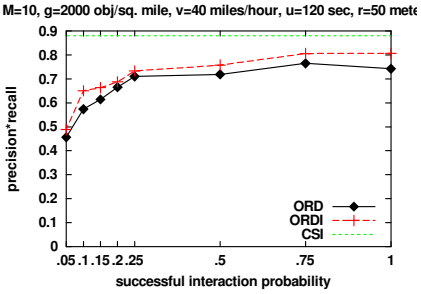

Figure 6

5 Opportunistic Report Dissemination with Invalidation (ORDI)

In this section we first present the ORDI algorithm in which invalidity reports are generated and propagated to reduce the fraction of incorrect reports in a moving object's reports database. Then we compare ORD, ORDI, and the client/server model with invalidation.

5.1 Description of the ORDI Algorithm

At each report producer o, whenever a resource R is detected invalid, o creates an *invalidity report* $i(R)$. $i(R)$ contains the following four attributes: (i) *resource-type* the resource type of R; (ii) *resource-id* the id of R; (iii) *timestamp* the time when report $i(R)$ is created; and (iv) *home-location* the home of R.

After $i(R)$ is produced, the validity report of R is removed from the reports database DB_o(*resource-type*), and $i(R)$ is inserted into DB_o(*resource-type*). In order to distinguish between validity reports and invalidity reports, each report is given an extra attribute *report-type* when it is inserted into a reports database. *report-type* indicates whether the report is a validity report or an invalidity report.

The invalidity report uses the same relevance function as a validity report, and is exchanged similarly to a validity report. The only difference is as follows. When an invalidity report $i(R)$ is received by an object o, o uses the resource-id attribute to search $a(R)$ in o's reports database. If $a(R)$ is found and its timestamp is smaller than that of $i(R)$, then the validity report is replaced by $i(R)$. If the validity report is not found, then $i(R)$ is either discarded or saved into DB_m(*resource-type*) based on its relevance, in the same way a validity report is treated. The reason $i(R)$ is saved is to invalidate a validity report that may arrive later.

5.2 Comparison with ORD and the Client/Server Model

In this subsection we compare ORDI with two algorithms. One is ORD and another is client/server with invalidation, or *CSI*. CSI works similarly as CS except the following. At each report producer o, whenever a resource R is detected invalid, o removes the report $a(R)$ from the centralized database. Thus the centralized database will not include $a(R)$ in any query answer since then.

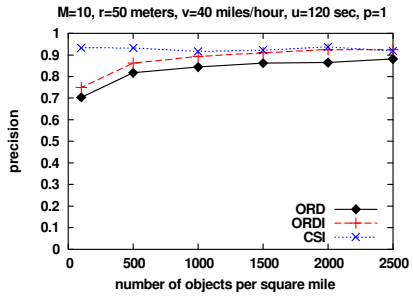

Figure 7(a)

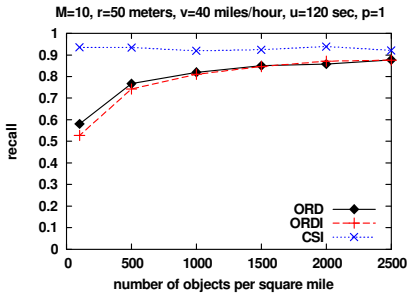

Figure 7(b)

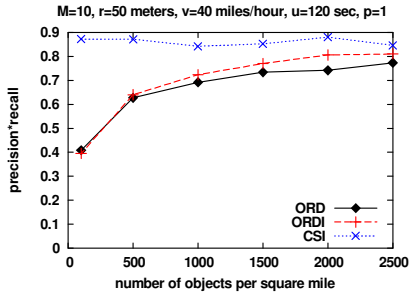

Figure 7(c)

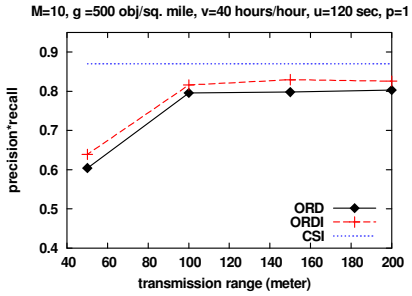

Figure 8

The simulation setup is the same as described in subsection 4.3. The results are shown in Figures 6, 7, 8. First let us compare ORDI and ORD. The precision of ORDI is higher than that of ORD (Figure 7(a)). This is because invalidation reports push incorrect reports out of the reports database and therefore the fraction of correct reports increases. However, there is little difference between ORDI and ORD in recall (Figure 7(b)). This is somehow surprising because invalidity reports share the same database with validity reports and therefore they may occupy spaces that could have been used to accommodate correct reports. The fact that ORDI is as good as ORD in recall suggests that in most cases invalidity reports occupy the spaces of incorrect reports, which is a desirable behavior. As a result, the precision-recall product of ORDI is better than that of ORD (Figure 7(c)).

Now let us compare ORDI and ORD with CSI. The precision of ORDI approaches to CSI beyond certain object density (Figure 7(a)) or transmission range (figure omit-

ted due to space limitations). This is similar to the observation we have in subsection 4.4 when comparing ORD and CS, and the explanation is also similar. However, neither ORDI nor ORD ever reaches CSI in recall (Figure 7(b)). ORDI never reaches CSI in recall, because with ORDI, invalidity reports share the same database with validity reports, whereas with CSI there are no invalidity reports in a moving object's reports database. In other words, the number of validity reports in a reports database with CSI is higher than that with ORDI. ORD never reaches CSI in recall, because with ORD there can be reports that refer to the resources that have become invalid; with CSI there are no such reports. As the result, ORDI and ORD never reach CSI in precision-recall product (Figures 7, 8).

The effect of the successful interaction probability to ORDI is similar to that to ORD (see Figure 6).

6 Relevant Work

Resource discovery and publish/subscribe in mobile ad hoc networks are usually implemented by building a routing structure for resource information (see e.g. [8, 9, 11, 15]). Most of these works rely on routing structures. However, the constructed routing structure may easily become obsolete in a highly dynamic and partitionable network environment. Work has also been done on data dissemination in mobile peer-to-peer networks [10, 12, 14, 16, 18, 20, 21]. These methods use the gossiping/epidemic communication paradigm. However, they consider dissemination of regular data objects rather than spatio-temporal resources, and they do not rank the resources for determining what to broadcast.

This paper differs from our prior work (e.g. [19, 22]) on the same topic in multiple aspects. The theoretical analysis of the propagation pattern is new. The comparison with the client/server model is new, and so is the invalidation algorithm.

7 Conclusion

In this paper we devised an algorithm, ORD, for dissemination of spatial and temporal resource-information in a mobile peer-to-peer environment, in which the resource-information database is distributed among the hotspots and moving objects. We analyzed ORD theoretically, using differential equations, and experimentally, using simulations. We compared ORD with the client/server model by simulations. The performance measures are relevance-weighted precision and recall. We determined that ORD performs better when the object density and the wireless transmission range increase. ORD reaches the client/server model in performance when the object density or the transmission range is high enough. We also studied the impact of successful interaction probability, and determined that reducing the successful interaction probability is the same as simply reducing the object density. Thus ORD can perform as well as the client/server model in low success probability environment by increasing the object density. Finally, we studied a variant of ORD, ORDI, which uses invalidity

reports to increase precision. The experimental results show that ORDI is better than ORD.

References

1. IEEE Computer Society. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.1997.
2. J. Haartsen, et al. Bluetooth: Vision, Goals, and Architecture. ACM Mobile Computing and Communications Review, 2(4):38-45, October 1998.
3. http://www.ubisense.net/technology/uwb.html
4. http://firechief.com/ar/firefighting_roborescuers_increase_disaster/
5. Markowetz, et al. Exploiting the Internet As a Geospatial Database, International Workshop on Next Generation Geospatial Information, 2003.
6. IEEE 802.11g Standard. http://grouper.ieee.org/groups/802/11/index.html
7. J. Li, et al. Capacity of Ad Hoc Wireless Networks. MobiCom 2001.
8. Saumitra M. Das, Himabindu Pucha, and Y. Charlie Hu. Ekta: An efficient dht substrate for distributed applications in mobile ad hoc networks. WMCSA 2004.
9. Y. Hu, et al. Exploiting the synergy between peer-to-peer and mobile ad hoc networks. In HotOS-IX, 2003.
10. W. Zhao, M. Ammar, E. Zegura. A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Networks. Mobihoc, Tokyo Japan, May 2004.
11. C. Frank, et al. Consistency challenges of service discovery in mobile ad hoc networks. MSWiM 2004.
12. F. Perich et al. On Data Management in Pervasive Computing Environments. IEEE Trans. Knowledge and Data Engineering, 16(5), May 2004.
13. M. J. McGlynn and S. A. Borbash. Birthday protocols for Low Energy Deployment and Flexible Neighbor Discovery in Ad Hoc Wireless Networks. Proc. of MobiHoc 2001.
14. K. Rothermel, et al. Consistent Update Diffusion in Mobile Ad Hoc Networks. Technical Report 2002/04, CS Department, University of Stuttgart, 2002.
15. Y. Huang and H. Garcia-Molina. Publish/Subscribe Tree Construction in Wireless Ad-Hoc Networks. MDM 2003, pages 122-140.
16. M. Papadopouli and H. Schulzrinne. Effects of Power Conservation, Wireless Coverage and Cooperation on Data Dissemination Among Mobile Devices. MobiHoc 2001.
17. Datta, et al. Updates in Highly Unreliable, Replicated Peer-to-Peer Systems. ICDCS 2003.
18. Vahdat and D. Becker. Epidemic Routing for Partially Connected Ad Hoc Networks, Technical Report CS-200006, Duke University, April 2000.
19. O. Wolfson, B. Xu. Opportunistic dissemination of spatio-temporal resource information in mobile peer-to-peer networks. In PDMST'04, 2004.
20. U. Centintemel, et al. Power-efficient Data Dissemination in Wireless Sensor Networks. MobiDE 2003.
21. S. Goel, et al, Grassroots: A Scalable and Robust Information Architecture. Technical Report DCS-TR-523, CS Department, Rutgers University, June 2003.
22. O. Wolfson, B. Xu, Y. Yin. Dissemination of Spatial-Temporal Information in Mobile Networks with Hotspots. *DBISP2P*, 2004.