Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2010, Article ID 185791, 11 pages
doi:10.1155/2010/185791

*Research Article*

# A Probabilistic Protocol for Multihop Routing in VANETs

**Junichiro Fukuyama**

*Toyota InfoTechnology Center USA, c/o Telcordia Technologies, One Telcordia Drive, Piscataway, NJ 08854-4157, USA*

Correspondence should be addressed to Junichiro Fukuyama, jfukuyama@us.toyota-itc.com

Received 25 February 2010; Accepted 10 August 2010

Academic Editor: Soumaya Cherkaoui

Vehicular ad hoc networks (VANETs) allow communications over sequences of vehicles with radio devices. There are many possible applications over a VANET such as traffic jam warning, collision warning, parking lot reservations, camera picture feed , and so forth. There have been quite a few results in the area seeking for a fast and reliable communication protocol due to their potential. VANETs, however, are pointed out as difficult for numerical optimizations due to frequent changes in their topologies. As a result, heuristic methods such as GPSR have been mainly used for routing packets over multihop communications. In this paper, we present an algorithm to precompute the probability that communication is possible between specified source and destination in a VANET, under certain mathematical assumption. The proposed new protocol for multihop communication refers to a lookup table containing the precomputed data to decide a good packet forwarder quickly. We create a simulation testbed that seems challenging for all the existing multihop routing protocols for VANETs, in which we test ours. We see much improved performances over GPSR after the algorithm is refined for some practical issues.

## 1. Introduction

Vehicular ad hoc networks (VANETs) enable communication between vehicles or between a vehicle and infrastructure. The idea of having intervehicle communications connected to a wired network has been investigated since the 1980s. Over a VANET, we can achieve traditional safety applications such as collision, icy road, and red light warnings, as well as nonsafety applications such as traffic information dissemination, reservation query, camera picture feed and so forth. Recently, there has been an emerging trend of utilizing mobile communication for environmental issues. It is possible to obtain significant information from VANETs to improve the uses of gas or other resources.

When there are not sufficient roadside units (RSUs) or direct communications between distant vehicles are preferred, it usually takes more than one step of vehicle-to-vehicle (V2V) communications to send information from a specified source to destination. The transmission range of a radio device is normally 100–200 m for V2V, which is much smaller than the dimension of the considered area. Researchers have studied such multihop communications extensively not only because VANET applications have a large market potential, but also they are scientifically interesting [1–5].

It has turned out very difficult to optimize parameters of a VANET due to its highly mobile nature. Links can be disconnected so frequently that it is occasionally impossible to accurately predict the existence of an end-to-end connection. Reference [6] sheds light on the theoretical aspect of this issue. As a result, carry-and-forward type heuristic methods have been mainly used for routing packets over multihop communications in VANETs.

The current most common scheme for source-destination routing in VANETs is the so-called geographical forwarding. It chooses car(s) closest to the destination $d$ and pass the packet. Geographical forwarding was first introduced as GPSR in [7]. The result has been referred to many other papers that seek a better performance by similar methods. One of them, called PBRV [8], resolves the routing loop problem of GPSR.

The main advantage of geographical forwarding is its computational speed per hop. It is a simple task to calculate the distance to $d$ for every car in the proximity, and quick computation is very important to forward a packet in short latency. However, we can easily create counter examples for

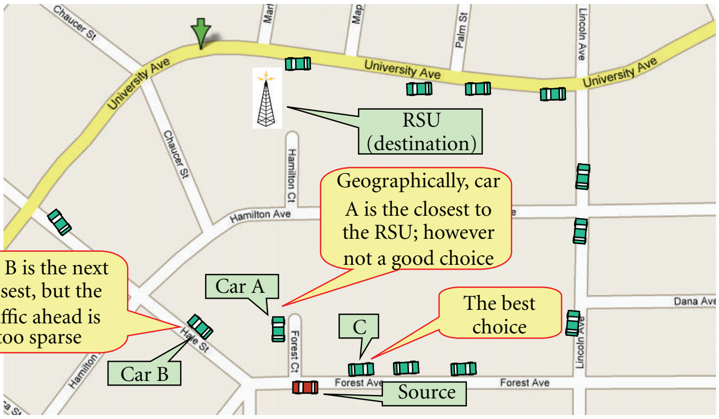


Figure 1: A counter example for geographical forwarding.

geographical forwarding such as the ones seen in [2] and Figure 1. A similar situation may happen more frequently when the market penetration rate, that is, the ratio of the cars equipped with radio devices, is low.

Another protocol called MDDV is developed in [9]. It uses the following two in addition to the geographical forwarding.

(i) *Trajectory-based forwarding:* first determine an approximate trajectory from the source to the destination on the map then only considers cars that roughly follow it.

(ii) *Opportunistic forwarding:* chooses some cars that meet certain conditions to give them the right to broadcast the packet in the proximity.

MDDV is designed for an arterial road or highway where considered cars are basically running in the same direction. It is reported to produce short latency and high delivery ratio in such a scenario but does not have a mechanism to treat a downtown situation.

Recently, a protocol called VADD has been proposed [4]. It considers the probability for a packet to reach the destination in addition to the positions and directions of the moving cars in the proximity. VADD handles a situation as in Figure 1 better than the previous two. It, however, calculates the probability at every intersection, seeming to have large invehicle computational time. It is based on the idea of dynamic path-selection approach. Nevertheless, one can point out that a large amount of the input information to VADD is static: the fixed probability for a packet to be sent successfully from one intersection to another.

There have been some other forwarding schemes known such as delaybounded routing [3], CAR [2], and GeOpps [1].

In this paper, we present an algorithm to precompute the probability that the communication is possible between specified source and destination in a VANET, under certain mathematical assumption. We propose a new protocol for multihop communication that refers to a lookup table containing the pre-computed data to decide a good packet forwarder quickly.

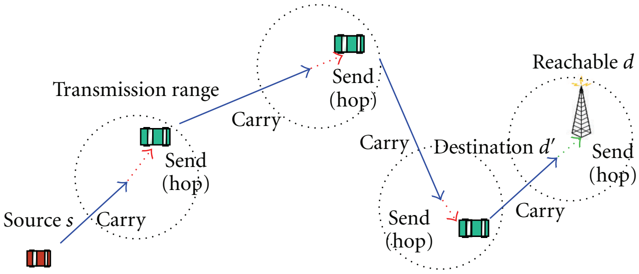


Figure 2: An example of a communication path.

We create a simulation testbed that seems challenging for all the existing multihop routing protocols for VANETs, in which we test ours. After the algorithm is refined for some practical issues, it showed dramatically improved performances over GPSR.

The rest of the paper is organized as follows: in Section 2, we describe our general method to pre-compute such probabilities. Section 3 shows the simulation results, followed by conclusions stated in Section 4.

## 2. A Method to Precompute the Probability of Multihop Communication

The main idea of the probabilistic protocol we propose in this paper is to compute in advance the probability that it is possible to forward the packet from the source $s$ to destination $d$. We call it the communication probability from $s$ to $d$.

Suppose for simplicity that two cars can communicate whenever they are within a given transmission range $R$ of the radio. We will modify it later so the algorithm may better suit real situations. With this simplification, the communication from $s$ to $d$ is possible if and only if there is a sequence of cars to forward the packet as shown in Figure 2. We call such a sequence a communication path. Now, our communication probability is the probability that such a path exists. In this section, we present an efficient algorithm to compute the probability under certain assumption as well as a general way to measure its inputs. We will also consider how to refine the algorithm for practical use.

*2.1. Basic Notation.* We assume that every considered object is on the map $M$ that is a Euclidean graph whose nodes are geographical locations or *positions*, and edges are roads. Denote by $x$ a position on $M$, for which we write $x \in M$ instead of $x \in V(M)$. Let $m$ be the number of positions on $M$. Notice that $x$ is a two-dimensional vector.

Let $R$ be the *transmission range* of the radio. As stated before, we assume at this moment that any two cars within the distance can communicate with each other. A position $x' \in M$ is called an *R-neighbor* of $x \in M$ if the Euclidean distance from $x$ to $x'$ is at most $R$.

Consider that time $t = 0, 1, 2, \ldots, T$ where $T$ is the *time limit*. To express that a car is at position $x \in M$ when time is $t$, it is convenient to couple $t$ with $x$. A *space time* $u = (x, t)$ is meant to be such pairing.

A *carry* is a pair $h = (u_1, u_2)$ of space times. It means that a car appears at $u_1 = (x_1, t_1)$ and moves to $u_2 = (x_2, t_2\ )$. We can understand carries as the solid arrows in Figure 2, while the dotted ones represent *hops*. We say $x_1$ and $x_2$ are the *source* and *destination* of the carry, and $t_1$ and $t_2$ are *start* and *arrival time*, respectively.

Formally, a *communication path* is a $k$-tuple $E = (h_1, h_2, \ldots, h_k)$ of carries $h_i$ such that the start time of $h_{i+1}$ equals the arrival time of $h_i$, and the source of $h_{i+1}$ is an $R$-neighbor of the destination of $h_i$ for every $i < k$. We require the integer $k$ to be at most a given *maximum number H of hops*. We can also say that $E$ is a communication path from $u$ to $u'$, where $u$ is the first space time of $h_1$ and $u'$ the second one of $h_k$. The arrival time and destination of $E$ are defined as those of $u'$, respectively.

Let a source $s$ and destination $d$ be given. A space time $u = (x, t)$ is said to be *reachable from* $(s, 0)$ *in k hops* if there exists a communication path from $(s, 0)$ with no more than $k$ carries whose destination is an $R$-neighbor of $x$ and arrival time is at most $t$. The *communication probability* $\rho(s, d, T)$ from $s$ to $d$ with time limit $T$ is the probability that $(d, T)$ is reachable from $(s, 0)$ in $H$ hops.

Figure 2 shows a communication path from $(s, 0)$ to $(d', t)$ with 4 carries and 3 hops. The space time $(d, t)$ is reachable from $(s, 0)$ in 4 hops.

Our goal in this mathematical framework is to compute $\rho(s, d, T)$ efficiently. Once we have the communication probabilities, the averages of other various random variables could be computed quickly. For example, the expected delay time $\tau(s, d)$ to send a packet from $s$ to $d$ is

$$\begin{aligned}\tau(s,d) &= \sum_{t=1}^{T} t\bigl(\rho(s,d,t) - \rho(s,d,t-1)\bigr) \\ &= T\rho(s,d,T) - \sum_{t=0}^{T-1} \rho(s,d,t)\ .\end{aligned} \tag{1}$$

*2.2. Assumptions to Reduce the Hardness of Computation.* Denote by $(A, u)$ the probabilistic event that car $A$ appears at space time $u$ and by $P(A, u)$ its probability. In the rest of the paper, $P(\cdot)$ expresses the probability of the argument event, where we omit obvious parentheses, that is , write $P(A, u)$ instead of $P((A, u))$ and so forth. Let $\Omega$ be the set of all $(A, u)$. Logical sum, product, and implication are denoted by $\bigvee$, $\bigwedge$, and an arrow, respectively. Consider the discrete probability space $(\Omega, \mathcal{F}, P)$ as in Appendix A, where $\mathcal{F}$ is the set of events over $\Omega$. We will have all formal discussions in this framework.

The probabilities $P(A, u)$ are not sufficient information to compute the exact values of $\rho(s, d, T)$, because the events $(A, u)$ possibly depend on each other even for a single car $A$. For example, $P((A, u_1) \bigwedge (A, u_2))$ may not be $P(A, u_1)P(A, u_2)$ due to the dependence; if paths from the source $s$ to $u_i$ overlap, $P(A, u_i)$ affect each other.

We formulate the problem of computing communication probabilities as follows; given a map $M$, initial positions of cars, time limit $T$, and transmission range $R$, assume that subsequent movements of two cars are independent of each other. That of every car $A$ is restricted so that $P((A, u_2) | (A, u_1))$ is a given fixed value, where $u_1 = (x_1, t)$ and $u_2 = (x_2, t + 1)$ are space times, and $P(\cdot|\cdot)$ denotes conditional probability as defined in Appendix A. It means that the probability of $A$ moving from $x_1$ to $x_2$ is given when time transits from $t$ to $t + 1$. Compute the communication probability from $s \in M$ to $d \in M$ for all the pairs of $s$ and $d$ from the above information. This problem seems hard to compute in polynomial time. We, in fact, conjecture that it is NP-complete even when the input instance is restricted such that the number of possible routes taken by each $A$ is bounded by a polynomial in the problem size. (We define the size as the number of cars added by $m + T + H$.) The difficulty is attributed to the dependence between communication paths that are events in $\mathcal{F}$.

We introduce the following assumption to reduce the hardness of computation. Denote by $[u, k]$ the event in $\mathcal{F}$ that the space time $u$ is reachable from $(s, 0)$ in $k$ hops.

*Steady traffic assumption*: the following two are true in $(\Omega, \mathcal{F}, P)$.

(i) All the carries are independent.

(ii) $[u, k]$ for each $k \le H$ are independent.

If it is true, the probability $P([u, k])$ takes a value that only depends on $u$ for each fixed $k$, letting the reachability to $u$ not affect each other. Then, the algorithm shown in Section 2.3 will compute the communication probability exactly.

There are some input instances for which the assumption holds strictly. One such case is significant: when we have the exact schedule of every car, or $(A, u)$ satisfy the following.

*Exact schedule assumption*: $P(A, u)$ is either 1 or 0 for every car $A$ and space time $u$.

It means that it is either true or false if $A$ appears at $u$. The following lemma says that it is only a special case of the steady traffic assumption.

**Lemma 1.** *The exact schedule assumption implies the steady traffic assumption.*

*Proof.* Suppose that $P(A, u)$ is either 0 or 1 for every $(A, u)$. The probability of $(A, u)$ given any condition is either 0 or 1. The probability that a communication path occurs is 0/1. Thus, $P([u,k])$ is also 0/1 for any $k \le H$. The event $[u,k]$ is independent of the others as the last paragraph in Appendix A says. Since carries $(u,k)$ have probabilities 0 or 1, it implies the steady traffic assumption. □

Thus, if we have exact schedules of the cars, we can simply use the algorithm in Section 2.3 setting $P(A,u) = 1/0$ to compute the exact communication probability.

A natural question occurs; how true is steady traffic assumption in real traffic scenes? We think that $[u,k]$ and carries are well independent in practice if considered $u = (x,t)$ are distinct enough. Events $[u,k]$ with similar $x$ and $t$ affect each other, since they share large common parts of major communication paths. In Section 3, we compare the performance of the probabilistic protocol with that of GPSR in a challenging simulation testbed. We choose 100 m and 5 seconds as the unit differences of $x$ and $t$, respectively, to avoid considering similar $u$ too often. With the coarse resolutions, the probabilistic protocol will successfully capture the tendency of the vehicle mobility to contribute to good communication performances.

*2.3. The Basic Algorithm.* We show below our basic algorithm to compute communication probabilities. Its primary input is the *carry probability* $P(u_1,u_2)$ that there exists a car that appears at the space time $u_1$ and moves to $u_2$ for every carry $(u_1,u_2)$.

Notice that

$$(u_1,u_2) = \bigvee_{\text{every car } A} (A,u_1) \bigwedge (A,u_2), \tag{2}$$

in the probability space $(\Omega,\mathcal{F},P)$, since the carry $(u_1,u_2)$ is the event in $\mathcal{F}$ that some $A$ appears at $u_1$ and moves to $u_2$. Recall the problem of computing communication probabilities formulated in Section 2.2. Since the conditional probability of $(x_2,t+1)$ given that $(x_1,t)$ is input for all $x_i$ and $t$, it means that the probability of the event $(A,u_1)\bigwedge(A,u_2)$ is determined uniquely. Movements of distinct cars are independent. The probability of the carry $(u_1,u_2)$ is uniquely determined by the input instance of the problem.

Now, assume the steady traffic assumption. *SteadyTraffic* illustrated in Algorithm 1 computes the communication probability for any given pair of source and destination.

Its correctness proof is found in Appendix B. We will present a general method to obtain $P(u_1,u_2)$ from field data in Section 2.4 and a more practical version in Section 2.5.

The asymptotic running time of *SteadyTraffic* is found $O(Hm^2T^2R^2)$ in a straightforward manner. In our simulation test with a careful choice of the parameter set, it runs in few minutes to finish computation for each $s$ and $d$.

If the exact schedule assumption holds, $P(u_1,u_2)$ is 1 if and only if there exists a car $A$ such that $P(A,u_1)\ P(A,u_2) = 1$. We can input them into *SteadyTraffic* to compute $\rho(s,d,T)$.

*2.4. To Measure Carry Probabilities from Field Data.* The primary input to Algorithm *SteadyTraffic* is the probabilities of $P(u_1,u_2)$ of the carries. Theoretically, all we need is $P((A,u_1)\bigwedge(A,u_2))$ for every car $A$ and space times $u_i$ that is, the probability that the given car $A$ appears at $u_1$ and moves to $u_2$. Then, the carry probability is computed as

$$P(u_1,u_2)0 = 1 - \prod_{\text{every car } A} \Big(1 - P\Big((A,u_1)\bigwedge(A,u_2)\Big)\Big), \tag{3}$$

by (1) and (3).

There are ways to compute $P((A,u_1)\bigwedge(A,u_2))$ from field data. The following two steps provides a general one.

*Measure the intersection probabilities.* Some nodes $v$ on the map $M$ represent intersections. The edges $l$ from $v$ are the roads from $v$. The *intersection probability* $Q(l,l\prime,v)$ is the probability that a car moves from the road $l$ to $l'$ at the intersection $v$. We first measure this quantity from real field data, by simply finding the ratio of the cars to choose the road $l'$ out of all the cars running toward $v$ on $l$.

*Calculate* $P((A,u_1)\bigwedge(A,u_2))$. We repeat the process below for a number of times to average the obtained values of $P((A,u_1)\bigwedge(A,u_2))$. At the beginning of each round, we configure the initial positions of the cars randomly but based on the real statistics. Then, do the following:

(1) fix each carry $(u_1,u_2)$ and car $A$. Let $u_i = (x_i,t_i)$ for $i = 1,2$.

(2) At time $t = 0,1,2,\ldots,t_1$, simulate the probabilistic movements of the car using the intersection probabilities so that we know the probability of $A$ being at $(x,t)$ for all $x \in M$ and $t \le t_1$. In the end, we know the probability of $A$ at $u_1 = (x_1,t_1)$. Store it into $p_1$.

(3) Fix the position of $A$ at $t = t_1$ as $x_1$. Simulate the probabilistic movements of $A$ at time $t = t_1+1, t_1+2,\ldots,t_2$. Find the probability of $A$ at $(x_2,t_2)$, and store it into $p_2$.

(4) Now, $P((A,u_1)\bigwedge(A,u_2)) = p_1 \cdot p_2$ is the probability that $A$ appears at $u_1$ and moves to $u_2$.

*2.5. Practical Refinements.* Although the method described so far is mathematically correct for the presented mobility model, we faced essential difficulties when we implemented *SteadyTraffic* on a testbed described in the next section. Abstracting the problems, we devised a variant of *SteadyTraffic* that is meant to be used for practice. It is illustrated in Algorithm 2 as *SteadyTraffic 2*.

We summarize below general deployment problems and our solutions in the modified algorithm. First, values computed by *SteadyTraffic* may include propagated errors. As stated as the steady traffic assumption in Section 2.2, we want the reachability events $[u,k]$ and carries to be well independent of others. If we consider $[u,k]$ with similar space times $u = (x,t)$ too often, it is likely to increase the calculated communication probability much more than the real value. In other words, space times whose car density is large may not be much different from those with small density for the algorithm *SteadyTraffic*.

```
Algorithm SteadyTraffic
 Inputs:
    $T$: time limit,
    $H$: maximum number of hops,
    $M$: map with $m$ positions,
    $R$: transmission range,
    $P(u_1, u_2)$: carry probability for every $(u_1, u_2)$, that is, probability
that a car appears at space time $u_1$ and moves to $u_2$,
$s, d \in M$: source and destination on the map $M$.
Output:
    Communication probability $\rho(s, d, T)$ from $s$ to $d$ with time limit $T$.
begin
    (1). /* The final value of Answer $(x, t, k)$ will be the probability that there
        exists a communication path from $(s, 0)$ to $(x, t)$ with at most $k$ carries. */
    (2). For every space time $u = (x, t)$ do Answer $(x, t, 0) = 0$;
    (3). Answer $(s, 0, 0) = 1$;
    (4). for $k = 1$ to $H$ do
        4-1. for every space time $u = (x, t)$ do
            4-1-1. Answer $(x, t, k) = P((s, 0), u)$;
            4-1-2. for every other space time $u_1 = (x_1, t_1)$ such that $t_1 \le t$ do
                    4-1-2-1. /* Let $D(x)$ for $x \in M$ be the set of $R$-neighbors of $x$ */
                    4-1-2-2. $p_1 = 1 - \prod_{x_2 \in D(x_1)}(1 - \text{Answer } (x_2, t_1, k-1))$;
                    4-1-2-3. $p_2 = P(u_1, u) \cdot p_1$;
                    4-1-2-4. Answer $(x, t, k) =$ Answer $(x, t, k)(1 - p_2) + p_2$;
            4-1-3. end for
        4-2. end for
    (5). end for
    (6). return $1 - \prod_{\substack{x \in D(d) \\ t \le T}}(1 - \text{Answer } (x, t, H))$;
 end
```

Algorithm 1

We multiply, to handle the problem, carry probabilities $P(u_1, u_2)$ by $(e(x_1)e(x_2))^{\beta}$, where $u_i = (x_1, t_i)$ are space times, $e(x)$ are the expected car densities at $x \in M$, and $\beta \in (0, 1)$ is a numeric parameter to adjust. Step 3-1-2-1 of *SteadyTraffic2* performs this data modification. It has an effect of emphasizing the differences between large and small car densities. We will choose $\beta = 1/2$ in Section 3.

We could also use coarse space and time resolutions to prevent the error propagation, that is, to use large values of unit differences of $x$ and $t$ to avoid introducing similar $u = (x, t)$. It also speeds up the algorithm significantly.

Loop 3-1-2 of *SteadyTraffic2* has a slightly different structure from 4-1-2 of *SteadyTraffic*; the former finds any new carry $(u_1, u)$ in the proximity of $u_2$ such that i)–v) and updates Answer$(x, t, k)$ at $u$ simply assuming that the newly detected communication path is independent of the others. Condition iii) discards carries of low probabilities.

The above schemes of simplification and data modification effectively compute good approximate values of communication probability with reasonable computational time. This practically solves our first problem of error propagation.

Secondly, we have a node differentiation problem. Even if the communication probabilities are accurate, it is occasionally hard in run time to choose a good packet forwarder among the nearby vehicles. We illustrate the problem in Algorithem 1. To resolve it, we restrict considered communication paths so that they are not extended in a specified direction. Conditions iv) and v) of Loop 3–1–2 filter every path with $x$- or $y$-component running in the opposite direction to given *dir*. We call it *north*, *south*, *west,* and *east mode* when $dir = (0, -1), (0, 1), (1, 0),$ and$(-1, 0)$, respectively. (The map origin is supposed to locate at the upper-left corner.) For example in the north mode, *SteadyTraffic 2* does not extend a communication path to location more southern than the current position. It is suggested to combine values of more than 1 modes to refer to in run time to find a good packet forwarder.

Third, the probability of successful packet transmission is usually not uniform in a real traffic scene. We approximate it by its time average, that is, we assume that *1-hop packet error ratio* $\xi(x_1, x_2)$ varies over positions $x_i \in M$ and is constant over time. Steps 3-1-2-2 and 5 of *SteadyTraffic2* include related multiplications. We could use values of $\xi$ measured in field data possibly in conjunction with an analytical formula such as the one in [10].

## 3. Simulation Results

*3.1. Testbed Design.* We did a performance evaluation of the probabilistic protocol in a simulation testbed shown in Figure 4. Its dimension is 1000 m in height and 2000 m in

```
 Algorithm SteadyTraffic2
Inputs:
   $T, H, M, R, P(u_1, u_2), s, d \in M$: same as SteadyTraffic,
   $\xi(x_1, x_2)$ : 1 hop packet error ratio from $x_1 \in M$ to $x_2 \in M$,
   $e(x)$: expected car density at $x \in M$,
   $\beta, \epsilon \in (0, 1)$: numeric parameters,
   $dir \in \{(0, -1), (0, 1), (1, 0), (-1, 0)\}$ : vector representing the moving
      direction.
Output:
    Approximate communication probability $\rho(s, d, T)$
from $s$ to $d$ with time limit $T$.
begin
   (1). For every space time $u = (x, t)$ do Answer $(x, t, 0) = 0$;
   (2). Answer $(s, 0, 0) = 1$;
   (3). for $k = 1$ to $H$ do
      3-1. for every space time $u = (x, t)$ do
         3-1-1. Answer $(x, t, k) = P((s, 0), u)$;
         3-1-2. for every other space time $u_2 = (x_2, t_2)$ and $u_1 = (x_1, t_2)$
               such that
               i) $t_2 \leq t$, ii) $x_1 \in D(x_2)$ , iii) $P(u_1, u) \geq \epsilon$,
               iv) $(x_1 - x_2) \cdot dir \geq 0$ and iv) $(x - x_1) \cdot dir \geq 0$
               /* $D(x)$ is the set of $R$-neighbors of $x \in M$ */
               /* $\cdot$ represents the inner product of two vectors */
               do
            3-1-2-1. $p_1 = P(u_1, u) \ (e(x_1)e(x))^{\beta}$;
            3-1-2-2. $p_2 =$ Answer $(x_2, t_2, k - 1)(1 - \xi(x_1, x)) \ p_1$;
            3-1-2-3. Answer $(x, t, k) =$ Answer $(x, t, k) \ (1 - p_2) + p_2$;
         3-1-3. end for
      3-2. end for
      (4). end for
      (5). return $1 - \prod_{\substack{x \in D(d) \\ t \leq T}} (1 -$ Answer $(x, t, H)(1 - \xi(x, d)))$;
   end
```

Algorithm 2: A practical variant of *SteadyTraffic*.

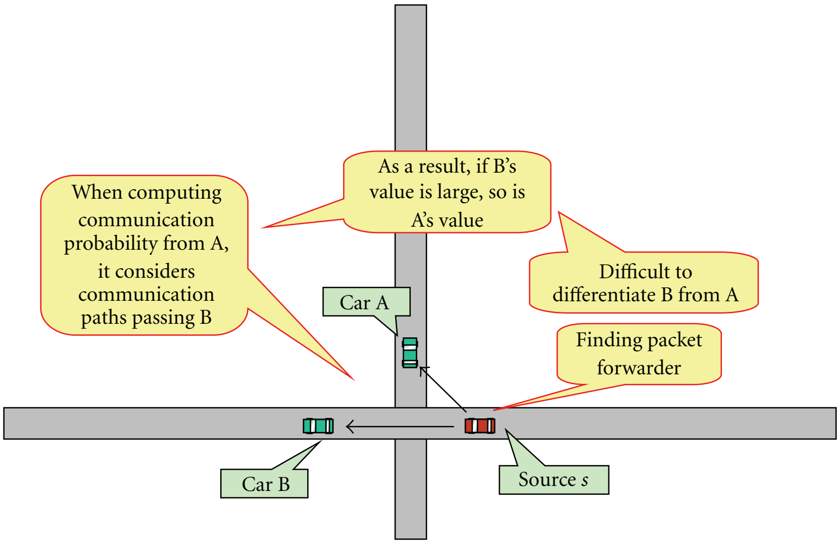

As a result, if B's value is large, so is A's value
When computing communication probability from A, it considers communication paths passing B
Difficult to differentiate B from A
Car A
Finding packet forwarder
Car B
Source $s$

Figure 3: The node differentiation problem.

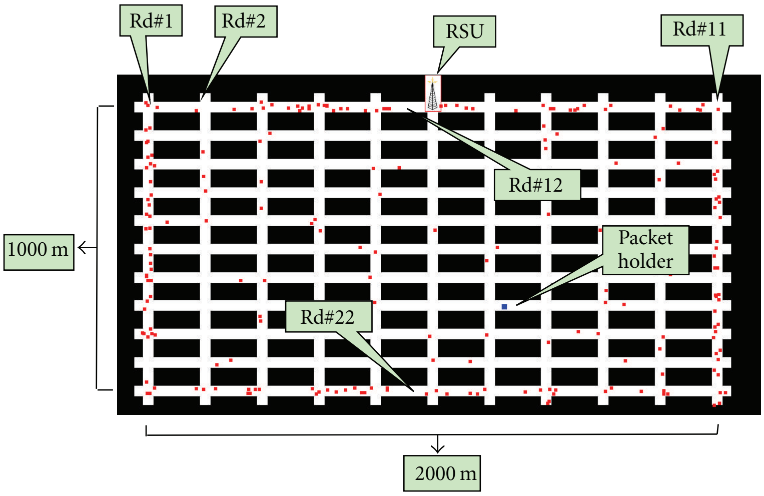


Figure 4: The Simulation Testbed.

width. A similar road configuration can be found in a big city in the United States such as the one in Chicago downtown. There are 121 intersections in total, the top-left and bottom-right ones of which are assumed to locate at the positions $(0, 0)$ and $(1000, 2000)$, respectively.

An RSU is placed at $(0, 1000)$. Roads 1, 11, 12, and 22 are called *arterial* and are busy, while the other *paths* are not. A car with a packet called *packet holder* enters the testbed at position $(1000, x)$ and time 0, where $x$ is either 0, 200, 400, 600, 800, or 1000. The cars keep moving in the next 70 seconds trying to forward the packet to the RSU.

Because the inner roads have sparser vehicles, and there is only one RSU in a fairly large area, it is not easy for the geographical forwarding scheme to have a packet reach the destination. The usual best way is to make a detour on arterial roads. It is essential to *quickly inform the cars of a good forwarding path computed from the likelihood of packet reachability*. However, the number of intersections is large so that there are many choices for a forwarding path. These make the testbed challenging for the existing routing protocols.

A car moves at the speeds of 35 and 25 mph on arterial roads and paths, respectively, except around intersections. Within a 50 m radius of every intersection, the car speed drops down to 15 mph. The following is the list of parameters input to the traffic simulator.

(1) $r_1$: the probability for a car to turn from an arterial road to another, or from a path to another.

(2) $r_2$: the probability for a car to turn from an arterial road to a path.

(3) $r_3$: the probability for a car to turn from a path to an arterial road.

(4) $r_4$: the probability that a car enters the testbed in each half second, at the start/end position of an arterial road.

(5) $r_5$: the probability that a car enters the testbed in each half second, at the start/end position of a path.

We tested two scenarios with different parameter sets shown in Table 1. Although the car movement rules are made relatively simple, both scenarios create traffic situations sufficiently realistic to a preliminary multihop packet forwarding test. Figure 4 is a snapshot of Scenario 1. Scenario 2 has sparser paths.

In this setting, we assume transmission range $R = 150\,\text{m}$ with one hop packet error ratio $\xi(x_1, x_2) = 0.15$ for every distinct pair of $x_i \in M$.

All the cars are equipped with radio devices sending their positions to the nearby cars per second as *heart beat (beacon) messages*. The relative velocity of two cars does not exceed 70 mph or 31.11 m/sec. With heartbeat messages broadcasted every second, where $R = 150\,m$, we can assume that each car $A$ knows by its background job the existence of every other car $A'$ in the proximity. Once $A$ receives a packet, it quickly computes the necessary values for $A'$ to decide the next packet holder.

It is a common agreement that the initialization time over IEEE 802.11 p standard is about 300 msec. The total time necessary to forward a packet is no more than 500 msec if it is not too large. Assume also that all the cars have a common clock so that packet transmission processes are performed in global half second time slots. The time resolution of the traffic simulator is set as 0.5 second.

Setting the maximum number $H$ of hops as 20, we run *SteadyTraffic2* to pre-compute the communication probabilities. We averaged the values in the north and west modes in the left half of the testbed and the ones in the north and east modes in the right half. It took about 3 hours on average to finish computation for each scenario. We stored the obtained values in a lookup table to refer to in run time.

*3.2. Decision for the Next Packet Holder.* Now, we have a good set of communication probabilities $\rho(x, x', T)$ from $x \in M$ to $x' \in M$ such that if $x$ or $x'$ is on a path, they are usually distinguishably smaller than values with both $x$ and $x'$ on arterial roads. However, it is still not obvious how to make a good decision for the next packet holder with $\rho(x, x', T)$.

Table 1: Parameter values to the traffic simulator.

| | $r_1$ | $r_2$ | $r_3$ | $r_4$ | $r_5$ |
|---|---|---|---|---|---|
| Scenario 1 | 0.4 | 0.1 | 0.94 | 0.3 | 0.01 |
| Scenario 2 | 0.4 | 0.06 | 0.97 | 0.3 | 0.01 |

We also pre-computed the *effective average numbers*

$$\bar{k}(x) = \frac{\sum_{\substack{0\le t\le T\\ 0<k\le H}} k(\text{Answer }(x,t,k) - \text{Answer }(x,t,k-1))}{\sum_{0\le t\le T}\text{Answer }(x,t,H)}, \tag{4}$$

of hops with Answer$(x,t,k)$ obtained when we run STEADY TRAFFIC. They turned out to be excellent index values for deciding a good packet forwarder.

Let $\Delta_H(x)$ be the horizontal distance from $x \in M$ to $d$, and $\Delta_V(x)$ the vertical distance from $x$ to $d$. Furthermore, define

$$\Delta(x) = \begin{cases} \Delta_H(x) & \text{if}\, \Delta_{\text{H}}(\text{s}) \ge \Delta_{\text{V}}(\text{s}), \\ \Delta_V(x) & \text{otherwise.} \end{cases} \tag{5}$$

It measures the distance to the destination from $x$ that is designed suitable for the probabilistic protocol.

In run time, decide the next packet holder as follows: if $\Delta(x)$ is much smaller than $\Delta(s)$, say at most 1/4, then we regard that the packet is approaching to the destination, so we choose, within the transmission range, a car *closest to $d$ among the ones with relatively high values of $\rho(x,d,T)$ and low $\bar{k}(x)$*. If $\Delta(x) > 1/4\Delta(s)$, we choose a car *farthest from $s$ with relatively high $\rho(x,d,T)$ and low $\bar{k}(x)$*, regarding that the packet is not close enough to the destination. We call the former and latter cases *approach* and *detour modes*, respectively.

More precisely, find if

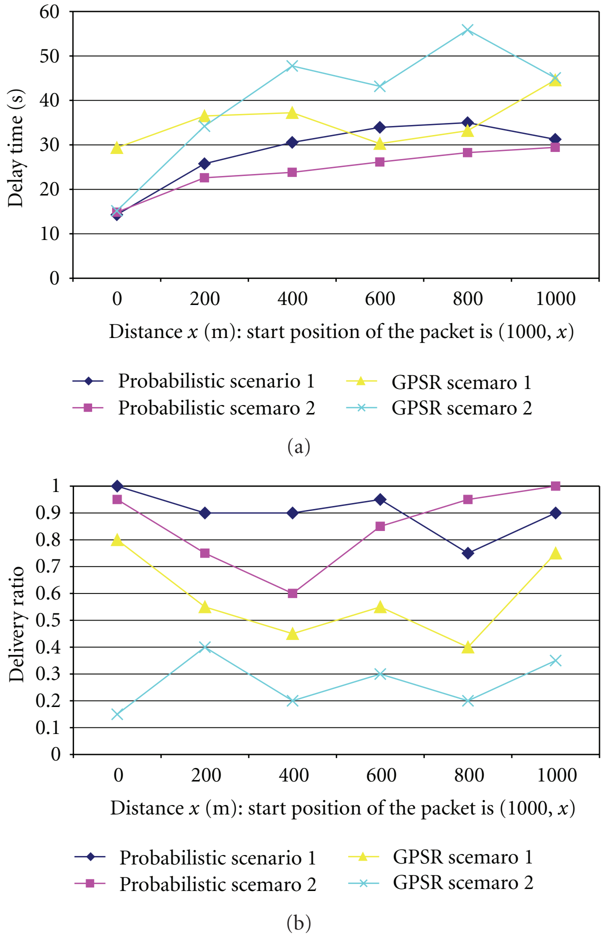
$$S = \left(\rho(x,d,T) \ge \max_{\rho}\alpha_{\rho}\right) \bigwedge \left(\bar{k}(x) \le \min_{\bar{k}}\alpha_{\bar{k}}\right), \tag{6}$$

,for every car at position $x$ in the proximity. Here, $\alpha_\rho$ and $\alpha_{\bar{k}}$ are input constants, $\max_\rho$ is the maximum value of $\rho(x,d,T)$ and $\min_{\bar{k}}$ the minimum value of $\bar{k}(x)$ in the proximity, respectively. Setting $\alpha_\rho = 0.4$ and $\alpha_{\bar{k}} = 1.05$ in the approach mode, pass the packet to the car closest to $d$ such that $S$ is true. In the detour mode with $\alpha_\rho = 0.8$ and $\alpha_{\bar{k}} = 1.05$, pass to the car farthest from $s$ such that $S$.

The above decision process is quick since it does not require heavy in-vehicle computation. The running time is comparable to that of GPSR.

*3.3. Measured Data.* We show the measured delivery ratio and delay time in Figure 5, plotting them over the horizontal position $x$ of the packet holder at time $t = 0$. We executed 20 trials for both Scenarios 1 and 2, each of which run for 70 seconds. The performances of the probabilistic protocol are compared with those of GPSR. We summarize the related parameter values in Tables 2 and 3.

As mentioned before, we run *SteadyTraffic2* in north and east modes for each source in the west half of the testbed. The east half is handled symmetrically. We took the average of the two modes to refer to in the run time.



(a)

(b)

Figure 5: Measured delivery ratio and delay time.

When the probabilistic protocol run in both scenarios, we observed expected packet trajectories in most cases. They initially run on the south arterial road (no.22) either to east or west, then go north to reach the RSU on the north arterial road. Our scheme of switching from the detour to approach mode functioned well.

This did not happen for GPSR; occasionally, the packet could not find a forwarder in the north effectively so that it migrated in the middle of the testbed for a long time.

As a result, the probabilistic protocol dramatically improves the delivery ratio over GPSR. Its average delay time over all start position $x$ is smaller than that of GPSR. The improvement, however, is not as drastic as that of the delivery ratio. It is due to the lengths of trajectories why it tends to

Table 2: Run-time parameter values for data measurement.

| | |
|---|---|
| Iteration count | 20 |
| Clock cycle | 0.5 sec |
| Trial time | 70 sec |
| Transmission range | 150 m |
| Numeric parameter $\alpha_\rho$ in the approch mode | 0.4 |
| Numeric parameter $\alpha_\rho$ in the detour mode | 0.8 |
| Numeric parameter $\alpha_{\overline{k}}$ | 1.05 |

Table 3: Parameter values for *steadytraffic2*.

| | |
|---|---|
| Time resolution | 5 sec |
| Space resolution | 100 m |
| Time limit $T$ | 60 sec |
| Maximum number $H$ of hops | 20 |
| Packet error ratio $\xi(x_1, x_2)$ | 0.15 uniform |
| Numeric parameter $\beta$ | 1/2 |

produce not too small latency, that is, packets move on a much longer way than those controlled by GPSR.

## 4. Conclusions

We have presented the notion of a communication path and probability in a general framework that handles the nature of multihop routing in a VANET. Under certain mathematical assumption, the communication probability is proven computed accurately by the algorithm *SteadyTraffic*. Use this quantity to decide a good packet forwarder in a real VANET. The simulated results show that the probabilistic protocol improves the performances much in a very challenging testbed, after the algorithm is refined for the practical issues.

For further research, it is significant to find a method to use *SteadyTraffic* or *SteadyTraffic2* in real-time computation. It is possible that a car is provided road density information by an extra method such as cellular network. It would be nice if one can find, quickly without precomputation, good paths over V2V multihop communications for a packet or a group of packets. The running time of the algorithms must be improved much.

In addition, it is theoretically interesting to seek for a way to efficiently compute the probability of logical sum $\vee_{((e \in S)\ e)}$ for an event set $S \subset \mathcal{F}$ that is not necessarily independent. The hardness of computing communication probabilities derives from the mutual dependence of communication paths. There may be a scheme for fast computation under an assumption of restricted dependence.

## Appendices

## A. Review for Independent Events in a Discrete Probability Space

A *probability space* is a triple $(\Omega, \mathcal{F}, P)$, where $\Omega$ is the *sample space*, $\mathcal{F}$ is the *event set*, and $P : \mathcal{F} \to [0,1]$ the *probability measure*. For the formulated problem of computing communication probabilities, we have

$\Omega$: the set of $(A, u)$, that is, the events that car $A$ appears at space time $u$,

$\mathcal{F}$: the set of all events over $\Omega$, that is, the set of Boolean expressions in terms of$(A,u)$, and

$P$: probability that each event $\mathcal{F}$ occurs.

This probability space $(\Omega, \mathcal{F}, P)$ is *discrete* because $\Omega$ is finite.

A nonempty event set $S \subseteq \mathcal{F}$ is said to be *independent* if

$$P\left(\bigwedge_{e \in S'} e\right) = \prod_{e \in S'} P(e), \tag{A.1}$$

for every subset $S'$ of $S$. Here, we regard that both hand sides are 1 if $S'$ is empty. We also say that the elements in $S$ are independent events. The *conditional probability of $e \in \mathcal{F}$ given $S$* is

$$P(es) = \begin{cases} \dfrac{P\left(e \wedge \bigwedge_{e' \in S} e'\right)}{P\left(\bigwedge_{e' \in S} e'\right)}, & \text{if } P\left(\bigwedge_{e' \in S} e'\right) \neq 0 \text{ and } S \neq \emptyset, \\ P(e), & \text{otherwise } 0. \end{cases} \tag{A.2}$$

If $S$ is independent,

$$P\left(\bigvee_{e \in S} e\right) = 1 - \prod_{e \in S} (1 - P(e)) \tag{A.3}$$

holds. We can derive it formally from the well-known inclusion-exclusion principle. Intuitively, it is true since the probability that any $e$ occurs is 1 minus probability $p$ that none of them occurs. But $p$ is the product of $1 - P(e)$ because the events $e \in S$ are independent. The equation implies

$$\begin{aligned} P\left(\bigvee_{e \in S} e\right) &= P\left(\bigvee_{e \in S - \{e_0\}} e\right) + P(e_0) \\ &\quad - P\left(\bigvee_{e \in S - \{e_0\}} e\right) \cdot P(e_0), \end{aligned} \tag{A.4}$$

for each $e_0 \in S$.

The event set $S$ is independent if and only if $P(e|S') = P(e)$ for every $e \in S$ and subset $S' \subseteq S$ that does not contain $e$. Suppose that $P(e|S') = P(e)$ is true for each $e$ and $S'$. By the definition of conditional probability $P(e \wedge \bigwedge_{e' \in S'} e') = P(e \mid S) \cdot P(\bigwedge_{e' \in S'} e') = P(e) \cdot P(\bigwedge_{e' \in S'} e')$. Thus, $P(\bigwedge_{e \in S'} e) = P(e) \cdot P(\bigwedge_{e' \in S' - \{e\}} e')$ . Argue recursively for $S - \{e\}$ to have $P(\bigwedge_{e \in S'} e) = \prod_{e \in S'} P(e)$. Since it is true for any subset $S' \subseteq S$, $S$ is independent only if part is shown similarly.

We say *$e$ is independent of $S$* if $P(e \mid S') = P(e)$ for each $S' \subseteq S$. By the above claim, a nonempty event set $S \subseteq \mathcal{F}$ is independent if and only if each $e \in S$ is independent of $S - \{e\}$.

An event $e \in \mathcal{F}$ such that $P(e)$ is 1 or 0 is independent of $\mathcal{F} - \{e\}$. It is intuitively clear that if the probability of $e$ is 1/0, $e$ either occurs or does not occur under any condition $S \subset \mathcal{F} - \{e\}$, so $P(e \mid S) = 1/0$. We can formally prove it by considering the disjunctive normal form of $e \wedge \bigvee_{e' \in S}(e' \vee \neg e')$, that is, the logical sum of $e \wedge f_1 \wedge f_2 \wedge \cdots \wedge f_k$, where $f_i$ are either $e_i$ or $\neg e_i$ for all $e_i \in S$.

## B. Correctness of Algorithm SteadyTraffic

**Theorem 1.** *Algorithm SteadyTraffic correctly computes the communication probability from s to d with time limit T, for every input instance satisfying the steady traffic assumption.*

*Proof.* We show that the final value of Answer $(x, t, k)$ is the probability that there exists a communication path from $u_0 = (s, 0)$ to $u = (x, t)$ with at most $k$ carries.

Prove it by induction on $k$. For its basis $k = 0$, we regard true that there exists a communication path from $u_0$ to $u_0$ with zero carries by probability 1. Step 3 sets Answer $(s, 0, 0)$ correctly.

To prove induction step, fix each space time $u = (x, t)$ selected by Step 4-1. Denote by $(u, k)$ the event that there exists a communication path from $u_0$ to $u$ with at most $k$ carries. The claim to be proven is rephrased as

$$\text{Answer } (x, t, k) = P(u, k) \text{ at step 4–2.} \tag{B.1}$$

It is straightforward to prove it for $k = 1$. Assume that $k \geq 2$.

Recall that $[u, k]$ expresses the event that $u = (x, t)$ is reachable from $u_0$ in $k$ hops, and $D(x)$ the set of $R$-neighbors of $x$. Let $u_1 = (x_1, t_1)$ be chosen by Step 4-1-2. Denote by $\widehat{D}(u)$ the set of space times $(x', t)$ such that $x' \in D(x)$. The symbols satisfy the following relations:

$$(u, k) = (u_0, u) \ \vee \bigvee_{\text{space times } u_1} [u_1, k - 1] \bigwedge (u_1, u), \tag{B.2}$$

$$[u_1, k - 1] = \bigvee_{u_2 \in \widehat{D}(u_1)} (u_2, k - 1). \tag{B.3}$$

There is only one communication path from $u_0$ to $u$ with 1 carry, that is , the carry $(u_0, u)$ itself. Every other communication path passes a space time $u_2 = (x_2, t_1)$ that is reachable from $u_0$ in $k - 1$ hops. The above relations are hence true. First, we prove the following general statement.

**Lemma 2.** *For each $k \leq H$, the events $(u, k)$ are independent.*

*Proof.* By induction on $k$ with basis true by the steady traffic assumption (i). Assume true for $k - 1$ and prove true for $k$. □

*Claim 1.* each $(u_1, u)$ in (B.2) is independent of $[u_1, k - 1]$ and the other $(u_1, u)$.

*Proof.* $[u_1, k - 1]$ is a product of carries that are not equal to $(u_1, u)$. The steady traffic assumption (i) implies the claim. □

*Claim 2.* the sets

$$S(u_1) = \left\{(u_2, k - 1); u_2 \in \widehat{D}u_1, P(u_2, k - 1) > 0\right\}, \tag{B.4}$$

for all the space times $u_1$ are pairwise disjoint.

*Proof.* Suppose that there are two $u_1$, say $u_1^1$ and $u_1^2$, such that $T = S(u_1^1) \cap S(u_1^2)$ is nonempty. The probability $P([u_1^i, k - 1])$ is the product of $P(u_2, \mathrm{k} - 1)$ for each $i = 1, 2$ and all $u_2 \in \widehat{D}(u_1^i)$, by (B.3) with induction hypothesis. Since $T$ is not empty, it means

$$P([u_1^1, k] \wedge [u_1^2, k]) \neq P([u_1^1, k]) \cdot P([u_1^2, k]). \tag{B.5}$$

The events $[u_1, k]$ are not independent, which contradicts the steady traffic assumption (ii). Thus, $S(u_1)$ must be pairwise disjoint. □

By the above two claims, the events $(u_0, u)$, $(u_1, u)$, and $[u_1, k - 1]$ in (B.2) are independent. By (A.3),

$$P(u, k) = 1 - (1 - P(u\_0, u)\ )P', \tag{B.6}$$

where

$$P' = \prod_{\text{space times } u_1} (1 - P([u_1, k - 1])P(u_1, u)). \tag{B.7}$$

*Claim 3.* let $S'(u)$ be the union of $S(u_1)$ such that $P(u_1, u) \neq 0$. Then, they are pairswise disjoint for all the space times $u$.

*Proof.* Similarly to Claim 2. If there are two $u$, say $u^1$ and $u^2$, such that $S'(u_1)$ and $S'(u_2)$, are not disjoint, the probability $P([u^1, k] \bigwedge [u^2, k])$ is not $P([u^1, k])P([u^2, k])$. □

We now see with (A.1) that each $(u, k)$ is independent of the others. Consider two distinct $u$, say $u^1$ and $u^2$. The carries $(u_0, u^1)$ and $(u_1, u^1)$ are distinct from $(u_0, u^2)$ and $(u_1, u^2)$, respectively. The events $[u_1, k - 1]$ are independent by Claim 3 and induction hypothesis. They imply that $(u, k)$ is independent of the others.

The lemma follows the above, completing the proof.

We verify below that Loop 4-1 correctly computes $P(u, k)$ in Answer$(x, t, k)$. By the lemma, (B.6) holds throughout the process. Let $U_1$ be the set of $u_1$ chosen so far by Step 4-1-2. We prove by induction on its size that Answer $(x, t, k)$ currently holds the value of (B.6) for

$$P' = \prod_{u_1 \in U_1} (1 - P([u_1, k - 1])P(u_1, u)). \tag{B.8}$$

For its basis, the set $U_1$ is regarded as empty before Step 4-1-2 so that $P'$ in is 1 and $P(u, k) = P(u_0, u)$. Step 4-1-1 correctly sets Answer$(x, t, k)$ for it.

To prove its induction step, assume Answer $(x, t, k)$ has the value of (A.1) for $P' = \prod_{u'{}_1 \in U_1 - \{u_1\}}(1 - P([u'{}_1, k - 1])P(u'{}_1, u))$, where $u_1$ is currently chosen by 4-1-2. Due to (6), where $(u_2, k - 1)$ are independent, Steps 4-1-2-2 and 4-1-2-3 compute $P([u_1, k - 1])$ and $P([u_1, k - 1])P(u_1, u)$ as $p_1$ and $p_2$, respectively.

Regard $[u_1, k-1] \bigwedge (u_1, u)$ as $e_0$ in (4), and $[u'_1, k-1] \bigwedge (u'_1, u)$ and $(u_0, u)$ as the other events in $S$. Step 4-1-2-4 correctly computes $P(u, k)$ in (B.6) for $U_1$ as Answer $(x, t, k)$. This proves the induction step.

We have shown that Answer $(x, t, k) = P(u, k)$ at Step 4-2. Finally, Step 6 computes the probability that the space time $(d, T)$ is reachable from $u_0 = (s, 0)$ in $H$ hops, returning $\rho(s, d, T)$ correctly. This completes the proof of the theorem.

## References

[1] I. Leontiadis and C. Mascolo, "GeOpps: geographical opportunistic routing for vehicular networks," in *Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WOWMOM '07)*, June 2007.

[2] V. Naumov and T. R. Gross, "Connectivity-aware routing (CAR) in vehicular ad hoc networks," in *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM '07)*, pp. 1919–1927, 2007.

[3] A. Skordylis and N. Trigoni, "Delay-bounded routing in vehicular ad-hoc networks," in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '08)*, pp. 341–350, May 2008.

[4] J. Zhao and G. Cao, "VADD: vehicle-assisted data delivery in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 3, pp. 1910–1922, 2008.

[5] J. Fukuyama, "A probabilistic protocol for multi-hop routing in VANETs," in *Proceedings of IEEE International Conference on Communications Workshops (ICC '09)*, 2009.

[6] D. Kumar, A. A. Kherani, and E. Altman, "Route lifetime based optimal hop selection in VANETs on highway: an analytical viewpoint," in *Proceedings of the 5th International IFIP-TC6 Networking Conference*, vol. 3976 of *Lecture Notes in Computer Science*, pp. 799–814, May 2006.

[7] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for wireless networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 243–254, August 2000.

[8] R. Baldessari, A. Festag, A. Matos, J. Santos, and R. Aguiar, "Flexible connectivity management in vehicular communication networks," in *Proceedings of 3rd International Workshop on Intelligent Transportation (WIT '06)*, pp. 211–216, 2006.

[9] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter, "MDDV: a mobility-centric data dissemination algorithm for vehicular networks," in *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks (VANET '04)*, pp. 47–56, October 2004.

[10] M. Killat, F. Schmidt-Eisenlohr, H. Hartenstein et al., "Enabling efficient and accurate large-scale simulations of VANETs for vehicular traffic management," in *Proceedings of the 4th ACM International Workshop on Vehicular Ad Hoc Networks (VANET '07)*, pp. 29–38, September 2007.

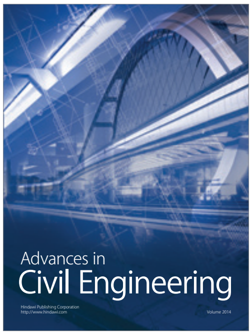
Journal of
Engineering

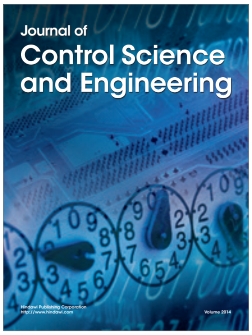
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

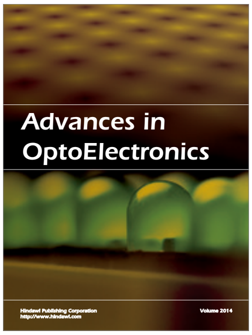
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

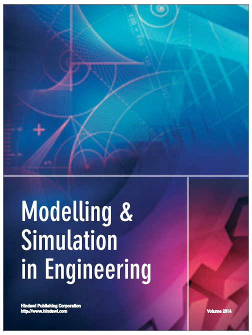
Journal of
Robotics

Journal of
Electrical and Computer Engineering

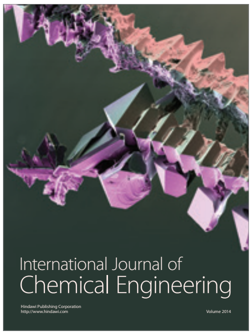

Advances in
OptoElectronics

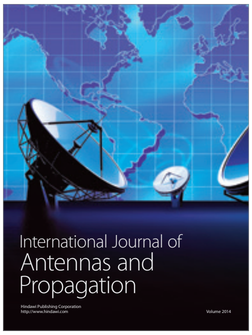
VLSI Design

International Journal of
Navigation and Observation

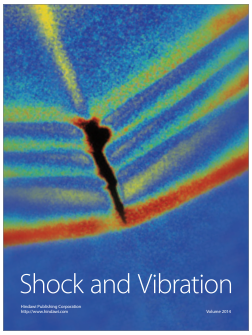

Modelling &
Simulation
in Engineering

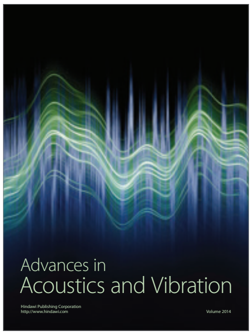
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration