Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks

Thrasyvoulos Spyropoulos, Konstantinos Psounis, Cauligi S. Raghavendra
(All from the University of Southern California)

SIGCOMM-2005, Philadelphia

Presented by: Harshal Pandya

On 10/31/2006 for CS 577 - Advanced Computer Networks
Abstract

- **Intermittently Connected Mobile Networks (ICMN)** are sparse wireless networks where most of the time there does not exist a complete path from the source to the destination. It can be viewed as a set of disconnected, time-varying clusters of nodes.

- These fall into the general category of **Delay Tolerant Networks**, where incurred delays can be very large and unpredictable.

- Some networks that follow this paradigm are:
  - Wildlife tracking sensor networks
  - Military networks
  - Inter-planetary networks

- In such networks conventional routing schemes such as DSR & AODV would fail.
An example of Intermittently Connected Mobile Networks (ICMN)

- S is the source & D is the Destination
- There is no direct path from S to D
- In this case all the conventional protocols would fail
- Thus, the authors introduce a new routing scheme, called **Spray and Wait**, that “sprays” a number of copies into the network, and then “waits” till one of these nodes meets the destination.
Basic Idea behind Spray & Wait

- In such networks, the traditional protocols will fail to discover a complete path or will fail to converge, resulting in a deluge of topology update messages.

- However, this does not mean that packets can never be delivered in such networks.

- Over time, different links come up and down due to node mobility. If the sequence of connectivity graphs over a time interval are overlapped, then an end-to-end path might exist.

- This implies that a message could be sent over an existing link, get buffered at the next hop until the next link in the path comes up, and so on, until it reaches its destination.

- This approach imposes a new model for routing. Routing consists of a sequence of independent, local forwarding decisions, based on current connectivity information and predictions of future connectivity information.

- In other words, node mobility needs to be exploited in order to overcome the lack of end-to-end connectivity and deliver a message to its destination.
A possible solution
Advantages of Spray & Wait

- Under **low** load, Spray and Wait results in **much fewer** transmissions and comparable or smaller delays than flooding-based schemes.

- Under **high** load, it yields **significantly better** delays and fewer transmissions than flooding-based schemes.

- It is **highly scalable**, exhibiting good and predictable performance for a large range of network sizes, node densities and connectivity levels. As the size of the network and the number of nodes increase, the number of transmissions *per node* that Spray and Wait requires in order to achieve the same performance, decreases.

- It can be easily **tuned online** to achieve given QoS requirements, even in unknown networks.

- Using only a **handful of copies per message**, it can achieve comparable delays to an oracle-based optimal scheme that minimizes delay while using the lowest possible number of transmissions.
Related Work

• A large number of routing protocols for wireless ad-hoc networks have been proposed in the past. The performance of such protocols would be poor even if the network was only slightly disconnected.

• When the network is not dense enough (as in the ICMN case), even moderate node mobility would lead to frequent disconnections.

• In most cases $t_{\text{repair}}$ is expected to be larger than the path duration, this way reducing the expected throughput to almost zero according to the formula:

$$\text{throughput} = \min\{0, rate \left(1 - \frac{t_{\text{repair}}}{PD}\right)\}$$

PD = average Path Duration

• Another approach to deal with disconnections is to reinforce connectivity on demand, by bringing additional communication resources into the network when necessary (e.g. satellites, UAVs, etc.)
Related Work

• Similarly, one could force a number of specialized nodes (e.g. robots) to follow a given trajectory between disconnected parts of the network in order to bridge the gap.

• There, a number of algorithms with increasing knowledge about network characteristics like upcoming contacts, queue sizes, etc. is compared with an optimal centralized solution of the problem, formulated as a linear program.

• A number of mobile nodes performing independent random walks serve as DataMules that collect data from static sensors and deliver them to base stations.

• In a number of other works, all nodes are assumed to be mobile and algorithms to transfer messages from any node to any other node, are sought for.

• *Epidemic Routing:*
  - Here, each node maintains a list of all messages it carries, whose delivery is pending. Whenever it encounters another node, the two nodes exchange all messages that they don’t have in common. This way, all messages are eventually spread to all nodes, including their destination.
  - But it creates a lot of contention for the limited buffer space and network capacity of typical wireless networks, resulting in many message drops and retransmissions.
Related Work

• **Randomized Flooding:**
  – One simple approach to reduce the overhead of flooding and improve its performance is to only **forward a copy with some probability** \( p < 1 \)

• **History-based or Utility-based Routing:**
  – Here, each node maintains a utility value for every other node in the network, based on a timer indicating the time elapsed since the two nodes last encountered each other. These utility values essentially carry indirect information about relative node locations, which get diffused through nodes’ mobility.
  – Nodes forward message copies only to nodes with a higher utility by some pre-specified threshold value \( U_{th} \) for the message’s destination. Such a scheme results in superior performance than flooding.
  – But these schemes face a dilemma when choosing the utility threshold.

• **Oracle-based algorithm:**
  – This algorithm is aware of all future movements, and computes the **optimal set of forwarding decisions** (i.e. time and next hop), which delivers a message to its destination in the minimum amount of time.
  – This algorithm cannot be implemented practically, but is quite useful to compare against proposed practical schemes.
What should we expect from Spray & Wait?

• Perform significantly *fewer transmissions* than epidemic and other flooding-based routing schemes, under all conditions

• Generate *low contention*, especially under high traffic loads

• Achieve a delivery *delay* that is better than existing single and multi-copy schemes, and *close to the optimal*

• Be *highly scalable*, that is, maintain the above performance behavior despite changes in network size or node density

• Be *simple* and require as little knowledge about the network as possible, in order to facilitate implementation

• *Decouple* the number of copies generated per message, and therefore the number of transmissions performed, from the network size

• Thus it should be a *tradeoff* between single and multi-copy schemes.
Definition

• Definition 3.1: Spray and Wait routing consists of the following two phases:

  – **Spray phase:** For every message originating at a source node, L message copies are initially spread – forwarded by the source and possibly other nodes receiving a copy – to L distinct “relays”

  – **Wait phase:** If the destination is not found in the spraying phase, each of the L nodes carrying a message copy performs direct transmission (i.e. will forward the message only to its destination)

• This does not tell us how the L copies of a message are to be spread initially. So an improvement over Spray & Wait is Binary Spray & Wait
Binary Spray & Wait

• Binary Spray & Wait:
  – The source of a message initially starts with L copies; any node A that has n > 1 message copies, and encounters another node B with no copies, hands over to B, n/2 and keeps n/2 for itself; when it is left with only one copy, it switches to direct transmission

  – To prove that Binary Spray and Wait is optimal, when node movement is IID, the authors state and prove a theorem

• Theorem 3.1: When all nodes move in an IID manner, Binary Spray and Wait routing is optimal, that is, has the minimum expected delay among all spray and wait routing algorithms

  – Proof: Let us call a node active when it has more than one copies of a message. Let us further define a spraying algorithm in terms of a function f : N → N as follows

  – When an active node with n copies encounters another node, it hands over to it f(n) copies, and keeps the remaining 1 − f(n)
Binary Spray & Wait

- Any spraying algorithm (i.e. any f) can be represented by the following binary tree with the source as its root: Assign the root a value of L; if the current node has a value n > 1 create a right child with a value of 1−f(n) and a left one with a value of f(n); continue until all leaf nodes have a value of 1.

- A particular spraying corresponds then to a sequence of visiting all nodes of the tree. This sequence is random. On the average, all tree nodes at the same level are visited in parallel.

- Further, since only active nodes may hand over additional copies, the higher the number of active nodes when i copies are spread, the smaller the residual expected delay.

- Since the total number of tree nodes is fixed \((2^{1+\log L} - 1)\) for any spraying function f, it is easy to see that the tree structure that has the maximum number of nodes at every level, also has the maximum number of active nodes at every step.

- This tree is the balanced tree, and corresponds to the Binary Spray and Wait routing scheme. As L grows larger, the sophistication of the spraying heuristic has an increasing impact on the delivery delay of the spray and wait scheme.
Binary Spray & Wait

- Figure compares the expected delay of Binary Spray & Wait and Source Spray & Wait as a function of the number of copies L used, in a 100×100 network with 100 nodes.
- This figure also shows the delay of the Optimal scheme.

Figure 1: Comparison of Source Spray and Wait, Binary Spray and Wait, and Optimal schemes (100×100 network with 100 nodes).
Spray & Wait

• In many situations the network designer or the application itself might impose certain performance requirements on the protocols (e.g. maximum delay, maximum energy consumption, minimum throughput, etc.).

• Spray and Wait can be tuned to achieve the desired performance.

• To do so the authors summarize a few results regarding the expected delay of the Direct Transmission and Optimal schemes.

• Lemma 4.1: Let M nodes with transmission range K perform independent random walks on a $\sqrt{N} \times \sqrt{N}$ torus.

  - The expected delay of Direct Transmission is exponentially distributed with average

\[
ED_{dt} = 0.5N \left( 0.34 \log N - \frac{2^{K+1} - K - 2}{2^K - 1} \right)
\]
The expected delay of the Optimal algorithm is:

\[ ED_{opt} = \frac{H_{M-1}}{(M - 1)} E D_{dt} \]

where \( H_n \) is the \( n^{th} \) Harmonic Number, i.e., \( H_n = \sum_{i=1}^{n} \frac{1}{i} = \Theta(\log n) \).

**Lemma 4.2:** The expected delay of Spray and Wait, when \( L \) message copies are used, is upper-bounded by:

\[ ED_{sw} \leq (H_{M-1} - H_{M-L}) E D_{dt} + \frac{M - L}{M - 1} \frac{E D_{dt}}{L} \]

This bound is tight when \( L \ll M \).
Choosing L to Achieve a Required Expected Delay

• Assume that there is a specific delivery delay constraint to be met. This delay constraint is expressed as a factor $a$ times the optimal delay $ED_{opt}$ ($a > 1$)

• **Lemma 4.3:** The minimum number of copies $L_{\text{min}}$ needed for Spray and Wait to achieve an expected delay at most $a*ED_{opt}$, is independent of the size of the network $N$ and transmission range $K$, and only depends on $a$ and the number of nodes $M$

• Thus we get the following equation from the previous upper bounded equation of $ED_{sw}$

$$
(H_M^3 - 1.2)L^3 + (H_M^2 - \frac{\pi^2}{6})L^2 + \left(a + \frac{2M - 1}{M(M - 1)}\right)L = \frac{M}{M - 1}
$$

Note: $ED_{sw} = a*ED_{opt}$ and approximate the harmonic number $H_{M-L}$ with its Taylor Series terms up to second order, and solve the resulting third degree polynomial
Comparing Various L

Table 1: minimum $L$ to achieve expected delay

<table>
<thead>
<tr>
<th>a</th>
<th>1.5</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>exact</td>
<td>21</td>
<td>13</td>
<td>8</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>bound</td>
<td>N.A.</td>
<td>N.A.</td>
<td>11</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>taylor</td>
<td>N.A.</td>
<td>N.A.</td>
<td>10</td>
<td>7</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

- $L$ found through the approximation is quite accurate when the delay constraint is not too stringent.
Estimating $L$ when Network Parameters are Unknown

- In many cases both $M$ and $N$, might be unknown

- But for determining $L$, at least $M$ is required

- Hence we have to somehow estimate $M$ to find out $L$

- A straightforward way to estimate $M$ would be to count unique IDs of nodes encountered already. This method requires a large database of node IDs to be maintained in large networks, and a lookup operation to be performed every time any node is encountered

- A better method: We define $T_1$ as the time until a node encounters any other node. $T_1$ is exponentially distributed with average $T_1 = \frac{ED_{dt}}{M - 1}$

- We similarly define $T_2$ as the time until two different nodes are encountered, then the expected value of $T_2$ equals $ED_{dt} \left( \frac{1}{M-1} + \frac{1}{M-2} \right)$

- Canceling $ED_{dt}$ from these two equations we get the following estimate for $M$:

$$\hat{M} = \frac{2T_2 - 3T_1}{T_2 - 2T_1}$$
A better estimate of $T_1$ & $T_2$

- When a random walk $i$ meets another random walk $j$, $i$ and $j$ become *coupled*.

- In other words, the next inter meeting time of $i$ and $j$ is not anymore exponentially distributed with average $ED_{dt}$.

- In order to overcome this problem, each node keeps a record of all nodes with which it is coupled.

- Every time a new node is encountered, it is stamped as coupled for an amount of time equal to the mixing or relaxation time for that graph.

- Then, when node $I$ measures the next sample inter meeting time, it ignores all nodes that it’s coupled with at the moment, denoted as $c_k$, and scales the collected sample $T_{1,k}$ by $(M-c_k)/(M-1)$.

- A similar procedure is followed for $T_2$. Putting it altogether, after $n$ samples have been collected:

$$\hat{T}_1 = \frac{1}{n} \sum_{k=1}^{n} \left( \frac{M - c_k}{M - 1} \right) T_{1,k}$$

$$\hat{T}_2 = \frac{1}{n} \sum_{k=1}^{n} \left[ \left( \frac{M - c_{k-1}}{M - 1} \right) T_{1,k-1} + \left( \frac{M - c_k}{M - 2} \right) T_{1,k} \right]$$
Comparing the online estimators of $M$

**Abstract**

**Introduction**

**Related Work**

**Spray & Wait**

**Optimization**

**Simulation**

**Conclusion**

Estimated value of $M$ quickly converges with the actual value.

**Figure 2:** Online estimator of number of nodes ($M$) — 200 x 200 grid, transmission range = 0, $a = 0.98$, mixing time = 4000.

- This method is more than two times faster than ID counting
Scalability of Spray and Wait

- As the number of nodes in the network increases, the percentage of nodes \( \frac{L_{\text{min}}}{M} \) that need to become relays in Spray and Wait to achieve the same performance relative to the optimal, actually decreases.

![Graph showing percentage of nodes receiving a copy vs. number of nodes](image)

**Figure 3:** Required percentage of nodes \( \frac{L_{\text{min}}}{M} \) receiving a copy for spray and wait to achieve an expected delay of \( aE D_{\text{opt}} \).

- Also, the performance of Spray & Wait improves faster than optimal scheme !!! This can be proved using Lemma 4.4.
Spray and Wait actually decreases the transmissions per node as the number of nodes $M$ increases.

- **Lemma 4.4:** Let $L/M$ be constant and let $L << M$. Let further $L_{\text{min}}(M)$ denote the minimum number of copies needed by Spray and Wait to achieve an expected delay that is at most $a ED_{\text{opt}}$, for some $a$. Then $L_{\text{min}}(M)/M$ is a decreasing function of $M$.

  **Proof.** When $L \ll M$ we can use the upper bound of Eq. (1) to examine the behavior of Spray and Wait:

  \[
  ED_{sw} \leq ED_{dt} (H_{M-1} - H_{M-L}) + \left( \frac{M-L}{M-1} \right) \frac{ED_{dt}}{L}.
  \]

  Since $H_n = \Theta(\log(n))$, $H_{M-1} - H_{M-L} = \Theta(\log(\frac{M-1}{M-L}))$. Also, let $L = cM$, where $c$ is a constant ($c \ll 1$). Replacing $L$ in the previous equation gives us

  \[
  \Theta(\log(1 - 1/M) - \log(1 - c)) ED_{dt} + \left( \frac{M}{M-1} \right) \left( \frac{1-c}{c} \right) \frac{ED_{dt}}{M}.
  \]

  Now, for large $M$, $\frac{M-1}{M} \approx 1$. Therefore, keeping the size of the grid $N$ and transmission range $K$ constant we get that $ED_{sw} = \Theta(1) + \Theta(1) \frac{1}{M} = \Theta(\frac{1}{M})$.

  On the other hand, for constant $N$ and $K$, $ED_{opt} = \Theta(\frac{\log(M)}{M})$ as can be easily seen from Eq. (1). Hence, $\frac{ED_{sw}}{ED_{opt}} = \Theta\left( \frac{1}{\log(M)} \right)$ (i.e. decreasing with $M$), if $L/M$ is kept constant. This implies that if we require $\frac{ED_{sw}}{ED_{opt}}$ to be kept constant for increasing $M$, then $L/M$ has to be decreasing. $\blacksquare$
Scenario A : Effect of Traffic Load

• Assumptions:
  – 100 nodes move according to the random waypoint model in a 500 × 500 grid with reflective barriers.
  – The transmission range K of each node is equal to 10
  – Each node is generating a new message for a randomly selected destination with an inter-arrival time distribution uniform in [1, T_{max}] until time 10000
  – T_{max} is varied from 10000 to 2000 creating average traffic loads from 200 (low traffic) to 1000 (high traffic).

Abstract

Introduction
Related Work
Spray & Wait
Optimization
SIMULATION
Conclusion

Figure 4: Scenario A - performance comparison of all routing protocols under varying traffic loads.
Scenario B : Effect of Connectivity

- The size of the network is 200×200 and $T_{\text{max}}$ is fixed to 4000 (medium traffic load). The number of nodes $M$ and transmission range $K$, are varied to evaluate the performance of all protocols in networks with a large range of connectivity characteristics, ranging from very sparse, highly disconnected networks, to almost connected networks.

![Graph showing percentage of nodes in max cluster](image)

**Figure 5:** Expected percentage of total nodes in largest connected component, as a function of $M$ and $K$ (200 × 200 grid).
Scenario B: Number of transmissions for various transmission ranges for 100 & 200 nodes

For both networks, the number of transmissions for spray and wait are very less in number as compared to other schemes & are also more or less independent from the transmission range K.

Figure 6: Scenario B - Transmissions as a function of number of nodes M and transmission range K.
Scenario B: Delivery delay for various transmission ranges for 100 & 200 nodes

The delivery delay of Spray and Wait is significantly better than that of other schemes & depends on the transmission range. As the transmission range increases the delay decreases.

Figure 7: Scenario B - Delivery delay as a function of number of nodes $M$ and transmission range $K$. 

The delivery delay of Spray and Wait is significantly better than that of other schemes & depends on the transmission range. As the transmission range increases the delay decreases.
Conclusion

• Spray and Wait effectively manages to overcome the shortcomings of epidemic routing and other flooding-based schemes, and avoids the performance dilemma inherent in utility-based schemes.

• Spray and Wait, despite its simplicity, outperforms all existing schemes with respect to number of transmissions and delivery delays, achieves comparable delays to an optimal scheme, and is very scalable as the size of the network or connectivity level increase.
Some issues not discussed

- Power consumed
- Security
- Constrained Mobility of Nodes
Acknowledgements

- The slide design has been adopted from the presentation by Mike Putnam (Because I liked it a lot)

- Some figures have been adopted from the presentation by the authors

- All other figures have been taken from the actual paper
Thank you

Questions ?