Spray and Focus: Efficient Mobility-Assisted Routing for Heterogeneous and Correlated Mobility

Thrasyvoulos Spyropoulos
Planete Project
INRIA, Sophia-Antipolis
Thrasyvoulos.Spyropoulos@sophia.inria.fr

Konstantinos Psounis and Cauligi S. Raghavendra
Electrical Engineering and Computer Science
University of Southern California
{kpsounis,raghu}@usc.edu

Abstract

Intermittently connected mobile networks are wireless networks where most of the time there does not exist a complete path from the source to the destination. There are many real networks that follow this model, for example, wildlife tracking sensor networks, military networks, vehicular ad hoc networks (VANETs), etc. To deal with such networks researchers have suggested to use controlled replication or "spraying" methods that can reduce the overhead of flooding-based schemes by distributing a small number of copies to only a few relays. These relays then "look" for the destination in parallel as they move into the network. Although such schemes can perform well in scenarios with high mobility (e.g. VANETs), they struggle in situations were mobility is slow and correlated in space and/or time.

To route messages efficiently in such networks, we propose a scheme that also distributes a small number of copies to few relays. However, each relay can then forward its copy further using a single-copy utility-based scheme, instead of naively waiting to deliver it to the destination itself. This scheme exploits all the advantages of controlled replication, but is also able to identify appropriate forwarding opportunities that could deliver the message faster. Simulation results for traditional mobility models, as well as for a more realistic "community-based" model, indicate that our scheme can reduce the delay of existing spraying techniques up to 20 *times in some scenarios.*

1 Introduction

Traditionally, wireless ad hoc networks have been viewed as a connected graph over which end-to-end paths need to be established. Although this model has been quite successful in the wired world (telephone network, Internet, etc.), its shortcomings for the wireless environment have recently started to be recognized. Wireless propagation phenomena, power requirements [8], and a number of other operational or economic factors [2] indicate that wireless links may be short-lived and end-to-end connectivity more often than not intermittent.

Under such intermittent connectivity, conventional Internet routing protocols (e.g. RIP and OSPF) as well as mobile ad-hoc network routing schemes (e.g. DSR, AODV) fail. These try to establish complete end-to-end paths *before any useful data is sent*. If no such end-to-end paths exist most of the time, these protocols fail to deliver any data to all but the few connected nodes. However, this does not mean that packets can never be delivered in these 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-over-time* 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 and so forth, until it reaches its destination. This model of routing, often referred to as "mobility-assisted" routing, constitutes a significant departure from existing routing practices.

One approach that has shown good potential in this context is that of *controlled replication* or *spraying* [16, 17]. There, a small, fixed number of copies are generated and distributed ("sprayed") to different relays, each of which then carries its copy until it encounters the destination. By routing multiple copies independently, these protocols create enough diversity to "explore" the network efficiently, while keeping resource usage per message low. However, for this scheme to achieve good performance there are two key requirements: (a) to create enough diversity between the copy-bearing relays that will look for the destination in parallel, and (b) that each relay moves relatively quickly and frequently around the network, in order to carry a message through disconnected parts.

Although such desirable mobility characteristics could, to an extend, be found in some applications like, for example, VANETs [14], in many other scenarios, especially those involving human interactions, mobility is strongly corre-

Proceedings of the Fifth Annual IEEE International Conference
on Pervasive Computing and Communications Workshops(PerComW'07)
0-7695-2788-4/07 $20.00 © 2007 IEEE

lated in both time and space [3, 9, 13]. In such situations, the performance of simple spraying schemes can suffer, as they don't take advantage of existing transmission opportunities that could potentially forward the message closer to the destination over a partial path, or to a node more "socially" related to the destination [9, 13]. For this reason, we propose a novel protocol, called Spray and Focus, that overcomes the shortcomings of simple spraying algorithms, and outperforms both existing flooding-based schemes as well as existing spraying algorithms by up to $20\times$, under realistic mobility scenarios.

In the next section we describes our proposed solution, Spray and Focus, in detail. Then, in section 3 we use simulations to compare the performance of Spray and Focus against a number of existing schemes, as well as, simple spraying algorithms. Finally, we conclude this work in Section 4.

2 Spray and Focus

Existing spraying schemes [16, 17], generate and distribute ("spray") a small, fixed number of copies or "forwarding tokens" to a number of distinct relays. Then, each relay carries its copy until it encounters the destination or until the TTL (time-to-live) for the packet expires. By having multiple relays looking independently and in parallel for the destination, these protocols create enough diversity to explore the sparse connectivity graph more efficiently, and can discover a short *path-over-time* to the destination.

Although such schemes have been shown to perform well in some scenarios [16, 17], they require a high amount of mobility by network nodes to achieve this performance (as we will also show in Section 3). However, in many practical situations, the mobility of each node is limited to a small local area for the majority of time. An example where such local mobility might arise could be, for example, that of a university campus, where most people tend to stay or move locally within their buildings for long stretches of time [11]. To make our point more clear, consider for example the "Spray and Wait" scheme [16, 17] in such a scenario. This scheme consists of two phases: in the first phase it distributes a fixed number of copies to the first few relays encountered, and in the second phase each of these relays waits until it encounters the destination itself (i.e. "Direct Transmission" [20]). It is easy to see that, here, this scheme would spread all its copies quickly to the node's immediate neighborhood, but then few if any of the nodes carrying a copy might ever see the destination [3]. What is more, if the network is not too sparse, there might exist partial paths over which a message copy could be transmitted fast to a node closer to the destination. Yet, in schemes like Spray and Wait a relay with a copy will naively wait until it moves within range of the destination itself.

This problem could be solved if a sophisticated *single-copy* scheme is used to *further* route a copy after it's handed over to a relay, a scheme that takes advantage of transmissions (unlike Direct Transmission). With this in mind, we propose *Spray and Focus*, which in the second phase ("focus" phase) rather than waiting for the destination to be encountered, each relay can forward its copy to a potentially more appropriate relay, using a carefully designed utility-based scheme. In the next few sections, we describe our protocol in detail.

2.1 Spraying Phase

When a new message gets generated at a source, and needs to be routed to a given destination, Spray and Focus first enters the "Spray phase" for this message:

Definition 2.1 (Spray phase) *When a new message is generated at a source node it also creates L "forwarding tokens"* for this message. *A forwarding token implies that the node that owns it, can spawn and forward an additional copy of the given message, according to the following rules:*

- *each node maintains a "summary vector" with IDs of all messages that it has stored, and for which it acts as a relay; whenever two nodes encounter each other, they exchange their vectors and check which messages they have in common (as in epidemic routing [21]);*
- *if a node (either the source or a relay) carrying a message copy and $n > 1$ forwarding tokens encounters a node with no copy of the message, it spawns and forwards a copy of that message to the 2^{nd} node; it also hands over $\lfloor n/2 \rfloor$ forwarding tokens and keeps $\lceil n/2 \rceil$ for itself; (*Binary Spraying [17]*)*
- *when a node has a message copy but only one forwarding token for this message, then it can only forward this message further according to the rules of the "Focus phase" (more about this later).*

We next clarify a few practical issues and some of our choices.

Node encounters: First, when we refer to a "node encounter", we assume that nodes periodically transmit beacons to recognize each other's presence. We expect that the period of this beacon would have some effect on the performance of our protocol (if beacons are not sent often enough, some forwarding opportunities might be missed [3]). However, we assume that this is an issue that is handled by the underlying media access (MAC) protocol [5], and that, ideally, nodes "encounter" each other as soon as they come within communication range. Furthermore, there is an overhead involved in the exchange of the message summary. However, each message's ID is expected to occupy only a few bytes (e.g. source ID, destination ID, and sequence number), and buffer size limitations for relayed messages are expected to keep the number of messages queued relatively low.

Spraying mechanism: Another interesting question is how the number of copies (or forwarding tokens) should be distributed to different relays. In a homogeneous environment (i.e. IID node movement) it is beneficial to spread messages as quickly as possible, as all nodes are *statistically equivalent* [1]. What matters mainly there is how many relays are looking in parallel. It is proven in [17] that, if node movement is IID and nodes forward copies only to new nodes, the algorithm that minimizes spraying time is *Binary Spraying* [17].

In a heterogeneous environment, things are not that simple. Some nodes may be "better" relays for a given destination. Such, for example, could be nodes that tend to see the destination more often (e.g. work in the same building, or in general belong to the same social network [9, 13]). Ideally, we would like to be able to choose as relays the L nodes that most frequently encounter the destination. In an *offline* version of the problem, we could potentially formulate a *stochastic* version of the linear program described in [6] and solve for the minimum expected delivery time. However, in the online version of the problem, waiting for a "better" relay incurs a cost, because it means that opportunities to spread extra copies are forfeited. In other words, there are two conflicting strategies that can potentially reduce delay when a new node is encountered: (i) spawn and forward an extra copy right-away to increase parallelism, and (ii) defer forwarding until a relay that is more "correlated" with the destination is encountered.

To the best of our knowledge, the problem of finding an optimal distribution strategy in this general case is open. For example, in [7] the authors formulate one version of the problem using dynamic programming and find the optimal (centralized) solution for some special cases that are interesting, yet unrealistic. Here, since we want to come up with a simple practical scheme, we have chosen to be greedy in the spraying phase, using Binary spraying to minimize the amount of time it takes to spray all L copies. Then, we move the problem of looking for a possibly better relay to the focus phase (more on this later).

Number of copies used: Another interesting question is how many copies to use per message, i.e. what the value of L should be. In general, we would like this number to be only a small percentage of the total number of nodes. For the case of simple controlled replication, or Spray and Wait, analytical expressions exist that can calculate the number of copies needed to achieve an average delay that is α times the optimal one [17]. This could be performed by replacing α and the number of nodes M in the following equation, and solving for L[1]:

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

where $H_n^r = \sum_{i=1}^{n} \frac{1}{i^r}$ is the n^{th} Harmonic number of order r.

It can be shown that the delay of Spray and Wait is also an upper bound on the delay of Spray and Focus [20] (using the same spraying mechanism and the same number of copies, and assuming infinite bandwidth and storage). As a result, we can use the value of L from Eq.(1) as a somewhat pessimistic analytic estimate of the number of copies to use in Spray and Focus to achieve a given performance. To calculate the optimal number for Spray and Focus is not trivial. Calculating the delay of even one copy, routed according to a given utility function, involves modeling the message's movement as a Markov Chain, and cannot be derived in closed form [20]. Instead, we have found that a useful rule of thumb is to use $\frac{1}{3}$ to $\frac{1}{2}$ of the number produced by Eq.(1).

2.2 Focus Phase

When a relay for a given message has only one forwarding token left for that message, it switches to the "Focus phase". Unlike Spray and Wait, where in the Wait phase messages are routed using Direct Transmission (i.e. forwarded only to their destination) [16,17], in the Focus phase a message can be forwarded to a different relay according to a given forwarding criterion. Specifically, these forwarding decisions are taken based on a set of timers that record the time since two nodes last saw each other. Although the use of last encounter timers has been proposed in the past(e.g. [4]), we argue that any scheme that takes advantage of these timers needs to be carefully designed for the specific environment in hand (e.g. sparse networks with stochastic mobility) in order to achieve good performance. Let us turn then our attention to these timers. (Due to lack of space we omit some of the details of our utility-based mechanism, which can be found in [20].)

Age of last encounter timers with transitivity: Let's assume that each node i maintains a timer $\tau_i(j)$ for every other node j in the network, which records the time elapsed since the two nodes last encountered each other as follows: initially set $\tau_i(i) = 0$ and $\tau_i(j) = \infty, \forall i, j$; whenever i encounters j, set $\tau_i(j) = \tau_j(i) = 0$; at every clock tick, increase each timer by 1[2]. Position information regarding

[1] This equation holds for mobility models, where the probability density function of the hitting time between nodes has at least an approximately exponential tail. Although it has been proven that this is the case in a number of mobility models [15, 18], it is not clear how it would be applied to models with heavy-tailed encounter times [3], for example.

[2] In practical situations, each node would actually maintain a cache of the most recent nodes that it has encountered.

Proceedings of the Fifth Annual IEEE International Conference
on Pervasive Computing and Communications Workshops(PerComW'07)
0-7695-2788-4/07 $20.00

different nodes gets indirectly logged in the last encounter timers, and gets diffused through the mobility process of other nodes. Therefore, we can define a utility function, based on these timers, that indicates how "useful" a node might be in delivering a message to another node.

A number of different utility functions could be envisioned for this purpose. These could also take into account other relevant information (e.g. GPS position, speed, history of encounters, etc.) in addition to the timer values. However, it is beyond the scope of this paper to evaluate all these options, and we defer this for future work. Some efforts towards the design of multi-parameter utility functions can be found in [12]. Here, for simplicity, we will assume that these timers is our utility function (i.e. messages get forwarded to nodes with smaller and smaller timer values for the destination). A gradient-based scheme over this time-varying utility field can then be used to deliver a message to its destination, as has been noted in [4, 10]. This scheme will try to maximize the utility function for the destination.

Definition 2.2 (Single-copy Utility-based Routing) *Let every node i maintain a utility value $U_i(j)$ for every other node j in the network. Then, a node A forwards to another node B a message destined to a node D, if and only if $U_B(D) > U_A(D) + U_{th}$, where U_{th} (utility threshold) is a parameter of the algorithm.*

Timer values have the desirable behavior that their expected value increases as a function of distance. However, timers also quickly become poorer indicators of proximity as their value increases (this is commonly referred to as the "distance effect" [4]). In order to improve the efficiency of utility-based routing it is therefore desirable to reduce the uncertainty for higher timer values. To deal with this problem we propose the use of "transitivity" when updating the utility function. *When node A sees node B often, and node B sees node C often, A may be a good candidate to deliver a message to C (through B), even if A rarely sees C.* Therefore, when A encounters node B, it should also update (increase) its utility for all nodes for which B has a high utility. Although the idea of using transitivity has already been proposed in the context of a utility-based flooding scheme in [10], from an information theoretic perspective, this transitivity effect should successfully capture the amount of uncertainty resolved regarding the position of the destination, when a new node is encountered that has some additional (i.e. more recent) information[3]. We propose the use of the following transitivity function (details about the rational of this choice can be found in [20].

[3]Although heuristic transitivity functions like the one proposed in [10] might increase the number of forwarding steps per message, they may also decrease the "quality" of each step; the protocol's behavior starts resembling that of randomized routing.

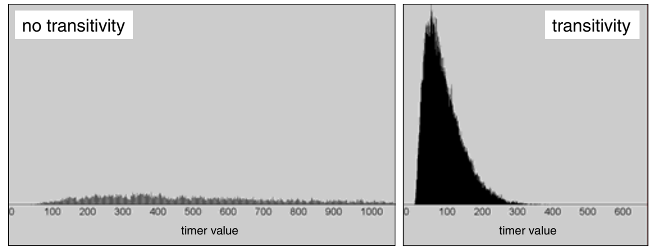

Figure 1. PDF of timer values for a node far from the destination, without transitivity (left) and with transitivity (right); ($N = 100 \times 100$, random walk mobility).

Definition 2.3 (Timer Transitivity) *Let a node A encounter a node B at distance d_{AB}. Let further $t_m(d)$ denote the expected time it takes a node to move a distance d under a given mobility model. Then: $\forall j \neq B : \tau_B(j) < \tau_A(j) - t_m(d_{AB})$, set $\tau_A(j) = \tau_B(j) + t_m(d_{AB})$.*

For example, the functions for the Random Waypoint and Random Walk mobility models would be:

$$\text{if } \tau_A(D) < \tau_B(D) - d_{AB}, \quad \text{set} \quad \tau_B(D) = \tau_A(D) + d_{AB} \text{ (waypoint)},$$
$$\text{if } \tau_A(D) < \tau_B(D) - d_{AB}^2, \quad \text{set} \quad \tau_B(D) = \tau_A(D) + d_{AB}^2 \text{ (walk)}$$

These transitivity functions quickly distribute fresh utility information far from the destination. As a result, they reduce the uncertainty regarding the position of a given node. As an example of the beneficial effect of transitivity on the accuracy of the information contained in timers, see Fig. 1, where a significant decrease in variance is achieved by transitivity.

2.3 Spray and Focus

We summarize here the functionality of the Spray and Focus protocol:

Message summary vectors: each node maintains a vector with IDs of all messages that it has stored, and for which it acts as a relay; whenever two nodes encounter each other, they exchange their vectors and check which messages they have in common; each message also carries a TTL (time-to-live) value; if that expires, the message gets discarded and its entry in the message summary vector erased.

Last encounter timers: each node i also maintains a timer $\tau_i(j)$ for every other node j in the network. Let $t_m(d)$ denote the expected time it takes a node to move a distance d under a given mobility model "m". Then, if a node A encounters a node B at distance d_{AB}:

- set $\tau_i(j) = \tau_j(i) = 0$;

0-7695-2788-4/07 $20.00 © 2007 IEEE

- $\forall j \neq B : \tau_B(j) < \tau_A(j) - t_m(d_{AB})$, set $\tau_A(j) = \tau_B(j) + t_m(d_{AB})$.

Spray and Focus forwarding: when a new message is generated at a source node create L "forwarding tokens", with L chosen according to Section 2.1; if a node (either the source or a relay) carries a message copy and: (i) $n > 1$ forwarding tokens — perform *Binary Spraying [17]*; (ii) $n = 1$ forwarding token — perform *Utility-based Forwarding* according to Definition 2.2 with the last encounter timers used as the utility function.

3 Simulation Results

In these section, we would like to compare the performance of Spray and Focus against existing flooding-based schemes, as well as against simple controlled replication algorithms. First, we would like to evaluate what the effect of connectivity is on the performance of different protocols. Intuitively, the less sparse the network the more the communication opportunities between nodes, and, thus, the better we expect the performance of the protocols to be. We'll use the same connectivity metric as [17].

Furthermore, we would like to evaluate the effect of dynamic connectivity on different protocols, that is, the effect of the mobility model. We'll first look at two popular mobility models, namely the Random Waypoint and the Random Walk models. Although these two models are rather simplistic, assuming IID node mobility among other things, they stand on opposite ends of the dynamic connectivity spectrum, and could be seen, to an extend, as a high-level abstraction of very "mobile" models (Random Waypoint) or very "local" ones (Random Walk). Specifically, the Random Waypoint model has one of the fastest mixing times ($\Theta(\sqrt{N})$), while the random walk has one of the slowest ones ($\Theta(N)$) [1].

We have used a custom discrete event-driven simulator to evaluate and compare the performance of the different routing protocols. A simplified version of the slotted CSMA (Carrier-Sense Multiple Access) MAC protocol has been implemented to arbitrate between nodes contenting for the shared channel (further details about the implementation can be found in [19]).

The routing protocols: We have implemented and compared the following routing protocols. We only depict results for multi-copy schemes in this study (a comparison of single-copy schemes could be found in [20]):

Epidemic routing ("epidemic") [21]: a node copies a message to every new node it encounters that hasn't got a copy already. For this and all other protocols, we choose TTL (time-to-live) values between $1000 - 10000$ time units for each message.

Randomized flooding or Gossiping ("random-flood"): like epidemic routing, but a message only gets copied with some probability $p < 1$ (we've used values between 0.5 and 0.05).

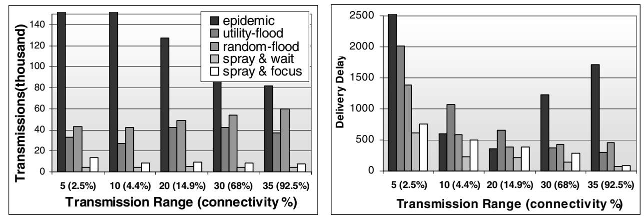

Figure 2. Random Waypoint Mobility.

Utility-based flooding ("utility-flood"): like epidemic routing, but a message gets copied only if the node encountered has a utility value higher than the current by some threshold U_{th} (we've used the utility function described in Section 2 the values for U_{th} are between 10 and 90).

Spray and Wait ("spray&wait"): We choose the number of copies L to be equal to about $10 - 15\%$ of all nodes M.

Spray and Focus ("spray&focus"): We have found that choosing L equal to about $5 - 10\%$ of the total nodes serves as a useful rule of thumb for good performance.

Random Waypoint Mobility: Let us first look at the performance under Random Waypoint mobility. This model has ideal characteristics for simple replication schemes like Spray and Wait, because all potential relays quickly get decorrelated (only after one epoch) and each relay moves on a straight line during an epoch, swiftly traversing a large part of the network. We'll assume that there are 100 nodes in a 200×200 network, and that a moderate number of CBR traffic sessions are uniformly distributed among all nodes to introduce contention.

Figure 2 depicts the number of transmissions and the average delay for the random waypoint, as a function of transmission range (respective connectivity values are shown in the parentheses). (We note that in all scenarios considered hereafter, all protocols except epidemic routing had delivery ratios above 90%, so we do not include plots for this metric.) As can be seen there, although Randomized and Utility Flooding can improve the performance of Epidemic routing they still have to perform way too many transmissions to achieve competitive delays. Nevertheless, Spray and Wait still manages to outperform all protocols in this ideal scenario, in terms of both transmissions and delay, for all levels of connectivity. Its performance was quite close to the optimal one (about $2\times$ larger), and Spray and Focus could not offer any improvement here (timers quickly become obsolete due to the high mobility).

Random Walk Mobility: Let us now assume that all nodes perform independent random walks (same network size and traffic load as in the Random Waypoint scenario). Although node mobility is still IID (uncorrelated), each node now takes a very long time to move from one side of

Proceedings of the Fifth Annual IEEE International Conference
on Pervasive Computing and Communications Workshops(PerComW'07)
0-7695-2788-4/07 $20.00 © 2007 IEEE

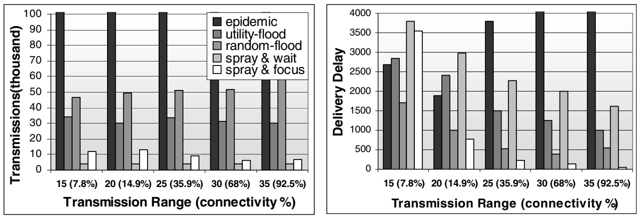

Figure 3. Random Walk Mobility.

the network to another, and thus carrying a message is not as beneficial as in the Random Waypoint case. Figure 3 depicts the number of transmissions and the average delay of all schemes for this scenario. As can be seen there, the delay of Spray and Wait suffers significantly in this scenario. Here, the few copies are spread locally, and then each message relay takes a very long time to traverse the network and reach the destination. Even if the number of copies were increased, the delay of the spraying phase would still dominate performance, since new nodes are found very slowly. On the other hand, Spray and Focus can overcome these shortcomings and excel (unless the network is too sparse, in which case forwarding opportunities are rare), achieving the smallest delay ($> 10\times$ faster than Spray and Wait) with only a few extra transmissions. Note also that, despite using the same utility function as Spray and Focus, Utility Flooding is still plagued by its flooding nature. This problem was even more pronounced when other existing utility functions were used [10]. This implies that *in disconnected networks, the use of a utility function is not enough by itself to improve performance, but rather has to be combined with controlled replication*.

Community-based Mobility: Popular mobility models like the ones we've examined so far, assume that each node may move equally frequently to every network location. Furthermore, such models usually assume that all nodes have the same mobility characteristics. However, numerous recent studies based on mobility traces from real networks (e.g. university campuses, conferences, etc.) have demonstrated that these two assumptions rarely hold in real-life situations [3, 11]. For this reason, we would also like to compare the performance of all protocols under a more realistic mobility model, called "Community-based Mobility Model", that is motivated by such traces and better resembles real node movement [18].

In the Community-based model, each node has its own small community ($c\times$ the size of the network, $c < 1$) inside which it moves preferentially for the majority of time (e.g. the user's department building on a campus). Every now and then it leaves its community (with probability p_l) and roams around the network for sometime (e.g. going to a class at a different building, to a dining hall, library, etc.), and the decides to return to its community (with probability p_r). Finally, each node may have different mobility characteristics (p_l, p_r) in addition to different communities. Some nodes may spent a very large amount of their time inside their community, while others may be more "mobile". The Community-based model allows for a large range of node heterogeneity to be captured, including even access points. (Further details about the model can be found in [18].)

We have compared the performance of all protocols in community-based mobility scenarios and we can draw similar conclusions about the relative performance of schemes (i.e. spraying schemes significantly reduce transmissions, and Spray and Focus achieves the minimum delays among all). Due to lack of space, respective simulation plots can be found in [19]. However, in order to better demonstrate the potential improvement that Spray and Focus can achieve over existing spraying schemes, in Fig. 4 we compare the delays of Spray and Focus and Spray and Wait only. We do so for 3 different scenarios with heterogenous mobility and mobility showing strong location preference. We measure the improvement achieved by our scheme as the ratio of the delay of Spray and Wait (SW) over that of Spray and Focus (SF) in the following scenarios:

Scenario 1: In this scenario the community size c is $\frac{1}{25}$. There is only one type of nodes, but p_l and p_r are chosen uniformly and independently for each node in $[0.05, 0.2]$ and $[0.6, 0.8]$, respectively.

Scenario 2: In this scenario we assume a bimodal distribution for node mobility. Specifically, there are two groups of differently behaving nodes: (*local nodes*) 90% of all nodes move locally most of the time (p_l chosen in $[0.05, 0.15]$) but may occasionally roam in the whole network ($p_r = [0.8, 0.9]$); (*roaming nodes*) 10% of the nodes roam quite often outside their community ($p_l = [0.2, 0.3]$ and $p_r = [0.5, 0.7]$).

Scenario 3: This last scenario includes the largest diversity among nodes: (*local nodes*) 40% of the nodes move locally most of the time ($p_l = [0.05, 0.15]$) but may occasionally roam in the whole network ($p_r = [0.8, 0.9]$); (*community nodes*) 40% of the nodes move only locally inside their own community; (*roaming nodes*) 10% of the nodes roam quite often outside their community ($p_l = [0.2, 0.3]$ and in $p_r = [0.5, 0.7]$); (*base stations*) 10% of the nodes are static and uniformly distributed in the network, corresponding for example to base stations or static sensors.

As is evident by Fig. 4, Spray and Focus, can clearly take better advantage of higher node heterogeneity and higher location preference and improve the performance of Spray and Wait by up to $20\times$.

4 Conclusion

In this work we have proposed an efficient mobility-assisted routing protocol to deliver data end-to-end in net-

Proceedings of the Fifth Annual IEEE International Conference
on Pervasive Computing and Communications Workshops(PerComW'07)
0-7695-2788-4/07 $20.00 © 2007 IEEE

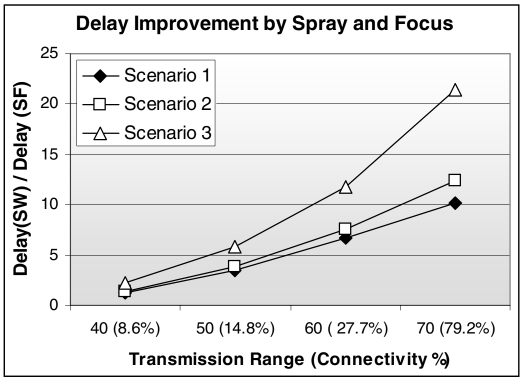

Figure 4. Performance improvement of Spray and Focus (SF) over Spray and Wait (SW) for community-based heterogeneous mobility.

works where connectivity is intermittent. Our scheme, Spray and Focus, builds upon a previous observation that controlled replication can be beneficial [16, 17]. However, it has increased intelligence compared to existing schemes in that it can successfully recognize and take advantage of potential opportunities to forward a message "closer" to its destination, according to an appropriately designed utility-function. As a result, it can achieve very good performance also in situations where existing spraying schemes may suffer significantly. Specifically, simulations we performed for popular as well as more realistic mobility models, motivated by existing real-world traces, show that Spray and Focus not only outperforms all existing mobility-assisted protocols in terms of both number of transmissions and delivery delay, but also reduces the delay of simple controlled replication algorithms by up to 20 times in some scenarios. This study, we believe, clearly identifies controlled replication and utility-based forwarding as a valuable combination for efficient routing in challenged networks.

References

[1] D. Aldous and J. Fill. Reversible markov chains and random walks on graphs. (monograph in preparation.). *http://stat-www.berkeley.edu/users/aldous/RWG/book.html*.

[2] E. Brewer, M. Demmer, B. Du, M. Ho, M. Kam, S. Nedevschi, J. Pal, R. Patra, S. Surana, and K. Fall. The case for technology in developing regions. *IEEE Computer*, 2005.

[3] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Impact of human mobility on the design of opportunistic forwarding algorithms. In *Proceedings of IEEE INFOCOM*, 2006.

[4] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli. Age matters: efficient route discovery in mobile ad hoc networks using encounter ages. In *Proceedings of ACM MobiHoc*, 2003.

[5] V. C. Giruka and M. Singhal. Hello protocols for ad-hoc networks: Overhead and accuracy tradeoffs. In *Proceedings of ACM/IEEE WOWMOM*, 2005.

[6] S. Jain, K. Fall, and R. Patra. Routing in a delay tolerant network. In *Proceedings of ACM SIGCOMM*, Aug. 2004.

[7] A. Jindal and K. Psounis. Optimizing multi-copy routing schemes for resource constrained intermittently connected mobile networks. In *to appear in IEEE Asilomar Conference on Signals, Systems and Computers*, 2006.

[8] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein. Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with zebranet. In *Proceedings of ACM ASPLOS*, 2002.

[9] J. Leguay, T. Friedman, and V. Conan. DTN routing in a mobility pattern space. In *Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN)*, 2005.

[10] A. Lindgren, A. Doria, and O. Schelen. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mobile Computing and Communication Review*, 7(3), 2003.

[11] M. McNett and G. M. Voelker. Access and mobility of wireless pda users. *ACM Mobile Computing and Communication Review*, 2003.

[12] M. Musolesi, S. Hailes, and C. Mascolo. Adaptive routing for intermittently connected mobile ad hoc networks. In *Proceedings of ACM WOWMOM*, 2005.

[13] M. Musolesi and C. Mascolo. A community based mobility model for ad hoc network research. In *Proceedings of ACM REALMAN*, 2006.

[14] J. Ott and D. Kutscher. A disconnection-tolerant transport for drive-thru internet environments. In *Proceedings of INFOCOM*, 2005.

[15] R.Groenevelt, G. Koole, and P. Nain. Message delay in manet (extended abstract). In *Proc. ACM Sigmetrics*, 2005.

[16] T. Small and Z. Haas. Resource and performance tradeoffs in delay-tolerant wireless networks. In *Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN)*, 2005.

[17] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Spray and wait: Efficient routing in intermittently connected mobile networks. In *Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN)*, 2005.

[18] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Performance analysis of mobility-assisted routing. In *Proceedings of ACM/IEEE MOBIHOC*, 2006.

[19] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Efficient routing in intermittently connected mobile networks: The multiple-copy case. *to appear in Transactions on Networking*, 2007.

[20] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Efficient routing in intermittently connected mobile networks: The single-copy case. *to appear in Transactions on Networking*, 2007.

[21] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University, 2000.