

Ferry Access Points and Sticky Transfers: Improving Communication in Ferry-assisted DTNs

Farzana Yasmeen[†], Nurul Huda, Shigeki Yamada
Department of Informatics, Soken-dai[†]
National Institute of Informatics (NII)
Tokyo, Japan
{bonhomie, huda, shigeki}@nii.ac.jp

Cristian Borcea
Department of Computer Science
New Jersey Institute of Technology (NJIT)
New Jersey, USA
borcea@cs.njit.edu

Abstract—Message ferries (MF) have been proposed as special nodes to improve communication in sparse, intermittent networks, such as delay-and disruption-tolerant networks (DTNs) by providing non-random message transfer opportunities to participating nodes. However, considering the mobility of nodes and the speed of the ferry, merely ensuring non-random encounters with hosts and the ferry do not guarantee optimal performance as the contact duration may not be enough to ensure adequate message transfers. The end-to-end performance could suffer significantly if message sizes are large. To address this issue, we present two methods that improve the performance of ferry-assisted DTNs: ferry access points (FAPs) and sticky transfers. FAPs supplement a ferry-assisted DTN with rendezvous points that increase the contact opportunities between mobile nodes and the ferry. Sticky transfers, upon mutual agreement, allow two encountered nodes to remain within transmission range of each other until they are able to complete necessary message transfers. This improves the number of successfully forwarded messages by allowing more messages to be transferred during the contact duration. Sticky transfers inherently reduce message transfer aborts. Our simulation results demonstrate that particular FAP placements improve the delivery ratio by as much as 25% for sparse networks and reduce the delay by as much as 40% for relatively dense networks. Sticky transfers on top of FAP-based DTNs achieved 100% delivery ratio for most situations, except in the case of highly-loaded, dense networks. Additionally, using sticky transfers resulted in lower delays by as much as 65% compared to non-sticky transfer cases.

Keywords—delay-tolerant networks; message ferries; ferry access points; sticky transfer protocol; performance evaluation;

I. INTRODUCTION

Message ferries (MF) have been proposed as a method to improve communication in DTNs [1, 2] by providing non-random message transfer opportunities in sparse, intermittent networks. The MF scheme utilizes special mobile nodes called ferries to provide communication services for nodes in the network [4, 5, 6, 7]. The ferries move in deployed areas along predefined routes, collect messages from source hosts, and carry them for delivery to corresponding destination hosts. Since ferries are restricted to predefined routes, mobile nodes must rely on opportunistic encounters with the ferry. Well-chosen paths for MFs can significantly improve the

probability of encounters; hence achieving higher delivery ratios and lower delays in the network. This paper presents two methods to further improve on the performance of ferry-assisted DTNs, namely: *ferry access points* (FAPs) and *sticky transfers*. FAPs are stationary rendezvous nodes deployed along the path of the ferry, which increase contact opportunities in ferry-assisted DTNs, while sticky transfers increase the contact duration among encountered mobile nodes to improve the capacity of the network and reduce transfer aborts. During a message exchange between two devices, limitations in the actual time to transfer data are observed due to the inherent mobility of hosts. When nodes move out of each others' transmission range, messages which are ready for forwarding cannot be forwarded. Additionally, if a message happens to be 'on the fly' during disconnections, the transmission is aborted. The *sticky transfer* mechanism uses coordination between nodes to achieve contact durations long enough to ensure sufficient data transfer among nodes before the contact is broken. In this method, once nodes have encountered each other, based on user preferences, they agree to stay within the transmission range of each other for a pre-negotiated amount of time.

To evaluate the two proposed methods, we performed simulations in a realistic city-based network environment with mobile nodes and a message ferry. We added FAPs at different positions of the ferry route based on proposed FAP placement models. We simulated the behavior of applying sticky transfer to the scenario with the best FAP placement model under varying loads, nodes densities, and bandwidth conditions. Our results show the following:

- Adding FAPs to an MF-assisted network always improves the performance of the network under all observed conditions. In node densities and shorter transmission ranges, the delivery ratio can be increased up to 25% compared to a network without FAPs.
- Sticky transfer greatly improves the performance of the network under sparse conditions, where infrastructure-support such as FAPs alone cannot improve the performance significantly as the encounter probability is

very low. Our results show that using sticky transfers in low node densities improve the delivery performance up to 70% in case of high loads and to 100% in case of low load situations. Sticky transfers also reduces the average end-to-end delay up to 65%.

- 'Waypoint' placement, where FAPs are placed at the points where the ferry stops briefly, can achieve a higher delivery performance and lower latency compared to other FAP placement strategies.

II. BACKGROUND AND RELATED WORK

Previous MF research explores how ferry routes can be optimized in terms of delay and bandwidth for stationary nodes [4]. A probabilistic approach to MF route design is presented in [6] where the ferry contacts stationary nodes with some probability p by waiting a finite amount of time in a certain area. Further work on MFs involves algorithms deploying multiple ferries into a network [7] and election algorithms to find ferry replacements in multiple ferry networks [8]. All of these papers consider stationary hosts only. In [5] it is explored how mobile nodes can proactively adapt trajectories to meet and exchange messages with a ferry.

Recent efforts have proposed adding infrastructure/ stationary resources to mobile networks to improve connectivity [9, 10, 11, 12]. Infrastructure, such as open access points, can be used opportunistically [9, 10]. The costs of adding different types of infrastructure are highly variable. A comparison among relays, base stations, and meshes [14] show that the average packet delivery delay in a vehicular deployment can be reduced by a factor of two by adding x base stations, the same reduction requires $2x$ mesh nodes or $5x$ relays. Their study also states that relays can be placed with least cost in networks that require no connections to electrical infrastructure or to the Internet and can be placed anywhere in the network [13]. Given the high cost of deploying base stations, relays are a more cost-effective solution for typical DTN scenarios. Infostations [11] and throwboxes [12] have similar strategies for improving performance of intermittent networks with infrastructure. In the Infostation model, users can connect to the network in the vicinity of ports (or Infostations), which are geographically distributed throughout the area of network coverage. Infostations provide strong radio signal quality to small disjoint geographical areas and offer very high rates to users in these areas. Throwboxes are relay nodes deployed anywhere in the network area and route information between mobile nodes in a disruption-tolerant fashion. FAPs [15] share the relay concept of throwboxes. In [15] we demonstrate that FAPs can increase contact opportunities by reducing the number of missed contacts between mobile nodes and the ferry. In this paper we show that specific FAP placement strategies can improve the performance of the network. To the best of our knowledge, there is no work closely related to sticky transfers.

III. FERRY ACCESS POINTS

A. Network Model Assumptions

We assume a sparse network composed of n mobile hosts, 1 MF and k FAPs. We consider the network to be deployed over a finite, two dimensional space. All devices, MF, MH and FAPs, communicate with each other through short range wireless interfaces (e.g., 802.11) and are equipped with data storage. Mobile hosts (MHs) generate data for other MHs in the network in the form of application layer data units called messages (i.e., bundles [2]). MHs can exchange messages with one another when they are within communication range. We assume messages are unicast.

A special node called the message ferry (MF) helps with the delivery of messages between MHs. The ferry achieves this by traversing a predetermined route repeatedly in the network area. We refer to each traversal through this route as a *tour*. If l is the length of the ferry route and f is the ferry constant speed then the time required by the ferry to complete one tour is l/f ; we call this time the tour-time. A ferry route is defined as $(W; T; f; r)$. W is a finite, ordered set of MF waypoints drawn from the network deployment area. $T = T_s : s \in W$ is the set of corresponding waiting times on the chosen waypoints and r is the transmission range. We assume waypoints to be co-located at points of interest (POIs). POIs are certain locations that mobile nodes visit with higher probability than other locations in the network. MF is neither a source nor destination and can interact with mobile hosts or FAPs.

We assume FAPs are disjoint and cannot directly communicate with one another. We envision FAPs as stationary, inexpensive, disposable units which have the following three features- (a) persistent storage, (b) wireless radio transceiver: speeds and ranges would be typical of Wi-Fi 802.11b or 802.11g specifications, and (c) an energy source: such as large batteries, which would supply energy to the FAPs for a time in the range of 36 to 48 hours.

B. Communication Model

In our communication model, we assume that all devices (MF, FAPs and MHs) are equipped with a similar radio of given communication range with radius, r . Devices can communicate with each other only when they are within a distance, d_i of each other that is less than the communication range, i.e., $d_i \leq r$. Devices are said to be in contact when they are within the communication range of each other. Upon each successful contact between a ferry or FAP and an MH, the ferry (or FAP) gives any messages it is carrying to the MH if the MH is the intended recipient of the message. Next, within the contact duration, the MH forwards messages onto the ferry. The MH retains copies of forwarded messages to forward to other MHs if an opportunity presents itself. When there is contact between two MHs, messages are exchanged only if the other MH has not already received a copy of that

message. Upon successful contact between a FAP and ferry, the ferry transfers messages to the FAP and keeps a copy of the messages in its buffer. Then the FAP transfers messages to the MF while retaining a copy of the message in its buffer. Copy retaining policy is implemented to improve delivery probability to final destinations. However, when receiving a message from a ferry, the FAP 'marks' the packet so that it is not re-transferred to the ferry again.

C. FAP Placement Models

We assume the ferry tours on a pre-defined route in the network. $W_i = (x_i, y_i)$ is the set of finite ferry waypoints, where $i = 1 \dots n$. Let ' k ' be the number of deployable FAPs and $h = (x_h, y_h)$ is the location where a FAP can be placed. We assume:

a) FAP's are placed on MF routes and not elsewhere in the network area.

b) At any time, at most one FAP can be located at a waypoint or between a pair of consecutive ferry waypoints in W . We denote this by $k_h \in \{0, 1\}$, where 0 is the absence of any k^{th} FAP at location h and 1 is its presence at location h . The number of deployable FAPs is upper bounded by the number of ferry waypoints and $k < |W_i|$.

With above assumptions we propose the following placement models:

1. Waypoint model: In this model, FAPs are placed on MF waypoints and $h \in W$.

2. Symmetric model: In the symmetric model, FAPs are placed at mid-Euclidian distance between two consecutive ferry waypoints in W on the ferry route.

3. Symmetric-Plus model: In the symmetric-plus, or symmetric+ model FAPs are placed at a distance d from the symmetric placement and $d \leq r$, where r is the transmission range of the ferry. This means, the FAP should be within communication range of the ferry as the ferry passes by.

4. Asymmetric model: In this model, FAPs are placed randomly between consecutive ferry waypoints.

Among the above models, *symmetric* placement is the most uniform, while *asymmetric* is a random strategy. *Symmetric+* placement increases the transfer duration between the ferry and the FAP compared to the *symmetric model*. *Waypoint* placement shares more contact opportunities with mobile nodes, compared to other models as they are placed at POIs. We evaluate the performance of the models with simulations in section V.

IV. STICKY TRANSFERS

Opportunistic networks (e.g. DTNs) rely on encounters of nodes to forward messages in the network. Routing protocols [17, 18] can determine what messages a node can forward to which encounter. The network performance of a DTN highly depends on time elapsed between encounters (inter-contact time) and the time two nodes remain in each others communication range once a contact is established

(contact-duration). The contact duration directly influences the capacity of opportunistic networks because it can limit the amount of data transferred between nodes. Using the proposed sticky transfer scheme, nodes can cooperatively increase the contact duration to improve the capacity of the network by agreeing to brief, temporary modifications in movement patterns. Nodes may opt-in or opt-out from sticky transfers at any time. This section presents the concept of sticky transfers (*sticky* has been used interchangeably) and the sticky transfer protocol for exchanging messages.

A. Definitions and Assumptions

The *natural contact duration* (or, contact time), T^C is the length of time for which two nodes remain within the transmission range of each other. We assume that a node is always in the transmission range of only one node. Thus, if at time t_0 , A comes into the transmission range of B and moves away from B at time t_1 then $T^C(A, B) = t_1 - t_0$. During sticky transfers, if multiple nodes are in the transmission range of each other, we assume the mutual encounter sequence comes naturally from the order in which nodes hear advertisement messages (i.e., "hello" beacons) from the other nodes. On the other hand, the time required to for A to complete transferring messages (defined by the routing protocol) to B is the required transfer duration, T^R . Let C be the transmission speed of the nodes. If node A has p messages to transfer to node B where, L_k is the size of the k^{th} message for $1 \leq k \leq p$, and node B has q messages to be transfer to node A , where L_l is the size of the l^{th} message for $1 \leq l \leq q$, then the *required transfer duration* between A and B is:

$$T^R = \frac{\sum_{k=1}^p L_k + \sum_{l=1}^q L_l}{C}. \quad (1)$$

Assuming that the message transfer starts immediately after nodes encounter, if $T^C(A, B) < T^R(A, B)$ then it is probable that aborts will happen and all of the messages which A wants to forward to B cannot be transferred within the expected contact time. Due to the inherent mobility of nodes, this situation often arises in DTNs. We propose sticky transfers to remedy this situation. With sticky, encountered nodes come to a mutual agreement on the time they will remain in each other's communication range a-priori to message transfers. If T^C is the natural but insufficient contact duration to transfer messages, the additional time the nodes should stick beyond their natural contact duration is $\delta = T^R - T^C$. Here, δ is the *sticky duration*, which is calculated by nodes using the sticky transfer protocol, defined in the following sub-section.

B. Sticky Transfer Protocol

In the proposed scheme, in order to 'stick', nodes may modify movement behavior based on mutual agreements

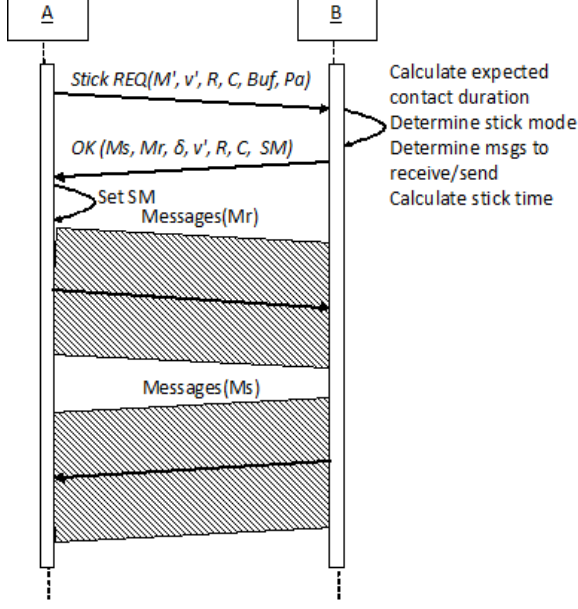


Figure 1. Sticky transfer protocol sequence diagram.

using movement status and preference information when the natural contact duration is not enough. As changing node movement depends upon user cooperation, to implement sticky transfers we realize users' agreement of modified movement through *user preferences*, P_i , where ' i ' is any user in the network. A user preference consists of an ordered list of acceptable sticky modes. The order defines the priority of user preferences, with higher priority modes coming first in the list.

Stick Modes: To accommodate user preferences, we propose five operational *sticky modes*: *Stop*, *Follow me*, *Follow you*, *Slow down* and *No stick*; which define policies for changing nodes' natural movement and speed, such that they may remain within the transmission range of another node for a pre-negotiated amount of time. The *Stop* (STP) mode is implemented by changing the relative speed of two nodes to zero. One way to achieve this is to change both of the nodes velocity to zero, i.e., stopping the nodes. Another way of achieving zero relative speed is to move one node with the same speed as the other in the same direction i.e., one node follows the other node. The mode of the node which is followed by the other node is called *Follow me* (FLW1) mode. The node which adjusts its speed and direction to the other node's speed and direction to follow the other node is called the *Follow you* (FLW2) mode. The relative speed of two nodes can be reduced by reducing one (or both) of the two node's speed. It may be that one node is already moving slower than the other. So, only the node that is moving faster needs to adjust its speed by slowing down. When a node reduces its speed to increase the contact duration we call it the *Slow down* (SLW) mode. Finally, a node may not agree

to stick for message transfers or may not even respond to stick requests. This mode is *No Stick* (NO STK) mode. When setting preferences, nodes may not be able to select some of the modes due to operational limitations. Furthermore, sticky transfers are technically possible when preferences of two nodes have 'compatible' modes that allow for the mobility of each to mutually extend the contact.

We assume that nodes have user preferences, P and status information I consisting of movement vectors, v' (i.e. speed, direction and current location), transmission range R , transmission rate C , free buffer size Buf , and message vectors, M' (i.e. message size and id). Fig. 1 shows the protocol sequence diagram for sticky message transfers (A sends the request first). We simplify the explanation of the protocol sequences here for brevity. Once A sends a sticky transfer request to B along with its status information, B calculates the natural contact duration, T^C between A and B . Next, B decides the messages (non-redundant) to receive from A and specifies them in a receive message vector M_r . and determines its own list of messages to send in the vector M_s . Both M_r and M_s are based on the empty buffer size of the intended receiver and expected transfer time. Next, B calculates an upper bound on the required transfer duration, T^R based on message sizes of the send message vector; receive message vector, and the transmission speed (Eq. 1). B compares T^C and T^R to determine if the natural contact duration is enough. B also determines if a compatible stick mode exists between A and B from preferences P_A and P_B . If T^C is sufficient for completing the message exchange, B will notify A through a NO_STK reply to notify sticky transfers are not necessary and set $\delta = 0$. The same response will result if A and B 's sticky preferences are not compatible. However, if sticky transfers are necessary (i.e. $T^C \leq T^R$) and compatible sticky modes exist, then B will set its own sticky mode and also set the mode of node A in a variable SM . Finally, node B will send an OK response (with send and receive message vector, stick duration, status information and stick mode) to A . Node A checks the response from B and sets its stick mode as defined in SM and then sends messages indicated in M_r to B , piggybacking the redefined M_s vector on one of the data messages. After receiving M_r , and the redefined vector M_s , B will send messages indicated in M_s to A to complete the transfer. After completing sticky transfers, the nodes will resume their natural movement.

As computation of the required transfer time may be slightly lower than necessary or nodes may opt-out from the stick transfer agreement, a limited number of aborts are still possible. Assuming that a routing protocol wants to spread K copies of a message to improve delivery, sticky transfers ensure that the K copies are spread faster in the network within fewer encounters. This leads to lower latencies and higher delivery ratios in network performance. However, if sticky transfers are used persistently in highly loaded

Table I
SIMULATION SETTINGS

	Speed	Max. number	Buffer size	Range	Transfer speed
Mobile node (MH)	0.5m/s - 10m/s	25	8GB	100m	11Mbps/54Mbps
Ferry (MF)	15m/s	1			
FAP	0 m/s	4			

network conditions, the effectiveness may be reduced due to lower mobility (i.e. nodes stopping for long periods to finish their exchanges). Sticky transfers may also lead to incessant buffer overflows when the network load is very high and the routing protocol makes multiple copies of each message.

V. PERFORMANCE EVALUATION

We evaluated the benefits of FAPs and sticky transfers on data transfers in DTN using simulations. We used a stick probability metric, SP to represent node willingness to sticky transfer requests. This SP is defined for each node in the network. A value of $SP = 0.0, 0.5$ and 1.0 indicate that a node will not agree, agree to 50% of the requests and always agree to a sticky transfer request respectively. We implement *Stop* (STP) as the mode for nodes to stick for message transfers.

A. Performance Metrics

The *message delivery ratio* is defined as the ratio of the number of successfully delivered messages to the total number of unique messages generated within the simulation duration. Our model assumes a FIFO policy for messages drops. The message delay is represented in time units of seconds as the average *end-to-end delay* and is the average time from the generation of a message to the earliest reception of the message at the destination. The message delay considers delivered messages only. The *overhead ratio* represents the quantity of excessive relays required on average to deliver messages to the destination and is an indicator of bandwidth and energy consumption. *Buffer occupancy* shows the amount of buffer space filled-up at nodes on average at any instance of time.

B. Simulation Settings

We used the ONE [16] simulator for simulating a downtown area ($4500m \times 3400m$) with 8 point-of-interests (POIs) to model shops, restaurants, tourist attractions, etc. The mobile nodes randomly select any one of these eight POIs to move. Pause times at POIs were between 1sec and 50secs. We consider a uniform traffic model where all mobile nodes were chosen as sources with random destinations. Mobile nodes use the Epidemic routing protocol [18] for forwarding message copies in the network. We simulate single ferry (MF) which moves according to the Map Route Movement

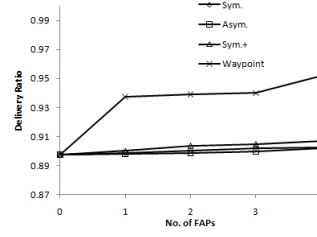


Figure 2. Delivery Ratio of FAP placement models.

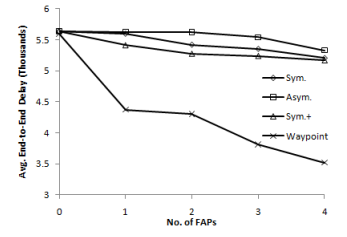


Figure 3. Avg. End-to-End Delay of FAP placement models.

model in the area and follows a certain route. Ferry waypoints were randomly drawn from the above eight POIs. The ferry wait time at each of the waypoints is 5secs. We model FAPs as stationary nodes in the simulation area and specify the exact location of each FAP in the network according to our proposed placement models in each simulation run. We ran each simulation for $80k$ secs. Message were generated every 5secs on average after a warm-up period of $1k$ secs and generation stopped after $31k$ secs. Message sizes, L were 5Mbytes and 20Mbytes. The average of 5 simulation runs have been plotted. We summarize important simulation settings in Table 1.

C. Results and Analysis

Under the above simulation settings, we first evaluate the performance of FAPs and compare the FAP placement models (without sticky transfer). We also observe the performance of FAP with sticky transfers under various network parameters.

1) *FAP Performance*: We evaluate each FAP placement model by varying the number of FAPs from 0 to 4. FAPs = '0' is the baseline case: meaning there are no FAPs or MF in the network; only MHs are present. We use 20 MHs in these simulations. For the *waypoint model*, FAPs were placed on ferry waypoints. For the *symmetric model*, FAPs are placed exactly at mid-points of two consecutive ferry waypoints. For the *symmetricPlus model*, FAPs are placed at a distance d from the *symmetric placement model* where $d \leq r$ and r is the range of the ferry. Under the *asymmetric model*, FAPs are placed randomly between consecutive ferry waypoints. We randomly generate coordinates within the map area for the *asymmetric* and *sym+* placement model. Devices transmission speeds of $C=54Mbps$ and $L=5MB$. From Fig.2 and Fig.3, we conclude that the waypoint placement model performs best under the given network conditions. Similar results were also found for other values of C and L .

2) *Sticky Transfer Performance*: We evaluate the benefits of the sticky transfer method in the MF network. We quantify the performance metrics by increasing the network density and comparing the following scenarios: (i) MH-only (no MF, FAPs, or sticky transfer); (ii) MHs with MF and FAP waypoint placement (no sticky transfers) and (iii) MHs with

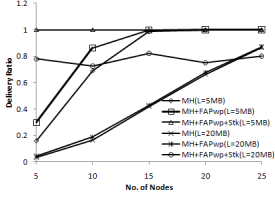


Figure 4. Delivery Ratio for $C=11\text{Mbps}$, $L=(5\text{MB}, 20\text{MB})$.

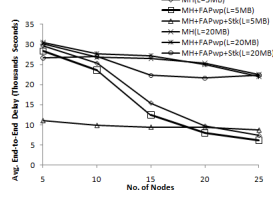


Figure 5. End-to-End Delay for $C=11\text{Mbps}$, $L=(5\text{MB}, 20\text{MB})$.

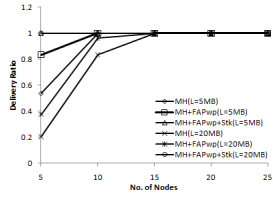


Figure 6. Delivery Ratio for $C=54\text{Mbps}$, $L=(5\text{MB}, 20\text{MB})$.

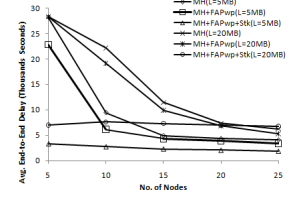


Figure 7. End-to-End Delay for $C=54\text{Mbps}$, $L=(5\text{MB}, 20\text{MB})$.

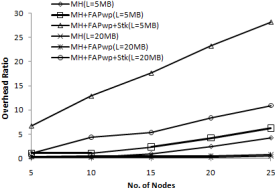


Figure 8. Overhead Ratio for $C=11\text{Mbps}$, $L=(5\text{MB}, 20\text{MB})$.

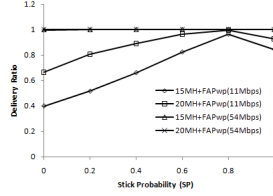


Figure 9. Delivery Ratio with varying stick probability (SP) for $MH=(15,20)$, $C=11\text{Mbps}$, $L=20\text{MB}$.

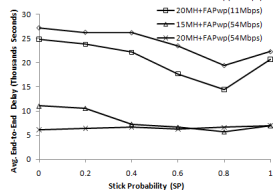


Figure 10. End-to-End Delay with varying stick probability (SP) for $MH=(15,20)$, $C=11\text{Mbps}$, $L=20\text{MB}$.

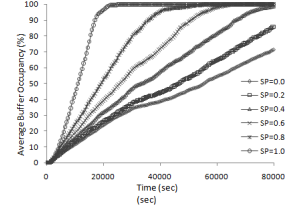


Figure 11. Buffer Occupancy at different stick probabilities (SP) for $MH=20$, $C=11\text{Mbps}$, $L=5\text{MB}$.

MF, FAP waypoint placement, and sticky transfer. We set $SP=1.0$ for nodes and the ferry, unless otherwise specified. FAPs always agree to stick (i.e. $SP=1.0$), as they are stationary nodes. We consider the scenario with 5 MHs to be a very sparse network. We also vary the bandwidth of the network by varying the transmission speeds, C , of all nodes: low bandwidth scenarios use 11Mbps , and high bandwidth scenarios use 54Mbps . We vary the network load by changing the message size: lower loads use $L=5\text{MB}$; higher loads use $L=20\text{MB}$.

Fig. 4 shows in case of low loads ($L=5\text{MB}$) when using the sticky transfer method, a 100% the delivery ratio is achieved under all density conditions. This is highly efficient especially for sparse networks. In case of high loads ($L=20\text{MB}$), the delivery ratio of sticky transfers is again best and steady, but only around 70%. Since the size of the messages is larger, it takes longer to finish a sticky session, and thus, the number of contacts between nodes in DTN decreases. Therefore, the positive effect of sticky transfer is counter-balanced by the negative effect of reduced contacts in DTN. Fig. 5 presents the end-to-end delays for $C=11\text{Mbps}$. In the case of low loads ($L=5\text{MB}$) and low network densities, the sticky transfer method reduces the delay up to 50%. However, under higher node densities, MH-only and MH with FAPs reduce the delay more by having more encounters with other nodes. In the case of high loads ($L=20\text{MB}$) and low node densities, the delay of sticky transfer is higher in comparison. This is because the delay is calculated as average for only those messages that are delivered successfully. Therefore, even though very few messages are delivered those are the ones that are delivered are upon encounters with direct destinations. Fig.

6 demonstrates that sticky transfer is capable of achieving 100% delivery ratio for both types of loads even in very sparse network conditions. As buffers are large and there is no TTL limit, all created messages are delivered within the simulation time due to the fast transmission speed of nodes. Fig. 7 shows that sticky transfer achieves lower delays in case of low network loads, ($L=5\text{MB}$) and low node densities by as much as 85%. This is because upon encounter, the required sticky-duration is much smaller and fewer contact opportunities are missed when nodes use sticky transfers. Fig. 8 shows the average overhead ratio at 11Mbps for both size messages. The overhead ratio is the number of extra forwards that messages (and all its copies) require for end-to-end delivery and is an indicator of bandwidth consumption. In the Epidemic protocol, multiple copies of the same message are created in the network until any one of the copies reaches the final destination. Compared to the MH-only case, the ferry network with sticky transfers spreads copies faster due to nodes always successfully forwarding copies because of sticky agreements. Also, small messages (i.e. 5MB) were spread faster compared to larger messages (i.e. 20MB), because within the same contact duration more smaller sized messages can be transferred. Thus, it could be said that to gain benefit of the high delivery ratios and faster delivery times, nodes would consume more energy due to message forwarding overhead when using sticky transfers. We found that at 54Mbps bandwidths for both size message, due to faster transmissions, both message sizes incur the same amount of overhead with sticky transfers. Fig. 9 shows that gradually increasing the stick probability significantly increases the delivery ratio, and is particularly effective for large message sizes. However, at $SP=1.0$, the delivery at

$C=54\text{Mbps}$ decreased from $SP=0.8$ because nodes always agreed to stick (STP is the sticky mode) and stopped for message transfers, which reduced node movement. Nodes sticking longer cause the number of contacts per hour to decrease significantly, reducing contact opportunities. In Fig. 10 the delay in general is more when transmission speeds are slower (11Mbps). As SP gradually increased, the delay decreased but increased again at $SP=1.0$ (due to reasons mentioned in Fig. 5 and Fig. 8). Fig. 11 shows the buffer occupancy (percentage of buffers becoming full) at mobile nodes on average over the simulation time for $C=11\text{Mbps}$. At $SP=1.0$ the buffers quickly become saturated, within 25% of the simulation time. As SP values increased, buffers became occupied more quickly. The same trends were found $C=54\text{Mbps}$, except buffers became fuller quickly at lower SP values due to faster transmission speeds. At high loads, high SP can cause *thrashing*, which is a condition where due to buffer limitations messages are dropped incessantly to make room for new ones. This can lead to reduced network performance.

VI. CONCLUSION

This paper demonstrates two fundamental ideas for significantly improving the performance of a delay-tolerant network (DTN) with ferries. By deftly placing ferry access points (FAPs) along the route of the ferry, both the delivery ratio and delay can be improved, with *waypoint* placement returning best results. By adding sticky transfers on top of FAP-inclusive MF networks, the delivery ratio can be further improved (reaching 100% in many situations), while the delay is significantly reduced compared to existing methods. The main benefit of sticky transfers come from the fact that multiple messages can be exchanged during longer contact times, thus propagating messages faster in the network. Reduced message aborts is a secondary benefit. Some side effects of using sticky transfers with high probability are increased energy consumption and thrashing. These drawbacks can be easily mitigated by choosing a sticky behavior that is suitable for the application and tuning the stick time ratio. We believe that the sticky transfer mechanism, and its associated benefits, is not only applicable for ferry-assisted DTNs but is rather a very broad framework that can be adopted widely for any opportunistic network application, such as rural area connectivity, environmental monitoring networks, emergency situations, search and rescue missions, etc. We plan to work extensively on the sticky transfer mechanism in the future.

REFERENCES

- [1] J. Pan, S. Paul, and R. Jain, *A Survey of the Research on Future Internet Architectures*, IEEE Communications Magazine, vol.49, no.7, pp.26-36, July 2011.
- [2] K. Fall, *A Delay-Tolerant Network Architecture for Challenged Internets*, ACM SIGCOMM, Germany, pp.27- 34, Aug. 2003.
- [3] E. M. Royer and T. Chai-Keong, *A Review of Current Routing Protocols for Ad hoc Mobile Wireless Networks*, IEEE Personal Communications, vol.6, no.2, pp.46-55, Apr. 1999.
- [4] W. Zhao and M. H. Ammar, *Message Ferrying: Proactive Routing in Highly-partitioned Wireless Ad hoc Networks*, IEEE FTDCS Workshop, Puerto Rico, pp.308-314, May 2003.
- [5] W. Zhao, M. Ammar, and E. Zegura, *A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad hoc Networks*, ACM MobiHoc, Japan, pp.187-198, May 2004.
- [6] M. B. Tariq, M. Ammar, and E. Zegura, *Message Ferry Route Design for Sparse Ad hoc Networks with Mobile Nodes*, ACM MobiHoc, Italy, pp.37-48, May 2006.
- [7] W. Zhao, M. Ammar, and E. Zegura, *Controlling the Mobility of Multiple Data Transport Ferries in a Delay-Tolerant Network*, IEEE INFOCOM, vol.2, pp.1407-1418, Mar. 2005.
- [8] J. Yang, Y. Chen, M. Ammar, and C.-K. Lee, *Ferry Replacement Protocols in Sparse MANET Message Ferrying Systems*, IEEE WCNC, vol.4, pp.2038-2044, Mar. 2005.
- [9] A. Balasubramanian, Y. Zhou, W. B. Croft, B. N. Levine, and A. Venkataramani, *Web Search from a Bus*, ACM CHANTS Workshop, Canada, pp.59-66, Sept. 2007.
- [10] V. Bychkovsky, B. Hull, A. Miu, H. Balakrishnan, and S. Madden, *A Measurement Study of Vehicular Internet Access Using in Situ Wi-Fi Networks*, ACM MobiCom, USA, pp.50-61, Sept. 2006.
- [11] D. J. Goodman, J. Borras, N. B. Mandayam, and R. D. Yates, *Infostations: A New System Model For Data And Messaging Services*, IEEE VTC, vol.2, pp.969-973, May 1997.
- [12] W. Zhao, Y. Chen, M. Ammar, M. Corner, B. Levine, E. Zegura, *Capacity Enhancement using Throwboxes in DTNs*, IEEE MASS, Canada, pp.31-40, Oct. 2006.
- [13] M. Ibrahim, A. A. Hanbali, and P. Nain, *Delay and Resource Analysis in MANETs in Presence of Throwboxes*, Performance Evaluation (Elsevier), vol.64, no.9-12, pp.933-947, June 2007.
- [14] N. Banerjee, M. D. Corner, D. Towsley, and B. N. Levine, *Relays, Base Stations, and Meshes: Enhancing Mobile Networks with Infrastructure*, ACM MobiCom, USA, pp.81-91, Sept. 2008.
- [15] F. Yasmeen, M. Huda, M. Haque, M. Aoki, S. Yamada, *Using Ferry Access Points to Improve the Performance of Message Ferrying in Delay-Tolerant Networks*, Proc. ICCNMC, France, pp.173-179, Aug. 2011.
- [16] A. Keranen, J. Ott, T. Karkkainen, *The ONE Simulator for DTN Protocol Evaluation*, Proc. SIMUTools, Rome, pp.1-10, Mar. 2009.
- [17] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, *Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks*, ACM WDTN Workshop, Rome, pp.1-10, Mar. 2009.
- [18] A. Vahdat and D. Becker, *Epidemic Routing for Partially Connected Ad hoc Networks*, Technical Report CS-2000-06, Duke University, April 2000.