Routing for Disruption Tolerant Networks: Taxonomy and Design

**Thrasyvoulos Spyropoulos · Rao Naveed Bin Rais · Thierry Turletti ·
Katia Obraczka · Athanasios Vasilakos**

Abstract Communication networks, whether they are
wired or wireless, have traditionally been assumed to be
connected at least most of the time. However, emerging
applications such as emergency response, special opera-
tions, smart environments, VANETs, etc. coupled with
node heterogeneity and volatile links (e.g. due to wire-
less propagation phenomena and node mobility) will
likely change the typical conditions under which net-
works operate. In fact, in such scenarios, networks may
be mostly disconnected, i.e., most of the time, end-
to-end paths connecting every node pair do not exist.
To cope with frequent, long-lived disconnections, *op-
portunistic routing* techniques have been proposed in
which, at every hop, a node decides whether it should
forward or store-and-carry a message. Despite a grow-
ing number of such proposals, there still exists little
consensus on the most suitable routing algorithm(s) in
this context. One of the reasons is the large diversity of
emerging wireless applications and networks exhibiting
such "episodic" connectivity. These networks often have
very different characteristics and requirements, making
it very difficult, if not impossible, to design a routing
solution that fits all.

In this paper, we first break up existing routing
strategies into a small number of common and tunable
routing modules (e.g. message replication, coding, etc.),
and then show how and when a given *routing module*
should be used, depending on the set of *network charac-
teristics* exhibited by the wireless application. We fur-
ther attempt to create a taxonomy for intermittently
connected networks. We try to identify generic *network
characteristics* that are relevant to the routing process
(e.g., network density, node heterogeneity, mobility pat-
terns) and dissect different "challenged" wireless net-
works or applications based on these characteristics.
Our goal is to identify a set of useful *design guidelines*
that will enable one to choose an appropriate routing
protocol for the application or network in hand. Finally,
to demonstrate the utility of our approach, we take up
some case studies of challenged wireless networks, and
validate some of our routing design principles using sim-
ulations.

Thrasyvoulos Spyropoulos
ETH Zurich, Switzerland
E-mail: spyropoulos.tik.ee.ethz.ch

Rao Naveed Bin Rais
INRIA - University of Nice, Sophia Antipolis, France
E-mail: Naveed-Bin.Rais@sophia.inria.fr

Thierry Turletti
INRIA, Sophia Antipolis, France
E-mail: Thierry.Turletti@sophia.inria.fr

Katia Obraczka
University of California at Santa Cruz, USA
E-mail: katia@soe.ucsc.edu

Athanasios Vasilakos
University of Western Macedonia, Greece
E-mail: vasilako@ath.forthnet.gr

1 Introduction

Traditionally, communication networks, regardless of
whether they are wired or wireless, have always been
assumed to be connected almost all the time. Here,
by connected networks, we mean that there exists at
least one end-to-end path between every pair of nodes
in the network most of the time. When partitions occur,
they are considered transitory failures and core network

functions such as routing react to these failures by attempting to find alternate paths. Even in wireless multi-hop ad-hoc networks (e.g., MANETs), where links are more volatile due to wireless channel impairments and mobility, partitions are still seen as exceptions and assumed infrequent and short-lived.

However, for some emerging applications like emergency response, special operations, smart environments, habitat monitoring, and VANETs, which are motivated by advances in wireless communications as well as ubiquity of portable computing devices, the assumption of "universal connectivity" among all participating nodes no longer holds. In fact, for some of those scenarios or applications, the network may be disconnected most of the time; in more "extreme" cases, there may never be an end-to-end path available between a source and a destination. Besides the application scenarios themselves, other factors contributing to frequent, arbitrarily long-lived connectivity interruptions include node heterogeneity (e.g., nodes with different radios, resources, battery life), volatile links (e.g., due to wireless propagation phenomena, node mobility), energy efficient node operation (e.g., duty cycling).

Networked environments which operate under such intermittent connectivity are also referred to as episodically connected, delay tolerant, or disruption tolerant networks (or DTNs). Clearly, traditional routing, including MANET routing protocols like OLSR [1], AODV [2], and DSDV [2] cannot deliver adequate performance in DTNs. Consequently, a number of new routing approaches have been proposed to cope with frequent, arbitrarily long-lived connectivity disruptions. They can be classified into three categories: *deterministic* or *scheduled*, *enforced*, and *opportunistic* routing. Deterministic routing solutions are used when contact information is known a priori. Jain et al. [3] showed how little or full information about contacts, queues, and traffic can be utilized to route messages from a source to a destination in the case of disruptions. They have presented a modified Dijkstra algorithm based upon information on scheduled contacts and compare the proposed approach against an optimal LP formulation. In order to deliver messages to otherwise disconnected parts of network (islands), enforced routing solutions like message ferries [4] and data mules [5] can be employed, where special-purpose mobile devices move over predefined paths in order to provide connectivity. Epidemic dissemination [6] is the basic form of opportunistic routing and works as follows. When node A encounters node B, it passes to B replicas of messages A is carrying which B does not have. In other words, epidemic routing is to episodically connected environments what flooding is to "traditional", well-connected networks. While on one hand

epidemic routing offers minimum delivery delay, it may be prohibitively expensive since it consumes considerable network resources due to the excessive amount of message duplicates generated.

Our focus here is on opportunistic approaches to DTN routing, i.e., where no contact information is known a priori and no network infrastructure (e.g., special-purpose nodes with controlled trajectories) exists to provide connectivity. Besides the question of when contact opportunities happen between nodes, a number of other factors also affect data forwarding, including available storage at peering nodes, contact duration, available bandwidth, message priority or expiration time, etc.

An ever growing number of protocols addressing these "opportunistic" DTN scenarios have been proposed. However, it is not at all clear how existing solutions can be applied to a variety of DTN applications given their requirements and underlying network characteristics (e.g., connectivity, node mobility and capability).

In this paper, we address this question and thus help map the design space of opportunistic DTN routing. We can summarize the contributions of this work as follows:

- First, we dissect opportunistic routing solutions identifying their basic *building blocks* in terms of the forwarding scheme employed, namely *message replication*, *forwarding*, and *(source and network) coding* (Section 2).
- We also identify a number of features that can be used to classify DTNs. Classifying DTNs according to their connectivity, mobility, and capability (i.e., storage, battery life, processing) of the participating nodes will be key to deciding what routing mechanism(s) to use in order to achieve adequate application-level performance (Section 4).
- We then proceed to map the opportunistic routing design space by drawing the correspondence between the proposed DTN taxonomy and the basic opportunistic routing building blocks (Section 5).
- Finally, through simulations, we conduct case studies of a number of challenged wireless network scenarios in order to validate some of our DTN opportunistic routing design principles and recommendations (Section 6).

The remainder of this paper is organized as follows. Section 2 discusses the routing strategies in intermittently connected network by dissecting the existing solutions into a small number of common and tunable routing primitives. Important utility functions for routing decisions are described in Section 3. Section 4 presents a DTN taxonomy by detailing the net-

work characteristics that are important in designing a routing protocol. DTN routing design guidelines and a discussion is presented in Section 5, and in the end, we provide some case studies of challenged wireless networks in Section 6.

2 Opportunistic Routing Primitives

In this paper, we focus on opportunistic routing approaches, i.e., where no information about connectivity or mobility is assumed to be known a priori and no special-purpose nodes (e.g., data mules or ferries) are used. The basic principle governing opportunistic routing is that when two nodes meet one another, they must decide whether to forward a message, or to carry it further. It represents a shift from basic "store-and-forward" to the so-called "store-carry-and-forward".

Due to its inherent characteristic of running without a priori knowledge, opportunistic routing is quite general and is also applicable to both scheduled and enforced connectivity scenarios since they may suffer from some non-determinism and uncertainty. For example, a bus that is scheduled to reach a bus stop at a certain instant may get stuck in a traffic jam, causing a deviation in its schedule, which ultimately may affect deterministic routing. Also, there can be other factors affecting scheduled behavior like weather, radio interference, and system failure.

Even though our focus in this paper is on networks or applications exhibiting frequent and long-lasting disruptions in connectivity, we should point out that node mobility has been shown to increase capacity of *connected* wireless networks [26]. Thus, DTN routing approaches can be employed in *connected* networks to harness node mobility for capacity reasons. Additionally, it is important to note that we are only targeting applications which disseminate data in a point-to-point manner. Multicast or broadcast applications require different routing strategies; however, we argue that insight from this work is also relevant for multipoint data dissemination services.

2.1 Routing as Opportunistic Forwarding

As previously pointed out, traditional routing protocols (including MANET routing) do not work well in environments prone to frequent and long-lived disruptions; these routing protocols assume almost always connected network and require an end-to-end path to exist in order for a source to send data to a destination. Paths are discovered either in a proactive (table-driven routing) or reactive (on-demand routing) man-

ner. This is not the case in a DTN-like environment, as it is possible that a path may never be available between source-destination pairs. Hence, the store-carry-and-forward routing paradigm is utilized in such scenarios; this means that a set of *independent, opportunistic* (i.e., no certainty about whether there will ever be a path to destination) *forwarding decisions* will attempt to *eventually* deliver messages to destinations.

In the remaining of this section, we define opportunistic routing based on the evolution of the message vectors at nodes as they encounter other nodes. It is important to note that as energy is a precious resource in mobile nodes, any node can turn to *sleep* mode to conserve battery lifetime. Thus, it is possible that two nodes are within communication range of each other but are unable to exchange any information, if any of them is in *sleep* mode. For clarity, we define the "encounter of two nodes" for the case when two nodes are within communication range of each other and are in *power on* mode.

Definition: If node A with a set of messages $S_{msg}^{(A)}(t)$ and a set of context information, $S_{ctxt}^{(A)}(t)$ at time t, encounters nodes $B_1, \ldots, B_n$, each with message vectors $S_{msg}^{(i)}(t), i \in [1, n]$ and context information $S_{ctxt}^{(i)}(t), i \in [1, n]$. Then opportunistic routing does the following:

- $S_{msg}^{(i)}(t + \Delta t) = f(S_{msg}^{(A)}(t), S_{msg}^{(1)}(t), \ldots, S_{msg}^{(n)}(t),$ $S_{ctxt}^{(1)}(t), \ldots, S_{ctxt}^{(n)}(t)), \forall i \in \{A, 1, \ldots, n\},$
- $S_{ctxt}^{(i)}(t + \Delta t) = f(S_{ctxt}^{(A)}(t), S_{ctxt}^{(1)}(t), \ldots, S_{ctxt}^{(n)}(t)), \forall i \in \{A, 1, \ldots, n\},$

where Δt is a random variable and is the time it takes to forward a message (medium access, transmission and propagation delay, etc.), and $f(.)$ denotes a function that will be applied to the message– and context vectors at the time of the encounter. The function $f(.)$ will depend on the type of routing primitive, e.g., replication, forwarding, etc.

We use the same notation to define three basic building blocks[1] of mobility-assisted opportunistic routing, namely *replication, forwarding,* and *coding,* based upon which, every opportunistic routing protocol can be constructed.

Next, we look into these three primitives in more detail, providing also specific examples. Let us assume that a node A which has a set of neighbors B_j encounters node $B_i, j \neq i$. A has then to decide whether to forward message m to B_i.

---

[1] We will use the terms building blocks and primitives interchangeably throughout the paper.

2.2 Message Replication

A relay A carrying a copy of m can decide to spawn a new copy of m and forward it to a newly encountered node, (B). This decision will depend on the message vectors of the two nodes (e.g., if the new neighbor does not have a copy of the message in question) as well as on the "context" of the two nodes (e.g., the new neighbor tends to see the message destination often). In other words, if nodes have infinite buffer space and if $m \notin S_{msg}^{(B)}(t)$, then

$$S_{msg}^{(B)}(t + \Delta t) = S_{msg}^{(B)}(t) \cup f_{rep}(S_{ctxt}^{(A)}(t), S_{ctxt}^{(B)}(t)),$$
$$S_{msg}^{(A)}(t + \Delta t) = S_{msg}^{(A)}(t),$$

where $f_{rep}(\cdot)$ is either $\{m\}$ or $\{\emptyset\}$ (the empty set). Several studies such as [27, 28, 23] have reported the benefits of replication for DTN routing. Note that in case where more than two nodes encounter each other at the same time, $f_{rep}(\cdot)$ would contain context information of all the nodes that meet each other at that time.

2.2.1 Greedy Replication

The simplest version of copy replication is performed in a "greedy" manner. When node A encounters any node, say B, and B does not have a copy of m, A will spawn and forward a copy of m to B; that is, $f_{rep}(S_{ctxt}^{(A)}(t), S_{ctxt}^{(B)}(t)) = \{m\}$:

If nodes have infinite buffer space and if $m \notin S_{msg}^{(B)}(t)$ then

$$S_{msg}^{(B)}(t + \Delta t) = S_{msg}^{(B)}(t) \cup \{m\},$$
$$S_{msg}^{(A)}(t + \Delta t) = S_{msg}^{(A)}(t).$$

This is a fast and robust method to distribute copies, creating a number of "copy custodians" that will look for the destination concurrently. Greedy replication is the basic primitive used by epidemic routing [6]. Epidemic routing has many variants and has been used by researchers as a baseline to evaluate DTN routing protocols, as it offers minimum average message delay at the cost of consuming maximum network resources. Prioritized Epidemic Routing (PREP) [54] is a recent greedy replication based protocol, where the stored bundles are prioritized based upon their expiry time and distance to destination in order to better utilize resources.

Generating and passing a new copy to *every* node encountered may produce considerably high overhead in terms of buffer space for storage and energy spent on transmission and reception. Variants of replication that control the number of copies or custodians of a message circulating in the network at any given point are quite effective in reducing overhead and still achieving adequate performance. They are described below.

2.2.2 Controlled Replication

Here, there is some "context" associated with each given message m. This context keeps track of the number of copies that have been created for m. If the perceived number of generated copies is smaller than some desired value L, then $f_{rep}(m, S_{ctxt}^{(A)}(t)) = \{m\}$. Otherwise, $f_{rep}(m, S_{ctxt}^{(A)}(t)) = \{\emptyset\}$. Below are some examples of controlled replication strategies:

- In *copy-limited replication*, each message copy generated is accompanied by a number of forwarding tokens ($fwd(m) \geq 1$). This number indicates how many extra copies of the message the new node can further create itself and replicate.

$$fwd(m) > 1 \Rightarrow S_{msg}^{(B)}(t + \Delta t) = S_{msg}^{(B)}(t) \cup \{m\},$$
$$fwd(m) = 1 \Rightarrow S_{msg}^{(B)}(t + \Delta t) = S_{msg}^{(B)}(t).$$

- In *time-limited replication*, each new message generated (say at time T_s) may be further replicated to nodes other than the destination, only for an amount of time T_{rep}. If t is the time a node B is encountered and B is not the message destination, then

$$t \leq T_s + T_{rep} \Rightarrow S_{msg}^{(B)}(t + \Delta t) = S_{msg}^{(B)}(t) \cup \{m\},$$
$$t > T_s + T_{rep} \Rightarrow S_{msg}^{(B)}(t + \Delta t) = S_{msg}^{(B)}(t).$$

- In *probability-limited replication* [24], a node decides to forward a copy of a message to any node it encounters with a specific probability p_i, where i indicates the service class to which the message belongs.

Controlled replication has been shown to achieve competitive delays with only a small fraction of the copies used by uncontrolled replication policies such as epidemic routing [6]. It is the strategy used in protocols like Spray and Wait [27, 23], more specifically the copy-limited version.

Controlled replication performs especially well when nodes are homogeneous and move frequently around the network. However, if candidate relays have very different capabilities, greedy– and even controlled replication may waste valuable message copies by forwarding them to nodes that are of little use in the delivery process. In heterogeneous scenarios, one may want to consider the capabilities, characteristics or context of candidate relays and hand over a copy of a message only if the perceived "utility" of a node as a copy custodian is high enough.

2.2.3 Utility-Based Replication

Here, the forwarding decision depends on the context of the current custodian and that of the candidate relay. Specifically, we assume that a set of parameters related to the nodes in question are evaluated to estimate the nodes' "utility" or "fitness" as a relay for a given message bound to a certain destination. This utility may correspond, for example, to the probability of the new node encountering the destination in the future. This and other utility functions will be discussed in detail in Section 3).

There are basically two variants of utility-based replication, namely *uncontrolled* and *controlled* utility-based replication, both of which are described below using our message vector notation:

- *Uncontrolled utility-based replication:* If $m \notin S_{msg}^{(B)}(t)$ AND $f_{rep}(S_{ctxt}^{(A)}(t), S_{ctxt}^{(B)}(t)) = \{m\} \Rightarrow S_{msg}^{(B)}(t + \Delta t) = S_{msg}^{(B)}(t) \cup \{m\}$.
- *Controlled utility-based replication:* If $m \notin S_{msg}^{(B)}(t)$ AND $f_{rep}(S_{ctxt}^{(A)}(t), S_{ctxt}^{(B)}(t)) = \{m\}$ AND $fwd(m) > 1 \Rightarrow S_{msg}^{(B)}(t + \Delta t) = S_{msg}^{(B)}(t) \cup \{m\}$.

Uncontrolled Utility-based replication has been used to reduce the overhead of epidemic routing [20, 14]. As an example, rather than handing over a copy to every new node encountered, each node maintains a probability measure of future encounters using the history of past encounters; based on this probability, a node forwards a new copy to a new neighbor only if the neighbor has a high enough (or higher than the current relay's) probability of a future encounter with the destination.

Controlled, utility-based replication, on the other hand, has been proposed in [10] to improve the quality of forwarding decisions made by Spray and Wait in heterogeneous environments. Encounter-Based Routing (EBR) [53] is another example of controlled, utility-based replication, in which future rate of node encounters is predicted using number of past encounters with nodes, and encounter metric is computed locally at each node. The number of replicas of a message, delivered to a relay node depends upon the ratio of encounter value that the relay advertises.

2.3 Message Forwarding

Unlike replication, under copy forwarding, a relay A carrying a message m may decide to hand that message over to a node B it encounters; by doing so, A relinquishes its copy of m and ceases to be one of its custodians. Clearly, forwarding incurs minimal message duplication overhead. It is beneficial when the initial

relay(s) chosen is(are) not the best one(s). Using our message vector evolution notation, we can define forwarding as follows. If $m \notin S_{msg}^{(B)}(t)$ then

$$S_{msg}^{(B)}(t + \Delta t) = S_{msg}^{(B)}(t) \cup f_{rep}(S_{ctxt}^{(A)}(t), S_{ctxt}^{(B)}(t)),$$
$$S_{msg}^{(A)}(t + \Delta t) = S_{msg}^{(A)}(t) - f_{fwd}(S_{ctxt}^{(A)}(t), S_{ctxt}^{(B)}(t)),$$

where $f_{rep}(\cdot)$ and $f_{fwd}(\cdot)$ take values either $\{m\}$ or $\{\emptyset\}$ (the empty set).

Forwarding a message can be performed either using a utility function or in a probabilistic manner (e.g., tossing a coin to decide, at each contact, if a message should be forwarded or not). If a utility function approach is used, each node i maintains a value for the utility function $U_i(j)$ for every other node j in the network. $U_i(j)$ which can be interpreted as the probability that node i will forward a message to node j, may be based on a number of different parameters (e.g., encounter history, mobility, friendship index with j, etc.). In general, $U_i(d)$ is a function of the context $S_{ctxt}^{(i)}(t)$ of node i, and possibly of that of node d, the destination, $S_{ctxt}^{(d)}(t)$. That is, $U_i(d) = g(S_{ctxt}^{(i)}(t), S_{ctxt}^{(d)}(t))$. If a node i carrying a message copy for a destination d encounters a node j with no copy of the message, then

- **Rule 1: Absolute utility criterion** If $U_j(d) > U_{th}$ for some U_{th} threshold value OR
- **Rule 2: Relative utility criterion** If $U_j(d) > U_i(d)$ (*relative utility criterion*), then

$$S_{msg}^{(B)}(t + \Delta t) = S_{msg}^{(B)}(t) \cup \{m\}$$
$$S_{msg}^{(A)}(t + \Delta t) = S_{msg}^{(A)}(t) - \{m\}$$

Scale Free Routing (SFR) [52] is an example of a routing protocol that is based on message forwarding, where single copy per message is used, and there is no replication. Forwarding is based upon some utility function, but if the utility function is lower than a certain threshold, the nodes with the highest mobility, and so can move the farthest in the network are chosen as relays and message is forwarded to these relay, which are called Ballistic Nodes. This protocol is based upon the concept of Levy Walks.

2.4 Message Coding

Messages may be coded and processed at the source, i.e., *source coding* or as they traverse the network, i.e., *network coding.* In the following subsections, both of these coding variants are presented.

Source Coding: Source coding aims at increasing delivery reliability and reducing worst-case delay. A notable example is *erasure coding* [29], in which the coding

is performed by the source, a coded part of a message is further treated as any other message in the network, and there is no specific implications on routing and forwarding.

A variation of source coding known as *distributed source coding* tries to minimize propagating redundant information in the network, and thus reduce overhead. Sensor networks, which are aimed at a variety of monitoring applications (e.g., environmental and habitat monitoring), are the typical target scenario for distributed source coding [30]. The basic idea behind distributed source coding is to take advantage of the data's inherent spatial and temporal locality to suppress propagation of unnecessary information. For example, in a sensor network tasked to measure the temperature field of a given region, nodes that are in close proximity to one another are expected to report similar temperature values. Through DSC strategies, nodes can identify such redundancies and perform *in-network aggregation* to reduce the volume of data transmitted in the network [31]. Another example of DSC is growth codes [32], which use coding redundancy at neighbors to avoid the impact of loss.

Network Coding: Network coding has been proposed as a way to increase the capacity of wireless network [33], [11]. The main idea behind network coding is to allow mixing of messages at intermediate nodes in the network. In this way, a receiver reconstructs original message, once it receives enough encoded messages. The network coding is shown to achieve maximum information flow in a network, which is not attainable with traditional routing schemes.

Linear network coding has been shown to achieve the capacity of information networks [34]. This coding scheme permits a node to apply a linear transformation to a vector (a block of messages over a certain base field) before passing it further in the network. It can be used to reduce the time to deliver a given flow, maximize the throughput, reduce the number of transmissions (and thus energy expended), etc.

Random network coding, where coding coefficients are chosen by each node randomly from a large enough field (often Z^8), and in a distributed manner, is an efficient method to implement network coding in practice (coding coefficients are sent as part of the packet, with only a small overhead) [35]. To take advantage of the benefits of network coding in a wireless, often "challenged", environment, the following modification of greedy replication have been proposed [33]: instead of transmitting single packets, linear combinations of packets are generated and transmitted; assume a node A has a set of linear combinations of N packets $S_{msg}^{(A)} =$

$\{\hat{m}_1, \hat{m}_2, \ldots, \hat{m}_m\}$ and encounters another node B. Then, it creates a linear combination of all its messages in the queue

$$\hat{m}_{new} = \sum_{i=1}^{m} c_i \hat{m}_i. \qquad (1)$$

Here, the addition is "modulo" the given base field chosen for network coding.

Finally, depending on the context of nodes A and B, $f_{code}(S_{ctxt}^{(A)}(t), S_{ctxt}^{(B)}(t)) = \{\hat{m}_{new}\}$ or $\{\emptyset\}$, and

$$S_{msg}^{(B)}(t + \Delta t) = S_{msg}^{(B)}(t) \cup f_{code}(S_{ctxt}^{(A)}(t), S_{ctxt}^{(B)}(t)). \qquad (2)$$

When enough independent combinations ($\geq N$) of the N messages, belonging to a given coding generation, have been received, a node can *decode* them to get the original N messages. Finally, the forwarding function $f_{code}(\cdot)$ might be for example:

- a random coin toss, i.e. $f_{code}(S_{ctxt}^{(A)}(t), S_{ctxt}^{(B)}(t)) = \{\hat{m}_{new}\}$ with some probability $p \leq 1$ [33].
- based on a utility function as described in Section 3.

One key problem with the network coding approach described above is that coding *every* single message together may result in never collecting enough independent combinations of messages to successfully decode, especially when the network in sparse or when the nodes' degree is low. Some control is needed on how many and which messages will be coded together. This is known as generation control. Coding messages from many different sessions and from large time or sequence number windows (large generations) might result in high delivery delays. On the other hand, using small generations limits the amount of gains achievable by network coding. Finally, even controlling the generations in a distributed manner, might pose significant challenges.

For these reasons, it has been suggested to implement network coding hop-by-hop, in an *opportunistic* fashion [11]. Assume that a node A with message vector $S_{msg}^{(A)}$ encounters a set of nodes $B_i, \ldots, B_n$ with message vectors $S_{msg}^{(B_1)}, \ldots, S_{msg}^{(B_n)}$. Let us further define the n sets $S_i^{(A)}$, such that

$$S_i^{(A)} = S_{msg}^{(A)} \cap \overline{S_{msg}^{(B_i)}}, i \in [1, n]. \qquad (3)$$

In other words, $S_i^{(A)}$ is the subset of A's messages that neighbor B_i does not have. Then, opportunistic network coding looks for a combination of messages in $\cup_i S_i^{(A)}$ that will result in maximizing the number of neighbor nodes, B_1 to B_n, that will be able to decode a new packet. A then *broadcasts* this message combination. Opportunistic network coding simply takes advantage of favorable traffic patterns to locally save some

transmissions, without requiring any generation control or imposing additional delays, but its performance still suffers in very sparse networks.

2.5 Routing as Resource Allocation

In this subsection, we look into DTN routing from a resource allocation point of view. In traditional DTN routing, routing is mostly performed based upon some utility function(s). The main aim is always to find a path to a destination with the available information. Almost all routing strategies are no exception to this, and thus they have an incidental effect on routing metrics (maximizing average delay or delivery ratio). Another angle to look at DTN routing is to treat it as a resource allocation problem. The purpose is to have an intentional effect on the DTN routing, rather than an incidental one, in order to maximize the performance of specific routing metrics. The idea is to forward or replicate a message to a relay, based upon the available resources in order to maximum the likelihood of message delivery, when two nodes meet. Note that resource allocation based routing is not a basic primitive of DTN routing, and can use any of the three basic primitives described in the previous subsections.

RAPID [46], [47] is the first protocol which treats DTN routing as a resource allocation problem. In RAPID, messages are ordered with respect to their utilities, keeping in view the goal of maximizing specific metrics (e.g. delay), which also allows computation of more sophisticated and desired metrics such as worst-case delivery delay and packet delivery ratio. The protocol translates a routing metric to per-packet utilities, and at every transfer opportunity, it is verified if the marginal utility of replication justifies the resources used. In a way, it is a replication-based protocol, but what differs it with the traditional replication scheme is resource allocation.

Erramilli et al. [55] has done a study that is based upon prioritizing messages to better manage network resources in a resource-constrained environment, where they use delegation forwarding [58] as their forwarding algorithm. Another protocol that is based upon the resource allocation concept is ORWAR (Opportunistic Routing with Window-Aware Replication) [56] that uses message utility based differentiation mechanism which allows allocation of more resources for messages with high utilities. Thus, it replicates messages in order of high utilities first, and removes messages in the reverse order, if needed. Again, this is a replication routing scheme, but the delivery of number of copies depends upon evaluation of the contact window.

2.6 Examples of DTN Routing Protocols

In Section 2, we have described three basic primitives based on which DTN routing can be built. We now proceed to identify the use of these primitives in some existing DTN routing protocols. Table 1 summarizes this correspondence between DTN building blocks, their variants and existing DTN solutions. The table shows examples of DTN-routing protocols and categorizes them in terms of the three main building blocks (i.e., replication, forwarding and coding). The first column represents the properties based on which the routing protocols are built, and the second column shows the routing protocol examples.

Take for example Epidemic Routing [6]: it is a typical case of "uncontrolled", i.e., with no constraints on the number of copies generated, message replication using a greedy approach; on the other hand, Spray and Wait [27] is an example of "controlled" greedy replication as it limits the number of copies for each message. Replication can also be made "smart" by using some utility functions as in [10]. Spray and Focus [45] is an example of a protocol that combines greedy replication with smart forwarding mechanisms. Performance and efficiency can further be improved if smart forwarding is used with smart replication. On the other hand, smart forwarding mechanisms can be used with source coding schemes such as Erasure Coding [29], and replication can be used with coding schemes [11], [32].

3 DTN Routing Utility Functions

We now turn our attention to utility functions that can be used in message replication (or forwarding) by the DTN routing primitives previously discussed. Candidate utility functions could be broadly categorized into *destination dependent* ("DD") and *destination independent* ("DI") functions. These utility function are very useful especially when the network as well as the participating nodes are heterogeneous. Many utility functions have been presented in [10], and are thoroughly investigated and applied to heterogeneous environments in [36] and [37].

3.1 Destination Dependent (DD) Utility

One node may be the best relay for one destination (d_1), and another node may be the best relay for a different destination (d_2). In other words, for DD utility functions, it is possible that the following is true:

$$U_i(d_1) > U_j(d_1) \text{ but } U_i(d_2) < U_j(d_2), d_1 \neq d_2. \qquad (4)$$

Table 1 DTN Routing primitives and their use by existing DTN routing protocols

	Forwarding	Replication	Coding
Greedy		Epidemic [6] PREP [54]	
Controlled		Spray and wait [27] SWIM [23]	
Utility Based	FRESH [38] Scale-free [52] Spray and Focus [45]	History-based Epidemic [14] Probabilistic flooding (Prophet) [20] Smart Replication [10] MV Routing [51] Encounter-based [53]	
Resource-allocation		RAPID [46], [47] ORWAR [56]	
Mobility Characteristics	Mobyspace [19], [42] Solar [15] Scale-free [52]	Maxprop [25]	
Routing Table Entry	Island hopping [12]		
Network (end-to-end)			LeBoudec [33]
Opportunistic			COPE [11]
Distributed source coding			Growth codes [32]

Below we describe a number of parameters that can be used to build destination dependent utility functions.

- **Age of Last Encounter:** It has been suggested that keeping track of past encounters with a given node can be helpful in successfully predicting future encounters. For example, each node could maintain a timer for every other node in the network that records the time elapsed since the two nodes last "saw" each other [38]. These timers could then act as indirect location information. Additionally, a node can keep a record of its encounters with another node by noting the last encounter time and the node's position at the time of encounter [39]. Although keeping the last encounter time for nodes does not provide any guarantee that a node would meet a destination in the future, yet it can be useful in predicting the current location of a destination. Because, nodes tend to move in a continuous manner (i.e., they don't ordinarily perform jumps in space), often, a smaller timer value implies a smaller distance to the destination, if we assume that the average speed of nodes does not vary too much. In case nodes are heterogeneous in terms of their characteristics and capabilities, some other parameters should be used in combination with age of last encounter in order to choose a "suitable" relay node. Note that the age of last encounter with a destination is related to the *instantaneous* fitness of a node as a candidate relay for that destination.

- **History of Past Encounters:** The age of last encounter is only a single "snapshot" of the history of past encounters and may not necessarily predict future encounters successfully. Instead, a node could maintain a "richer" set of information about past encounters with another node, like *frequency of encounters, average inter-encounter time, higher moments of inter-encounter time, average encounter duration*, etc. Such information could help identify more accurately good candidate next hops; on the other hand, keeping more information about encounters increases the overhead in terms of context data that needs to be stored. Also, depending upon the application requirements, a combination of past encounter parameters can be used to choose the best possible relay for a destination. Another consideration is how long to keep this history about a certain destination at a node as it may not be useful, or even misleading after a certain threshold of time depending upon the dynamics and mobility pattern of participating nodes. An example of this kind of utility function is Encounter Based Routing (EBR) [53], in which future rate of node encounter is predicted using information about past encounters with node.

- **Pattern of Locations Visited:** In the real world, mobile users move with certain purposes in mind (e.g., going to work, going to a class, going from work to lunch, etc.). Additionally, they may follow specific paths in between these locations due to geographical constraints. As a result, people tend to follow a *movement pattern* in their daily activities. These patterns are a function of a variety of parameters including professional activity, work and home location, etc. What is more, most people also tend to spend the majority of their time in a small subset of *preferred* locations, as opposed to indiscrim-

inately roaming everywhere (unless, this is part of their job, e.g., taxi driver, salesman, etc). "Location preference" as well as the periodic nature of human mobility (diurnal and weekly patterns) have been consistently demonstrated in a variety of real mobility traces [13]. Mobility patterns (known a priori or "learned" online by collecting appropriate statistics) could help identify a *profile* for a given node; nodes with a matching or similar mobility profile as the destination could be considered good candidate relays for messages to that destination [19], [42], [15].

– ***Social Networks*:** Humans are involved in complex social relationships (networks), and people who are socially-related to each other (e.g. friends, students in the same class, and colleagues in the same department) are expected to interact more often with each other. These social features can have important implications for networks formed by communication devices operated or carried by humans (e.g., vehicles, PDAs, laptops). Knowledge about existing social links could allow one to choose a "data relay" that has a much better chance of encountering the destination soon. Note that one way to gather information about social networks is by keeping a history of past encounters. However, there is additional data that is relevant in the context of social networks. For example, suppose that it is known a priori that A is a good friend of D, but B hardly knows D; then, even with no past encounter information of D at A or B, A can be considered a *better* relay for D than B. Another way to do this is by labeling the nodes with community names and by making nodes advertised the communities they belong to as they move and meet other nodes. The social network information about nodes can also be gathered by observing and estimating their mobility pattern.

Bubble [57] is one of the recent social-based forwarding protocol, in which forwarding is based upon identifying "hubs" and "centrality points" in the network. Having no information about a destination, a message is forwarded towards a more "popular" area or node, and then the forwarding mechanism tries to find the destination itself, or a node having the same "community" as the destination node. The logic behind finding a popular node first is that in a social network, some nodes tend to see other nodes more often than others.

– ***Traditional Routing Table Entry*:** In a network that is often disconnected, it is possible to have net-

work connectivity in parts of the network (connectivity islands). So, in such cases, each node could maintain a limited-range (e.g. n-hop) view of the topology in a proactive manner (link-state, distance vector) to improve performance. In many scenarios, complementing traditional routing mechanisms with "mobility-assisted" primitives to overcome partitions or other route failures may be a more suitable solution than replacing traditional routing altogether.

3.2 Destination Independent (DI) Utility

In this case, the "utility" of a given node is independent of any destination; rather, it depends on some characteristic(s) exhibited by a node. This implies that one node may be the best relay for most or all destinations. In other words, for DI functions it holds in general that:

$$U_i(d_1) \geq U_j(d_1) \Rightarrow U_i(d) \geq U_j(d), \text{for most or all } j, d. (5)$$

Examples of nodes which are highly preferable as relays for any destination could be nodes with high and frequent mobility (e.g., vehicles), nodes with many "friends" (e.g., *hubs* in scale-free networks), nodes with more resources (e.g., buses [25]), or nodes with high cooperative behavior (e.g., APs, routers or gateways, ferries). Below, we describe in more detail some destination independent parameters that should be considered when making forwarding decisions.

– ***Amount of Mobility*:** In some wireless network deployments, nodes may vary in different ways, e.g., some might be more mobile than others. In the case of a campus environment, nodes carried by humans may tend to be more static, while nodes attached to campus transportation vehicles (e.g., [25]) move around the campus periodically, some of which following regular trajectories. These more mobile nodes tend to traverse a wider portion of the network in the same amount of time than the more static nodes, and thus encounter a larger subset of other wireless nodes. As a result, they represent highly desirable relays, if a DTN-like routing strategy is employed. One way to identify such relays could be, for example, to use *labels* that represent the type of mobility exhibited by nodes, e.g. "BUS","TAXI", "PEDESTRIAN", "BASE STATION", etc. In some scenarios, it would not be too burdensome to manually configure a label (e.g., by setting some software parameter when installing a radio, say, on the top of a bus). Nevertheless, algorithms that estimate the "degree of mobility" *online* could also be deployed in self-organized, more dynamic environments [10].

– **Node Resources**: When forwarding a message to a node, the resources and capabilities of that node should be considered. Even if a certain node has some ties to the destination (e.g., close friendship), giving a message copy to that node might be a waste of resources, if it is almost out of battery. Chances are it will either turn itself off or run out of battery before it gets a chance of delivering the message. Similarly, if a candidate relay has its buffer almost full, it might be more prudent to prefer another node instead. This may not only result in smaller queuing delays, but may also reduce the probability of the message getting dropped later. Consequently, nodes may maintain the current status of their resources, which can be used to identify nodes that are "good" (or "bad") relays independent of the destination.

– **Cooperative Behavior**: Message forwarding is not free and consumes node resources including battery life and buffer space. So, it is possible that some nodes refuse to forward messages on behalf of others because either they have limited resources, or they are pre-configured with specific forwarding policies, or because they have been either compromised or are owned by an attacker. So, forwarding a message to such nodes would be disadvantageous. Consequently, forwarding decisions should also consider how cooperative nodes are in forwarding messages. Approaches to boosting cooperation among nodes include offering incentives to cooperating nodes, or penalizing non-cooperative ones. This has also implications in building trust among participating nodes, which is the topic of the DI parameter discussed below.

– **Trustworthiness**: Although a number of research efforts have been devoted to addressing various problems related to data delivery in wireless networks (e.g., media access, routing, and transport protocols), securing wireless communication is among the biggest challenges. This is due to a number of factors notably the shared, uncoordinated access to the wireless medium, as well as its inherent unreliability and non-determinism. The peer-to-peer, non-hierarchical nature of many emerging wireless applications requires collaboration among participating nodes so that data delivery can be accomplished. Malicious peers could exploit this to intervene with the network's normal operation or extract sensitive information, such as passwords, credit card numbers, etc., from packet streams. In other cases, malicious users could pretend to carry and forward other nodes' traffic, while in fact, they don't do so,

which may create drastic forwarding problem. What is more, wireless node resources like bandwidth and battery power will be scarce and valuable in the foreseeable future. Thus, non-malicious yet selfish users might be tempted to refuse carrying other's traffic. For these reasons, the utility of a node as a message relay might also be a function of the trust other nodes have in it, a trust which could be based on signed certificates, PGP-like architectures [43], reputation systems [44], etc.

3.3 Additional Considerations

It is certainly possible (and probably desirable) to define utility functions that take into account both the general, destination independent *fitness* of a node as well as destination specific information. For example, we can combine history of past encounters (DD utility) with nodes' mobility patterns, or their resources (DI utility) in order to define a hybrid utility function that is able to deliver messages to destinations more efficiently.

Most utility functions discussed above are based solely on a snapshot of the past (e.g., the last time node X encountered node Y). However, in real life scenarios node interactions may exhibit rich and intricate structure; it would thus be beneficial to explore learning techniques that try to use history over a window of time or feedback (e.g., from the destination) to make better routing decisions.

4 A Taxonomy of DTNs

In this section, we classify DTNs according to a set of characteristics relevant to routing. For example, a well-connected network whose nodes exhibit little or no mobility would imply that traditional MANET routing algorithms (e.g. OLSR [1], AODV [2], etc.) might be appropriate. Similarly, a network where nodes have little or no energy limitations (e.g., vehicles) would likely render routing protocols that focus on minimizing energy consumption inadequate. We start by describing the network features used in our DTN taxonomy.

4.1 Connectivity

Connectivity is an important characteristic of wireless networks. Two well-known definitions of network connectivity are (i) the probability that a path exists between two randomly chosen nodes [7], or (ii) the percentage of nodes connected to the largest connected

component [7]. Although these two definitions are slightly different, they have similar implications from a macroscopic point of view.

Traditional routing techniques assume the "Internet model" where networks are always connected. Partitions are treated as faults and routing attempts to mend them as soon as they are detected. Typically, alternate routes can be found and disconnections, if they happen, are ephemeral events. In multi-hop wireless ad hoc networks, or MANETs, due to node mobility, wireless channel impairments, limited node capabilities, etc, the assumption that the network is always connected no longer holds and routing had to be re-thought. However, partitions are still considered exceptions to normal operation and routing reacts by trying to find alternate paths. In case it fails and disconnections persist, data queued at nodes waiting to be forwarded starts to get dropped as queues fill up. In fact, it is well-known that the so-called reactive (or on-demand) routing protocols such as DSR [2] and AODV [2] perform poorly when disconnections are frequent and persist for arbitrarily long periods of time.

Recently, it has been recognized that in disruption tolerant networks, connectivity will be consistently below 1 (or 100%). As a result, the whole spectrum of possible connectivity values all the way from 0 (very sparse networks) to 1 (connected networks) need to be considered when designing routing algorithms.

It is well-known from percolation theory that, in networks consisting of randomly placed (or randomly moving) nodes, connectivity exhibits a *phase transition* behavior [8] as depicted in Fig. 1. Specifically, if connectivity is scaled by changing the nodes' transmission range, then the following can be observed [9]: (i) for (a large number of) low transmission range values, connectivity values are quite low: no large cluster exists, but rather very small clusters (few with 1 node), whose sizes are exponentially distributed, are found; (ii) when transmission range crosses some threshold value, connectivity starts increasing rapidly and quickly enters a region where a giant component is formed containing a large percentage of nodes, while the rest of the nodes form smaller clusters (again of exponentially distributed size).

This phase transition behavior has some important implications: *random networks*, i.e., those formed by randomly placing nodes (e.g., sensors scattered uniformly in the field) or randomly moving nodes (e.g., random direction), will be either *sparse* or *almost connected*, in most cases. But, if transmission range or number of nodes is low, we can have the case where nodes tend to form clusters (or connectivity islands) due to their mobility patterns. So, in the following, we focus
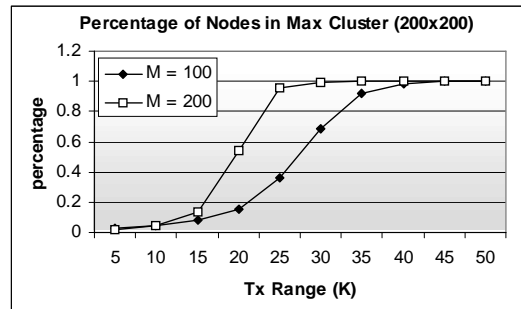

Fig. 1 Expected percentage of total nodes in largest connected component, as a function of the number of nodes (M) and transmission range (K) (200×200 grid).

on three different kinds of networks according to their connectivity, namely: *almost connected networks*, *sparse networks*, and *connectivity islands*.

Almost connected networks: Also known as "flaky nets", these networks more closely resemble the traditional MANET viewpoint of a connected graph. However, the graph here often exhibits partitions. A good percentage of end-to-end pairs are connected at any time, even though the paths might not be long-lasting. Traditional proactive– (e.g., link-state) or reactive routing protocols (e.g. DSR, AODV) could still deliver a part of the traffic successfully (although with a higher overhead for route discovery and maintenance). Yet, they are unable to deliver any traffic between nodes that lie in different partitions.

Mobility-assisted routing schemes can be beneficial in bridging disconnected parts of the network and are able to deliver traffic between any two nodes. Yet, hybrid protocols that can also take advantage of the existence of large connected clusters are desirable.

Sparse networks: This is a more challenging scenario. In these networks, transmission range is much lower and no large clusters exist. Most nodes have only a few neighbors or are isolated most of the time. Every now and then, two such nodes come into contact, at which time they can exchange data or other useful information, and soon go back to having no neighbors. It is evident that traditional– or even MANET routing protocols would fail to satisfy most end-to-end traffic requests, as very few contemporaneous paths exist. What is more, the small size or non-existence of clusters imply that routing modules that aim at maintaining multi-hop neighborhood information (2-hop, k-hop, etc.) have not much value to offer.

Instead, a message has to get routed predominantly by being carried using relays. Occasionally a new candidate relay is encountered and the routing protocol needs

to decide whether it should hand-over custody, replicate some of its messages, or continue carrying them. Consequently, node mobility is a crucial feature in these sparse networks, both in terms of how mobile nodes are, as well as how structured node mobility is (i.e., whether mobility patterns exist). Similar to network connectivity, mobility is another important feature and will be discussed in detail in Section 4.2 below.

It is thus important to *discover nodes that move frequently and quickly around the network* as well as *nodes whose mobility pattern might be correlated with that of the destination.* To do so, nodes may exchange useful information about themselves or other nodes encountered recently. If such information can be collected often enough (before it becomes irrelevant or obsolete), mobility-assisted routing policies can be used to deliver close-to-optimal performance.

Another important implication of sparse networks is that whenever two nodes encounter each other, there is only a small probability that other nodes are also within range. As a result, there is little contention, on average, at the MAC layer for each transmission, and there is also little (in-channel) interference. This suggests that available bandwidth (or buffer space) per contact is the limiting factor as far as performance is concerned. What is more, it suggests that forwarding or scheduling techniques that aim to choose the right neighbor (e.g., transmit to the "best" neighbor according to some utility function) [10] or combine packets for different neighbors (e.g. opportunistic network coding [11]) offer little gain here.

"Connectivity Islands": It has been observed that in real world deployments, node location does not typically follow a uniform distribution. Similarly, node mobility is usually non-uniform. In fact, it is often the non-uniform mobility process that creates the non-uniform node location distribution. Thus, even though the phase transition phenomenon described earlier might imply that networks are either sparse or almost connected, in real world different connectivity structures might be observed. For example, in vehicular networks nodes may tend to gather around different concentration points for reasons dependent on the transportation network (e.g., traffic lights, junctions, toll, etc.) or application (e.g., taxi booths at airports, popular locations, etc.) [12]. Other real world examples include *First Mile Solutions* [40] and *VLINK* [41].

This non-uniform placement or mobility of nodes can also be observed in a variety of other scenarios. Consider, for example, a campus with people mostly moving within their own departments [13], or herds of animals mostly moving together in packs [14]. These networks

can be seen as a set of separated islands of (full) connectivity, formed around a concentration point, with few or no contemporary paths between concentration points.

Connectivity Islands lie in between *Almost connected–* and *Sparse Networks.* On one hand, their sizable clusters imply that proactive routing approaches could help collect and maintain useful information about *immediately reachable* nodes. On the other hand, a large number of nodes outside the local cluster are not immediately reachable using traditional techniques. Instead, mobility-assisted routing should be used to move messages between different "islands", where no immediate path is available. Consider, for example, a scenario where some *anchor nodes* are stable over time and can serve as "connectivity points" (e.g., VANET concentration points at traffic lights are expected not to change often), but *attached nodes* change often. In these cases, routing can be done hierarchically where at the macroscopic level, relatively stable paths can be constructed and used to route traffic between "islands", while store-carry-and-forward is used on a microscopic level to forward messages when no routes exist, likely between "islands" [12]. What is more, if the nodes that are associated with a given concentration point are stable over time (e.g. nodes affiliated with a given department), macroscopic information about the mobility pattern [15] or community structure [16] between nodes could be used to route traffic across disconnected parts.

To summarize, if a routing table entry exists for a given destination on the *microscopic level* (i.e., populated by traditional routing techniques, such as proactive link-state (e.g., OLSR) or on-demand distance vector (e.g., AODV)), then no special measures are needed. If, however, no paths exist to that node, a routing entry can indicate a possible course of action on the *macroscopic level*, e.g., "send to connectivity island X". This latter action could be performed by, say, finding a node that is affiliated with X [15] or replicated or sprayed to a number of nodes, with the hope that one of them will soon visit X.

4.2 Mobility

Node mobility is another important factor to be considered when choosing adequate routing approaches, especially as the network becomes sparser. In particular, we will discuss two aspects related to node mobility as follows:

Amount of Mobility: The "amount of mobility" of a node can be defined as the percentage of the network traversed or "covered" by the node within a given amount of time. Alternately, it can also be expressed as

the number of new nodes (and thus either destinations or candidate relays) a given node encounters within a given time window. The following characteristics are needed to quantify mobility.

– **Node Speed**: Intuitively, the faster a node is moving, the more new area it should cover in a given amount of time, all other parameters unchanged. Additionally, if nodes move fast, they would have more chances to meet more nodes, thus increasing the number of contacts. On the other hand, if node speed is too high, contact duration is reduced, directly affecting routing protocol performance.

– **Pause Time and Frequency**: Depending upon the environment and the application, mobile nodes may tend to stay at a particular position for extended periods of time. We call this duration as the pause time. For example, in an exposition hall, nodes may move from one place to another and stay at the other place for some time before moving further. Again depending upon the application, the pause time may be used to deliver messages to destinations as it increases the contact duration when the node is in static position, as it has been shown that in some cases, the nodes that are static are more useful to relay messages because of their placement in the area (e.g., throwboxes[17], bus stops etc.). On the other hand, depending upon the scenario, the nodes that have longer pause times may not be as useful in the delivery process as mobile nodes. The nodes' periodicity of visiting places, or their frequency can also be exploited in the delivery process of messages.

– **Integration Time**: This is essentially the time it takes a node, starting at a given state of a mobility structure, to arrive to its stationary distribution; the higher the integration time, the more time it takes the average node to reach a randomly chosen destination.

In general, the larger the amount of *average* node mobility, the better the performance of routing protocols that rely on such mobility. Furthermore, in a number of situations it holds that the higher the average node mobility, the less sophisticated the design of a protocol needs to be. This seems to be in contrast with the traditional viewpoint that node mobility has a negative effect on routing protocol performance.

Structure of Mobility: The structure of the nodes mobility is equally important, and becomes significantly more so for sparser and "less mobile" networks. The following information about the structure of a node's mobility pattern is particularly important from a routing protocol's perspective:

– **Homogeneous vs. Heterogeneous Mobility**: Depending on a particular DTN application, participating nodes may all have the same capabilities and behavior. Conversely, in a heterogeneous deployment, nodes mobility may differ from one another. For example, one could reasonably assume that nodes in a sensor network have homogeneous capabilities and behavior (e.g., duty cycle operation). However, people forming a Pocket Switched Network [18] might have largely different mobility patterns from one another.
Nodes heterogeneous mobility affects protocol design in a number of ways. For example, some nodes will be better relays than others for delivering traffic. Some relays might be preferable for any destination[2], as in the case of nodes that move fast and frequently around the networks (e.g. vehicles). Protocols that are "smart" enough to discover and pick such advantageous relays are expected to perform better the more heterogeneous a network is. Attention is needed though to make sure not to overload a few nodes with relaying responsibilities; this will possibly have detrimental effects due to congestion or battery drainage. Alternatively, if the network is homogeneous, then simple greedy solutions may be adequate to achieve good performance.

– **Spatial and Temporal Correlation**: In addition to differences in the mobility pattern between nodes, individual nodes may exhibit specific mobility patterns which could be leverage to improve routing performance. For instance, a given node may visit some locations (e.g., a person's home or office) often which exemplifies spatial correlation of movement. Also, a given node may exhibit different mobility behaviors depending on the time of day (temporal correlation). For example, most employees might head to the company's cafeteria between $12 - 1$p.m. Finally, there might also exist correlations between the mobility of different nodes both in space (e.g., nodes that tend to visit the same locations [19]) and time (e.g., nodes that leave their "home" location at around the same times). In such cases, good relays may be *destination specific*, that is, a given node may be the best relay to deliver a message to destination X but may never do so for another destination Y. In some other cases, good relays may

---

[2] There are also cases where some nodes are better relays for certain destinations. Destination dependent and destination independent choice of relays is discussed in detail in Section 3.

be *time-specific*, which means that a given node can act as the best relay at a specific time for a destination (or during a specific time interval), and another node would serve as relay for another time interval. Protocols that possess the necessary intelligence to distinguish between relays in general, and more specifically, take advantage of mobility patterns they exhibit, are desirable.

– **Other Considerations:** In addition to the previous generic mobility characteristics, a given set of networked nodes may also exhibit mobility attributes that may result in special structures which should be accounted for by routing. This is the case of *disconnected islands* as discussed in Section 4.1. In several applications, a set of mobile nodes can create well-connected clusters (e.g., a military platoon, a nomadic community [20], wildlife herd or pack [14]) which may be far enough away from one another that they cannot communicate among them. It has been shown that, in these cases, hybrid protocols that take explicit advantage of this structure, using regular routing protocols within a cluster and mobility-assisted techniques to bridge such clusters, can achieve good performance [21], [12].

4.3 Node Resources

Although network and node resources are becoming less and less of an issue in wired networks, it is not typically the case for their wireless counterparts. Depending on the application, node capabilities such as bandwidth, storage, and battery lifetime may vary largely. Resource availability or lack thereof should play an important role in the design and performance of a routing protocol.

– **Bandwidth:** Networks which operate over a common shared wireless medium, the available bandwidth is always a valuable and often scarce resource. If bandwidth is limited, then routing protocols should be efficient, especially in terms of signaling and control information exchange. Furthermore, the more limited the available bandwidth, the more prudent the choice of forwarding opportunities needs to be.

– **Storage:** Sensor networks are the typical case where available memory at nodes might be limited relative to the amount of information that needs to be stored locally. Besides affecting the choice of the routing algorithm to be used, storage limitation also influences relevant routing protocol parameters (e.g., TTL) as well as mechanisms such as buffer replace-

ment policies and garbage collection [22,23]).

– **Battery Lifetime:** Power awareness is usually an important feature in routing protocols for wireless networks[3]. In the case of DTNs, it becomes even more critical, especially in the case of deployments in remote, hard to access regions where nodes may be left unattended for extended periods of time. There is also a recent work [64] that considers making throwboxes energy efficient in order to increase their lifetime while maintaining high efficiency of the system in terms of delivery ratio and latency. In order to minimize the energy waste in DTN, optimal searching or probing intervals are calculated using statistical information of contact opportunities in [59], [60], [61] and energy efficient sleep scheduling mechanisms are constructed in [62], [63].

Heterogeneous Node Capabilities: In addition to different mobility patterns, nodes may also have largely varying capabilities, like battery life, processing power, storage capability, etc. Imagine, for example, a scenario where some of the wireless nodes are vehicles (with little or no energy and storage limitations) while others are small PDAs carried by pedestrians. In such a scenario, it is important for the routing protocol to be able to identify the more capable nodes as they are possibly better candidates for relaying traffic than nodes that have barely enough resources to handle their own traffic.

4.4 Application Requirements

The discussion so far focused on network and individual node features and capabilities. In this section, we consider application-specific requirements, which must be taken into account when choosing or designing DTN routing mechanisms.

– *Message Content and Priority*: Despite the inherent delay tolerance of most DTN driving applications, there can be situations where some messages may be more important than others. For example, in a VANET network it is reasonable to assume that an accident notification message will have higher priority than a chat message, or announcements of nearby shops. In some cases, users might be willing to "pay" more for some of their traffic to get through quickly. Under such heterogeneous traffic requirements, different forwarding policies will be needed to serve the different types of traffic. What is more, not only is it important to ensure that a

---

[3] There are of course some notable exceptions, e.g., VANETs.

given protocol can deliver the desired performance (this is not always the case in such a partitioned environment), but the coexistence of the different protocols must be harmonic, as well.

– *Reliability*: In addition to different priority requirements, some messages may need to be sent reliably. Unlike conventional networks, acknowledging messages end-to-end in partitioned networks is not a trivial task and may often have a significant performance overhead (e.g., flooding an ACK message after successful reception at the destination). Furthermore, if a whole session of messages needs to be sent reliably, the considerably large delays of the *loosely* closed feedback loop may significantly reduce the ability to "pipeline" data through the network. What is more difficult in terms of reliability in a disruption-tolerant kind of network, is the ability to reliably deliver data in a certain order.

5 DTN Routing Design Guidelines

In the previous three sections, we have discussed different properties of DTNs such as connectivity, mobility and node resources, and have dissected DTN-based routing solutions with respect to their characteristics (replication, forwarding and coding). Now, we try to summarize the discussion by providing a correspondence between DTN-based routing solutions and the characteristics of different networks or applications. Having known, a priori, a given set of application characteristics and requirements, we can choose or build a specific kind of routing solution. For example, where connectivity and mobility are low, but the nodes have enough resources in terms of energy, bandwidth, and buffering, and we need a reliable solution, the epidemic routing or any of its variant such as Spray and Wait [27] can be employed. On the other hand, if the connectivity is low in an environment where nodes are highly mobile and nodes' resources are restricted and expensive (in terms of energy, buffering or processing), message replication schemes are better candidates to be utilized. If reliability is needed by a routing solution, only epidemic routing or message coding can be employed.

Table 2 aims at summarizing the correspondence between network characteristics and DTN routing solutions. The rows in the table represents the properties of networks (or applications), whereas each column provides a different routing solution. If read line-by-line (horizontally), it states which routing modules may be *useful* or *necessary* to cope with the given characteristic (one per line). If read column-by-column (vertically), then it describes particular scenarios where the

given protocol (one per column) is a better choice. We do not intend that this table is all-inclusive or without exceptions. It is only rather an indication of which routing strategies might match better which DTN environments. It is also important to note that this table characterizes the suitability of a routing solution according to the set of network or application characteristics that we have presented in Section 4.

In the following, we take up a few exemplary networks, summarize their characteristics and describe what kind of routing protocol is suitable for each network.

1. A typical Vehicular Ad hoc Network (VANET), where vehicles exchange information when they come into contact of each other. In such a network, at some places the network may be very dense whereas at other places, it is sparse. The speed of nodes is generally high (from tens to hundreds km/h). Normally, resources are not scarce, especially in terms of power and memory. When choosing a suitable routing strategy in the light of what has been presented in the paper, one may opt for controlled replication as the routing algorithm because nodes have sufficient resources available and mobility is high.

2. Habitat monitoring such as ZebraNet [14], where animals are equipped with wireless sensors with little memory and limited battery lifetime, and we want to collect information about living conditions and environment. Resources are very precious in such a network, and speed is low (a few m/sec) with large pause times. Animals live most of the time in groups, and different groups occasionally encounter each other, and may exchange information. A coding scheme can be beneficial in such a scenario, as it works better with low resources, and because we can aggregate groups information together in order to save transmissions.

3. A social network in which people belonging to the same social community or interest form a network. People may also move in between different communities depending upon their changing interests, and due to variations in their daily life routines (e.g., workplace, home, market). Nodes in such a network can have diverse variations in terms of connectivity, mobility and resources, which makes this kind of network heterogeneous. In such a network, a hybrid approach of routing may be useful. For instance, controlled replication scheme such as Spray and Wait [27] can be used within a community, while some utility based smart replication scheme could be used for inter-community traffic.

Table 2 Routing Module Applicability

		Epidemic	Replicate	Smart replicate	Focus	Manet	Code
Connectivity	low	√	√	√	√		
	high				√	√	
Amount of Mobility	low	√			√	√	√
	high		√	√			
Structure of Mobility	homogeneous		√		√		
	heterogeneous			√	√		
	correlated			√	√		
Resources	low		√	√			√
	high	√			√	√	
Priority			√			√	
Reliability		√					√

6 Case Studies

In this section we will present simulation results to support and demonstrate our claims (design principles) from the previous section. Our goal here is not to provide extensive simulation results or argue for specific protocols, but rather to demonstrate the validity of our analysis of the routing solution space.

For the simulation results presented here, we assume that the time units used are the clock ticks of the discrete-time simulator. A packet transmission takes one time unit, so, in principle, one could translate this into seconds if needed by considering packet size and bit rate, and as the results presented here provide a comparative evaluation of delay, so the time unit does not make a difference.

6.1 Pocket Switched Networks

Pocket Switched Networks have been recently proposed [18] as a special type of DTN networks. The idea is to extend Internet connectivity beyond access points, by taking advantage of all possible means of "communication", including peer-to-peer links (e.g. Bluetooth), ephemeral access to a connected infrastructure (e.g. wireless Infostations [48], as well as physical mobility.

In this paradigm, nodes are assumed to be carried in the users' "pockets", during their daily life activities. This implies that patterns existing in the daily movement of different nodes (e.g. time of commuting to work and means of transportation used, time spent in the office or in other job locations, etc.) as well as interaction and social patterns between different users, are expected to affect considerably the transmission opportunities "seen" by the nodes.

There have been a lot of experimental studies recently trying to discover and quantify these mobility and "inter-meeting" patterns between users or nodes [50]. Some key findings include the following [49]: (i) nodes tend to show strong location or peer preference;

that is each node has a number of access points (peers) that it visits (sees) more often than others; (ii) nodes are rather heterogeneous in their mobility and interaction behavior; some nodes tend to see all other nodes often, while other only see a small set of peers throughout the measurement periods; (iii) inter-contact times between nodes have "heavy-tailed" behavior.

The above create a scenario where the respective transmissions opportunities have detailed structure. As we mentioned in Section 2 and Section 3, utility-based routing protocols that take into account, for example, the age of last encounter between nodes, are capable of discovering and taking advantage of such structure. In Fig. 2, we compare the performance of 3 protocols for the traces collected at the Infocom 2005 conference scenario [49]: (i) epidemic routing, (ii) a controlled replication protocol that blindly hands-over copies [27], and (iii) one that maintains last encounter information between nodes, and may forward a message copy further to another node with a more correlated mobility or encounter pattern with the destination.

As can be seen there, using controlled replication rather than epidemic routing can utilize the available bandwidth much better. Furthermore, using a utility function to discover better relays for a given destination can improve performance even more.[4]

6.2 Metropolitan Networks with Heterogeneous Nodes

Even though nodes in the previous scenario exhibit different social and movement behavior, they all still correspond to humans, and specifically pedestrians. However, there are situations where a larger variety of nodes may collaborate or coexist to enable intermittent connectivity in a larger (metropolitan) scale. Such a sce-

---

[4] Note that this is a small scenario with only 41 nodes, so performance improvement is more modest than in other scenarios; we expect that a similar mobility pattern in a larger setting - more nodes and more locations - would further demonstrate the importance of using the "right" routing modules.
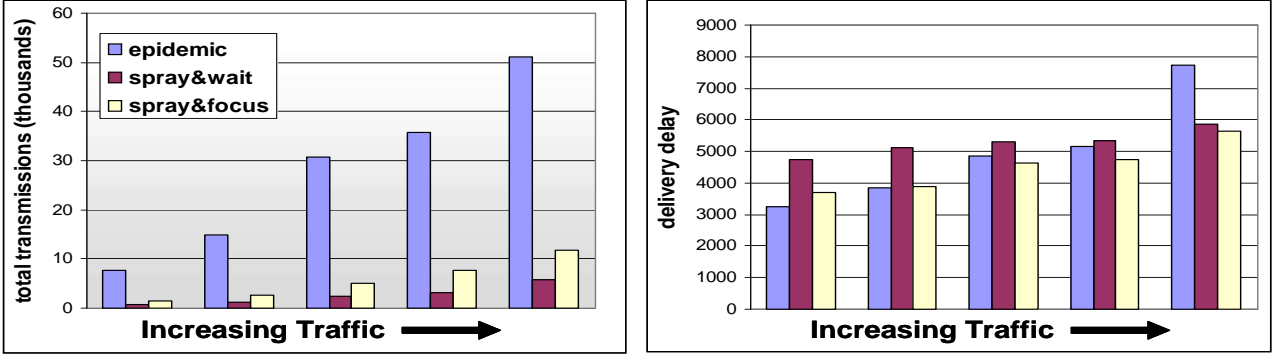
Fig. 2 Performance of different routing modules for trace-based mobility: Infocom 2005 traces collected by the HAGGLE project.
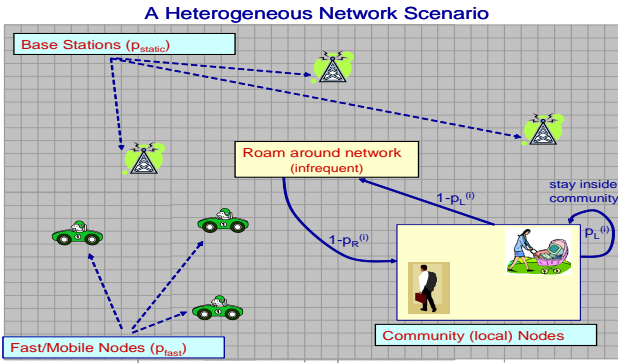

Fig. 3 Example scenario with heterogeneous wireless nodes

Fig. 4 Performance improvement of Smart Spraying over greedy Spray and Wait, as a function of the percentage p of "mobile" (useful) nodes; K is a node's transmission range.

nario might include, for example, nodes carried by pedestrians, other nodes mounted on vehicles, static nodes corresponding to base stations, sensors, or Throwboxes [17], etc., as shown in Fig. 3.

Scenarios like the one just described, involve a larger amount of heterogeneity. In addition to different social interactions, nodes in this case might also have largely varying amounts of resources as well as mobility ranges and speeds. For example, a node mounted on a car or a bus may cover a much larger network area than a node carried in the pocket of a pedestrian, and also may have no energy considerations. This implies the following: In the previous scenario some nodes may be better relays *for a specific destination* due, for example, to their social relation with the destination or their physical proximity to it; In this scenario some nodes may be better relays *for all destinations* due to some special capabilities of theirs like more resources or more peer encounters.

Imagine an example scenario where a percentage of nodes is mobile (e.g., cars, buses) and often performs long trips around the network, while the rest of the nodes move each inside its own local community only (e.g. campus, office building, etc.), which is much smaller than the total network area. In this network,
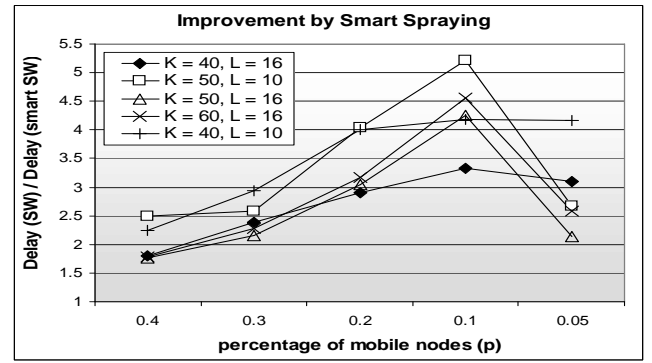
in order to route messages between nodes that lie in different communities, it is crucial to discover and take advantage of the few "mobile" nodes in the network. The rest of the nodes are useless for inter-community traffic.

In Fig. 4 we compare the delivery delay for two different routing strategies: (*Greedy Spraying*) in the first scheme, controlled replication is performed using a greedy distribution of the copies; all L copies of a message are handed over to the first L nodes encountered; (*Smart Spraying*) in the second scheme, we assume that each node carries a label that indicates what type of a node it is (e.g. "Vehicle", "Pedestrian", "Base Station", etc.)[5]. Copies of the messages are handed over only to nodes that carry a given label (e.g. "Vehicle"), that can travel outside the source's local community.

As can be seen in Fig. 4, blindly choosing relays could result in significant performance degradation in

---

[5] Although such labeling could be possible in some cases, we do not assume that a node must necessarily know its type; Instead, each node could maintain statistics of the "intensity of its mobility" by maintaining a running average of the number of different nodes it encounters within a given window of time; this could then be used as an estimate of how mobile a node is.

such a scenario. Although a few copies might happen to be handed over to "mobile" nodes that may eventually see a destination in a different community, most copies are wasted to nodes that rarely or never see the destination. On the other hand, a very simple optimization that tries to "read" a bit further into the structure of the surrounding network, could result in up to $5\times$ improvement. Specifically, the fewer the correct choices (i.e. the ratio of "good" over "bad" relays) the higher the potential improvement by trying to identify the good ones. Nevertheless, if the choices become too few, even "smart replication" is not powerful enough to discover the very few existing "paths-over-time", as evident in the plots as well. In that case, additional or different routing modules might be necessary to tackle the problem (e.g. flooding or utility-based forwarding).

6.3 Applications with Priorities

Despite the inherent delay-tolerance of the networks discussed, there can be situations where some messages may be more important than others. For example, in a VANET network it is reasonable to assume that an accident notification message will have higher priority than a chat message, or advertisements of nearby shops. Consequently, it would be useful to be able to treat priority messages preferentially, and ensure that *they get the best possible service, given the network limitations.* The questions is then, which routing strategy should be used for the priority messages and which for the non-priority ones, to satisfy the demands and semantics of both services?

Let us look at an example scenario where a $p\%$ of the messages have higher priority. In order to ensure the best possible service to these messages, we can use epidemic routing to route these messages only. Epidemic routing is guaranteed, under no buffer and bandwidth limitations to find the optimal paths in *any* scenario. Thus, it provides the *best effort, priority service* necessary in this context. The rest of the messages can be routed using a scheme like Spray and Wait. Spray and Wait: (i) generates very little traffic, which is important to not interfere significantly with the priority service; (ii) is robust enough to deliver good performance in a number of scenarios[6]. We have used simulations to answer the following two questions: what is the performance degradation to each service type, by the cross-traffic interference? how do these two services behave when the network becomes congested?

In Fig. 5 we assume there is 100 nodes that move according to the Random Waypoint mobility model in a 500×500 network. The velocity is considered as *1 grid unit per time unit*. The network area is measured in grid units, i.e. a network size of 500x500 is 500x500 grid units. It could be meters or whatever. So, each time unit, a node moves 1 grid unit. This is the simplest way to do the simulations, without fixing specific velocities, bit rates, etc. For Random Waypoint mobility model, the pause times are uniformly chosen in a [0,T], with T being relatively small. Quantitatively, the choice of value T does not make a real difference to the results obtained here.

We also assume that 10% of the messages (chosen randomly) have priority and routed using epidemic routing, and the rest of the messages are routed using Spray and Wait with $L = 16$ copies. We look first at how congestion affects the two traffic classes. As can be seen there, when traffic is not too high, both traffic classes coexist smoothly. Priority messages get the best possible services with at most $10 - 20\%$ degradation, while the delay of non-priority messages get increased a bit, and still remains competitive. On the other hand, if the network reaches congestion, it is important to note that it is the non-priority traffic class whose performance degrades the most and the fastest. This is very important, as it satisfies the semantics of a priority class, which is supposed to get the best service available.

Finally, in Fig. 6 we again depict the delivery delay and delivery ratio for the two traffic classes, as a function of the percentage of total messages that have high priority (we assume a fixed traffic load of 800 messages). We also include the delivery delay for the case where all messages are routed using epidemic routing, and for the case where all messages are routed using Spray and Wait. As is evident by these plots, using a different routing strategy for the two classes, achieves a much better trade-off than using the same routing protocol for all traffic, if the priority messages are only a fraction of the total messages (this is the *desirable* case in all priority services - see for example the air industry). Priority messages get better service than using Spraying for all traffic, which is the desired semantic. Furthermore, *both traffic classes get better service than if all messages were treated as priority!* Finally, if for some reason, the priority traffic increases (e.g. a major accident, natural disaster, etc.) it is the performance of the non-priority class that degrades first, with the priority traffic being again able to capture all available resources.

---

[6] We intend to look into using Spray and Focus rather than Spray and Wait for the non-priority service
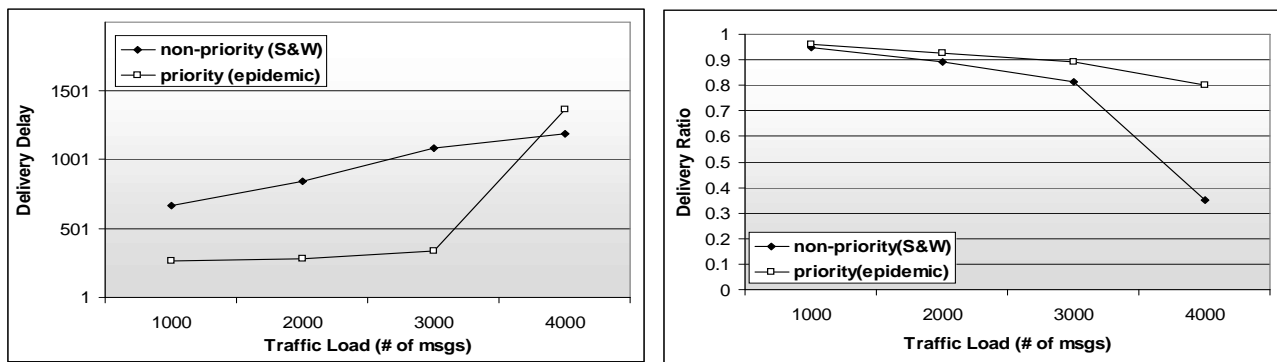
Fig. 5 Delivery Delay (left) and Delivery Ratio (right) for traffic classes with different priorities, as a function of total traffic.
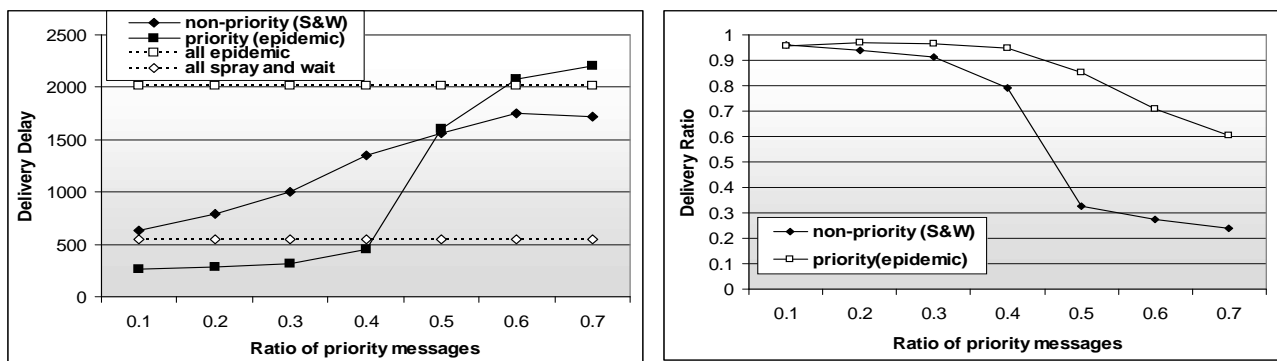

Fig. 6 Delivery Delay (left) and Delivery Ratio (right) for traffic classes with different priorities, as a function of the ratio of priority messages.

7 Conclusion

In this paper, we present a taxonomy of opportunistic routing protocols for DTNs. One of the main goals of our taxonomy is to have it serve as a set of guidelines for routing protocol designers and developers. The paper starts by defining basic building blocks used by existing DTN opportunistic routing schemes. We then create a taxonomy for intermittently connected networks based on network characteristics and application requirements. Finally, we present some case studies using a variety of existing DTN routing approaches to validate the proposed design principles and guidelines.

References

1. Thomas Clausen and Philippe Jacquet, "Optimized Link State Routing Protocol, draft-ietfmanet-olsr-11.txt", 2003.
2. Josh Broch, David A. Maltz, David B. Johnson, Yih-Chun Hu and Jorjeta Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols", In Proceedings of Mobile Computing and Networking, 1998.
3. Sushant Jain, Kevin Fall and Rabin Patra, "Routing in a Delay Tolerant Network", In Proceedings of ACM SIGCOMM, 2004.
4. W. Zhao, M. Ammar and E. Zegura, "A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Networks", In Proceedings of ACM/IEEE MOBIHOC, 2004.
5. R. Shah, S. Roy, S. Jain and W. Brunette, "Data MULEs: Modeling a Three-tier Architecture for Sparse Sensor Networks", IEEE SNPA Workshop, May 2003.
6. Amin Vahdat and David Becker, "Epidemic Routing for Partially Connected Ad Hoc Networks", Technical Report Number CS-200006, Duke University, 2000.
7. Dan Yu and Hui Li, "On the Definition of Ad Hoc Network Connectivity", In Proceedings of International Conference on Communications Technologies (ICCT), 2003, pages 990-994.
8. B. Krishnamachari, S.B. Wicker and R. Bejar, "Phase Transition Phenomena in Wireless Ad Hoc Networks", In Proceedings of IEEE Globecom, 2001, pages 2921-2925.
9. B. Krishnamachari, S.B. Wicker, R. Bejar and M. Pearlman, "Critical Density Threholds in Distributed Wireless Networks", In Proceedings of Communications, Information and Network Security, 2002, pages 1-15.
10. T. Spyropoulos, T. Turletti, and K. Obraczka, *Routing in Delay Tolerant Networks Comprising Heterogeneous Node Populations*, IEEE Transactions on Mobile Computing (TMC) Vol. 8, No. 8, pp. 1132-1147, August 2009.
11. Sachin Katti, Hariharan Rahul, Wenjun Hu, Dina Katabi and Muriel Medard, and Jon Crowcroft, "XORs In The Air: Practical Wireless Network Coding", In Proceedings of ACM SIGCOMM, 2006.
12. N. Sarafijanovic-Djukic, M. Piorkowski and M. Grossglauser, "Island Hopping: Efficient Mobility-Assisted Forwarding in Partitioned Networks", In Proceedings of IEEE SECON, September 2006.
13. Tristan Henderson, David Kotz and Ilya Abyzov, "The Changing Usage of a Mature Campus-wide Wireless Network", In Proceedings of the 10th annual international conference on Mobile computing and networking, 2004.

14. Philo Juang, Hidekazu Oki, Yong Wang, Margaret Martonosi, Li Shiuan Peh and Daniel Rubenstein, "Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet", In Proceedings of ACM ASPLOS, 2002.

15. J. Ghosh, S. J. Philip and C. Qiao, "Sociological Orbit Aware Location Approximation and Routing in MANET", In Proceedings of 2nd International Conference on Broadband Networks, 2005.

16. Mirco Musolesi and Cecilia Mascolo, "A Community Based Mobility Model for Ad Hoc Network Research", In Proceedings of ACM REALMAN, 2006.

17. Wenrui Zhao, Yang Chen, Mostafa Ammar, Mark Corner, Brian Levine and Ellen Zegura, "Capacity Enhancement using Throwboxes in DTNs", In Proceedings of IEEE MASS, 2006.

18. James Scott, Pan Hui, Jon Crowcroft and Christophe Diot, "Haggle: A Networking Architecture Designed Around Mobile Users", In Proceedings of IFIP Conference on Wireless On-demand Network Systems and Services (WONS), 2006.

19. Jeremie Leguay, Timur Friedman and Vania Conan, "DTN Routing in a Mobility Pattern Space", In Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005.

20. Anders Lindgren, Avri Doria and Olov Schelen,"Probabilistic routing in intermittently connected networks", SIGMOBILE Mobile Computing and Communication Review, Vol. 7 No. 3, 2003.

21. J. Boice, J.J. Garcia-Luna-Aceves and Katia Obraczka, "Disruption-Tolerant Routing with Scoped Propagation of Control Information", In Proceedings of International Conference on Communications (ICC), 2007, pages 3114-3121.

22. X. Zhang, G. Neglia, J. Kurose and D. Towsley, "Performance Modeling of Epidemic Routing", In Proceedings of IFIP Networking, 2006.

23. Tara Small and Zygmunt Haas, "Resource and Performance Tradeoffs in Delay-Tolerant Wireless Networks", In Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005.

24. P. Ramanathan and A. Singh, "Delay Differentiated Gossiping in Delay Tolerant Networks", In Proceedings of International Conference on Communications (ICC), 2008, pages 3291-3295.

25. J. Burgess, B. Gallagher, D. Jensen and B. N. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks", In Proceedings of IEEE INFOCOM, April 2006.

26. M. Grossglauser and D. Tse, "Mobility Increases the Capacity of Ad Hoc Wireless Networks", IEEE/ACM Transactions on Networking, vol 10, no 4, August 2002.

27. Thrasyvoulos Spyropoulos, Konstantinos Psounis and Cauligi S. Raghavendra, "Spray and Wait: Efficient Routing in Intermittently Connected Mobile Networks", In Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005.

28. Giovanni Neglia and Xiaolan Zhang, "Optimal Delay-Power Tradeoff in Sparse Delay Tolerant Networks: A Preliminary Study", In Proceedings of ACM SIGCOMM workshop on Challenged Networks (CHANTS'06), 2006.

29. Y. Wang, Sushant Jain, Margaret Martonosi and Kevin Fall, "Erasure Coding Based Routing for Opportunistic Networks", In Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005.

30. X. Zixiang, A.D. Liveris and S. Cheng, "Distributed Source Coding for Sensor Networks", IEEE Signal Processing Magazine, Vol. 21, Issue 5, September 2004.

31. Ignacio Solis and Katia Obraczka, "Efficient Continuous Mapping in Sensor Networks Using Isolines", In Proceedings of MobiQuitous, 2005, pages 325-332.

32. Abhinav Kamra, Vishal Misra, Jon Feldman and Dan Rubenstein, "Growth Codes: Maximizing Sensor Network Data Persistence", In Proceedings of ACM SIGCOMM, 2006.

33. Joerg Widmer and Jean-Yves Le Boudec, "Network Coding for Efficient Communication in Extreme Networks", In Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005.

34. S.-Y. R. Li, R. W. Yeung and N. Cai, "Linear Network Coding", IEEE Transactions on Information Theory, February 2003.

35. S. Deb, C. Choute, M. Medard and R. Koetter, "Data Harvesting: A Random Coding Approach to Rapid Dissemination and Efficient Storage of Data", In Proceedings of the IEEE INFOCOM, March 2005.

36. Rao Naveed Bin Rais, Thierry Turletti, and Katia Obraczka, *Coping with Episodic Connectivity in Heterogeneous Networks*, In Proceedings of the 11th International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), pp. 211-219, Canada, 2008.

37. Rao Naveed Bin Rais, Thierry Turletti, and Katia Obraczka, *MeDeHa - Efficient Message Delivery in Heterogeneous Networks with Intermittent Connectivity*, INRIA Research Report No. 7227, inria-00464085, 2010.

38. Henri Dubois-Ferriere, Matthias Grossglauser and Martin Vetterli, "Age Matters: Efficient Route Discovery in Mobile Ad Hoc Networks using Encounter Ages", In Proceedings of ACM MobiHoc, 2003.

39. M. Grossglauser and M. Vetterli, "Locating Mobile Nodes with EASE: Learning Efficient Routes from Encounter Histories Alone", IEEE/ACM Transactions on Networking, vol 14, no 3, June 2006.

40. First Mile Solutions, "http://www.firstmilesolutions.com".

41. KioskNet (VLINK), University of Waterloo, Canada, "http://blizzard.cs.uwaterloo.ca/tetherless/index.php".

42. J. Leguay, V. Conan and T. Friedman, "Evaluating MobySpace-based Routing Strategies in Delay-Tolerant Networks", Wireless Communications and Mobile Computing, May 2007.

43. S. Capkun, L. Buttyan and J. Hubaux, "Self-Organized Public Key Management for Mobile Ad Hoc Networks", IEEE Transactions on Mobile Computing, VOL.1, NO.1, 2002.

44. Paul Resnick, Richard Zeckhauser, Eric Friedman and Ko Kuwabara, "Reputation Systems, Facilitating Trust in Internet Interactions", In Proceedings of Communications of the ACM, December 2000, pages 45-48.

45. Thrasyvoulos Spyropoulos, Konstantinos Psounis and Cauligi Raghavendra, "Spray and Focus: Efficient Mobility-Assisted Routing for Heterogeneous and Correlated Mobility", In Proceedings of IEEE PERCOM, on the International Workshop on Intermittently Connected Mobile Ad hoc Networks (ICMAN), March 2007.

46. Aruna Balasubramanian, Biran Neil Levine and Arun Venkataramani, "DTN Routing as a Resource Allocation Problem", In Proceedings of the ACM SIGCOMM, August 2007.

47. Aruna Balasubramanian, Brian Neil Levine, and Arun Venkataramani, "Replication Routing in DTNs: A Resource Allocation Approach", To appear in IEEE/ACM Transactions on Networking, 2010.

48. Tara Small and Zygmunt J. Haas, "The shared wireless infostation model: a new ad hoc networking paradigm (or where there is a whale, there is a way)", In Proceedings of the 4th ACM international symposium on Mobile ad hoc networking and computing, June 2003.

49. Pan Hui, Augustin Chaintreau, James Scott, Richard Gass, Jon Crowcroft and Christophe Diot, "Pocket Switched Networks and Human Mobility in Conference Environments", In

Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005.

50. Marvin McNett and Geoffrey M. Voelker, "Access and Mobility of Wireless PDA Users", ACM Mobile Computing and Communication Review, 2003.

51. Brendan Burns, Oliver Brock and Brian Neil Levine, "MV Routing and Capacity Building in Disruption Tolerant Networks", In Proceedings of the IEEE INFOCOM, March 2005.

52. Minsu Shiny, Seongik Hongyy and Injong Rhee, "DTN Routing Strategies using Optimal Search Patterns", In Proceedings of the third ACM Workshop on Challenged Networks (CHANTS), September 15, 2008, San Francisco, California, USA.

53. Samuel C. Nelson, Mehedi Bakht and Robin Kravets, "Encounter-Based Routing in DTNs", In Proceedings of the IEEE INFOCOM, Rio de Janerio, Brazil, April 2009.

54. Ram Ramanathan, Richard Hansen, Prithwish Basu, Regina Rosales-Hain and Rajesh Krishnan, "Prioritized epidemic routing for opportunistic networks", In Proceedings of the 1st International MobiSys Workshop on Mobile Opportunistic Networking, Puerto Rico, 2007.

55. Vijay Erramilli and Mark Crovella, "Forwarding in Opportunistic Networks with Resource Constraints", In Proceedings of third ACM Workshop on Challenged Networks (CHANTS), September 15, 2008, San Francisco, CA, USA.

56. Gabriel Sandulescu and Simin Nadjm-Tehrani, "Opportunistic DTN routing with Window-Aware Adaptive Replication", In Proceedings of the ACM 4th Asian Conference on Internet Engineering (AINTEC), Bangkok, Thailand, November 2008.

57. Pan Hui, Jon Crowcroft and Eiko Yoneki, "BUBBLE Rap: Social-based Forwarding in Delay Tolerant Networks", In Proceedings of ACM MobiHoc'08, Hong Kong, May 2008.

58. Vijay Erramilli, Mark Crovella, Augustin Chaintreau and Christophe Diot, "Delegation Forwarding", In Proceedings of ACM MobiHoc'08, Hong Kong, May 2008.

59. H. Jun, M. H. Ammar, and E. W. Zegura, "Power Management in Delay Tolerant Networks:A Framework and Knowledge-Based Mechanisms", In Proceedings of IEEE SECON, 2005.

60. W. Wang, V. Srinivasan, and M. Motani, "Adaptive Contact Probing Mechanisms for Delay Tolerant Applications", In Proceedings of ACM Mobicom, 2007.

61. Eitan Altman, Amar Prakash Azad, Tamer Basar, and Francesco De Pellegrini, "Optimal Activation and Transmission Control in Delay Tolerant Networks", In Proceedings of IEEE Infocom, 2010.

62. Y. Xi, M. Chuah, and K. Chang, "Performance Evaluation of a Power Management Scheme for Disruption Tolerant Network", Lecture Notes in Computer Science, vol. 12, no. 5.6, pp. 370-380, December 2007.

63. Bong Jun Choi and Xuemen Shen, "Adaptive Asynchronous Clock based Power Saving Protocols for Delay Tolerant Networks", In Proceedings of IEEE Globecom'09, Honolulu, Hawaii, USA, 2009.

64. Nilanjan Banerjee, Mark D. Corner, and Brian Neil Levine, "Design and Field Experimentation of an Energy-Efficient Architecture for DTN Throwboxes", To appear in IEEE/ACM Transactions on Networking, 2010.