

## RESEARCH ARTICLE

# Exploiting Resource Heterogeneity in Delay-tolerant Networks<sup>†</sup>

Gabriel Sandulescu<sup>1</sup>, Peter Schaffer<sup>2</sup> and Simin Nadjm-Tehrani<sup>3</sup><sup>1</sup> Computer Science and Communications Research Unit (CSC), University of Luxembourg, Luxembourg<sup>2</sup> Interdisciplinary Centre for Security Reliability and Trust (SNT), University of Luxembourg, Luxembourg<sup>3</sup> Real-Time Systems Laboratory (RTSLAB), Linköping University, Sweden

## ABSTRACT

Routing in delay and disruption-tolerant networks (DTNs) relies on intermediary nodes, called custodians, to deliver messages to destination. However, nodes usually differ significantly in terms of available resources: energy, buffer space, and bandwidth. Routing algorithms need to make the most efficient use of custodian resources while also making sure those in limited supply are not exhausted. This paper proposes a distributed scheme for calculating resources available in node vicinity, as a tool to support meaningful routing decisions. A generic model is developed first, and is then applied to individual network assets. The model is based on a sparse network, where resources are potentially not uniformly distributed. It uses recent encounters to estimate resource availability in node vicinity. It is shown that a store-carry-forward scheme may benefit from accessing vicinity resource estimates. This knowledge allows nodes to implement meaningful custodian election and queue management strategies, approached here from a holistic perspective. It is demonstrated that routing protocols not only use up fewer resources overall, but also consume resources preferentially from nodes with higher resource levels, sparing nodes with limited supplies. As a result, disparities in available resources across the node population are significantly reduced, and nodes are less likely to leave the network as a consequence of resource depletion. Copyright © 2011 John Wiley & Sons, Ltd.

## KEYWORDS

routing; resource management; delay-tolerant networks

## 1. INTRODUCTION

Opportunistic routing schemes are designed to deliver messages in the absence of any scheduling information in partitioned networks. The latter have often been referred to in the literature as delay- and disruption-tolerant networks (DTN) [1]. Since encounter patterns and schedules are not known, the key idea is to leverage some network nodes for forwarding messages towards destination using the store-carry-forward paradigm. Many opportunistic schemes assume that nodes possess an average level of network resources that are homogeneously distributed over the network [2, 3].

However, resource distribution is not homogeneous in real networks, so the average resource assumption usually does not hold. Real networks may be composed of smartphones, car embedded computers, laptops, fixed throwboxes and sensors, which obviously have different energy requirements and different buffer space allowances. Moreover, neither message source nodes, nor message destination nodes need to be uniformly distributed across the node population. Therefore, network resource availability at nodes is uneven. Given this resource heterogeneity, having a local estimate for the resources available in the vicinity of a node would be highly beneficial for routing decisions. For example, based on such estimates, a node may decide to whom to forward its messages (custodian election) and which messages to transfer first (queue management).

In an attempt to enhance delivery performance and to minimise end-to-end delay in disconnected networks, recent studies have looked at node behaviour from several new perspectives. Some put forward the idea of social correlations between nodes, which may influence node

\* Correspondence to: Simin Nadjm-Tehrani, Real-Time Systems Laboratory, Linköping University, S-58183 Linköping, Sweden

<sup>†</sup> This paper is based on 'Vicinity Resource Cartography for Delay-Tolerant Networks: a Holistic Perspective' by G. Sandulescu, P. Schaffer and S. Nadjm-Tehrani which appeared in the Proceedings of the 3<sup>rd</sup> IFIP Wireless Days Conference (WD'10), Venice, Italy, October 2010.

contact distribution [4], while others propose techniques based on a combination of social context and node topology information [5, 6]. These techniques select some key nodes, among those that are located more centrally in the network or are more active socially, considering them to be better suited for forwarding incoming messages. In so doing, however, they place additional strain on the resources available to these key nodes, and they will be the first to fail as a consequence of network congestion, buffer exhaustion, or battery depletion. Thus, in the absence of any metrics for the resources available to nodes, these strategies may actually jeopardise initial performance gains by losing the most important nodes through overuse.

Other studies focus on the particular movements building up higher densities around points of interest [7]. However, in partitioned ad-hoc networks the localisation of such points of interest, acting as resource concentrators, is difficult to achieve. Moreover, resource concentrators may move in space and vary in time. For example, the distribution of resources in a disaster area cannot be planned in advance. Somewhat similarly, the configuration of points of interest in case of a traffic jam triggered by an accident varies in space and time as an effect of the accordion phenomenon [8]. Finally, even when the geographical distribution of resources is known in advance and is constant in time, being able to detect them autonomously, that is without central knowledge, makes a big difference in terms of protocol robustness.

Knowing the time-varying and space-varying resource distribution in a network can have a huge impact on the choice of routing strategies, even when only *approximate* knowledge is available. An important problem then is how to select custodians efficiently, i.e. depending on own resources and on resources available in the neighbourhood. For example, a node may choose not to forward a message at a particular encounter knowing that better opportunities than the current one may arise in the near future. Or alternatively, a node may choose to forward a message to a given fraction of nodes that it has selected from a list of nodes with the highest levels of available resources. Another problem to consider is how to prioritise messages in a message queue so as to give them a good chance to be transferred within the limited contact window of an encounter.

The scenario considered in this paper is the following: in a given perimeter (henceforth called playground) mobile nodes with heterogeneous available resources and various mobilities (pedestrians, cars, bicycles) use their mobility with a view to increasing connectivity. This can be imagined either (1) in a city context where nodes may form trusted cliques to avoid oversubscribed infrastructures or (2) in a disaster management context where infrastructure has been destroyed and nodes need to deliver messages relying on store-carry-forward mechanisms. In both contexts we assume that we are dealing with low node

density and high node mobility, as well as with the need to send relatively large messages through unicast.

In this paper, we propose a distributed scheme for estimating the resources available in the proximity of a node, with no a priori knowledge. We study the accuracy of this scheme and validate it in three different simulation settings showing that its inaccuracy is below 10% in the scenarios considered. For this study we consider separately space-varying and time-varying resource maps, and perform validation in both a random waypoint scenario and a disaster area scenario (using Bonn motion traces [9]). We then go on to propose a set of policies where those estimates are used for selecting a suitable custodian (custodian election) and for choosing the right message to transfer (queue management).

This paper is an extension of preliminary work [10] presented earlier. The work has been extended along two lines: (1) proposing a new custodian election policy allowing a node to select only a given fraction of nodes as custodians out of the total number of encounters of that particular node. This fraction is set as a protocol parameter. Hence, this proposal can be seen as a version of controlled replication, and as an effective method to limit resource consumption. Moreover, this replication is resource-aware, as it consists in selecting only the best-fitted custodians in the vicinity of a node, i.e. those that have the highest level of resources available; (2) showing, by means of extended simulations, that the proposed custodian election policies minimise the number of exhausted nodes by evening out resources for all the nodes in the network. What actually happens is that network resources are consumed preferentially from nodes with above average resources, while nodes with below average assets are spared as much as possible, which helps keep the number of completely exhausted nodes at a minimum. Our findings demonstrate a substantial performance enhancement, particularly in networks with heterogeneously distributed resources. It also reveals that a node choosing about 10-20% of all the nodes encountered as custodians is a good rule of thumb for achieving best results.

## 2. RELATED WORK

Under some simplifying assumptions, such as uniform mobility patterns, a large number of nodes and small message sizes, simple protocols such as Epidemic [11], or  $k$ -hop forwarding [12] can be studied analytically. Zhang et al. [13] have obtained a rich set of closed form formulas for average delivery delay and for number of copies sent under the following extended epidemic schemes:  $k$ -hop forwarding, probabilistic forwarding, and limited-time forwarding. Similarly, Jacquet et al. [14] have studied the theoretical upper bound of propagation delay in disconnected networks that can be achieved using any routing algorithm. However, in the case of more complex protocols, analyses are usually performed by comparative

simulations using specific mobility models or concrete traces.

An instance of such a complex protocol is RAPID [15], which introduced the idea of routing as a resource allocation problem. In other words, at every transfer opportunity, the marginal utility of replication should justify the resources used. The objective of the protocol is to minimise the total volume of resources used for achieving the required metrics, such as maximum or average delivery delay. The protocol takes into account resources such as bandwidth and buffer space, but does not consider energy. RAPID focuses on resources used overall, ignoring the actual nodes that make use of those resources. As different from the above, in this paper available resources are to be allocated from the node's perspective and the minimisation of available resources is done per node. While RAPID aims at minimising the total amount of resources consumed, this paper proposes mechanisms that avoid one node getting exhausted while the nodes close by still have plenty of resources. Another difference between the two protocols is that RAPID does not exploit the size of the actual messages transmitted, but only average message sizes. This means that there is no adaptation of the messages sent to the actual contact windows. This paper, on the other hand, considers message size as a significant factor and proposes an optimised mechanism for message forwarding depending on contact window size.

Another work which deals with the resource allocation problem proposes GBS (Global knowledge Based Scheduling and Drop) [16] which includes mechanisms for message scheduling and message deletion. While selecting the most appropriate message to send is also one of the objectives pursued in our work, we have not investigated message deletion policies. In addition to the objective presented in [16], however, we have investigated custodian election policies - i.e. we have analysed to which nodes it would be most appropriate to convey messages, based on their resource assessment.

Some other works start out from the idea that the Epidemic protocol [11] is pretty good at achieving high delivery rates but fails specifically when it starts dropping packets due to storage considerations and bandwidth overhead. Ramanathan et al. propose a protocol called PREP (PRioritised EPidemic) [17] where dissemination of messages is done in an epidemic manner but the protocol also includes smarter policies dealing with priority-based bundle transmission and bundle deletion. Plugging smarter policies into some protocol (Epidemic or other) is also an idea used in our work. PREP only analyses bandwidth and buffer space. In addition to these two resources our work also takes into account energy. Furthermore, PREP optimises routing only from a message perspective, similarly to what we call queue management policies in our paper. In addition to that, our work also analyses policies dealing with custodian election. We show that implementing custodian election

policies leads to more substantial improvements in delivery rate as compared with improvements from implementing queue management policies, especially when nodes are considered to have heterogeneous resources. However, our hypothesis differs from PREP, we consider resources heterogeneously distributed in the node population while in PREP these are considered homogeneously distributed.

To the best of our knowledge, there is no unified proposal in the literature linking together the most relevant resources in DTN: energy, buffer space, and bandwidth, into one single abstraction. In our case, past encounters are used to predict the evolution of available resources in the near-term. Prediction-based schemes have already been proposed in earlier works [3, 18], but they mostly deal with contact probability, thus ignoring the amount of resources that nodes are contributing to the network. On the other hand, in a heterogeneous environment, where resources range over a wide spectrum of types and levels, estimating resource availability remains an open question.

Another prediction-based protocol, MobySpace [19], proposes to construct a high-dimensional Euclidean space from node mobility patterns, while also assuming prior knowledge (or learning) of those patterns. We have also used Euclidean space in our approach, but we have focused on the resources available in node vicinity. Moreover, the equations we have proposed are not tightly constrained by one mobility model or another, as demonstrated by the use of two alternative scenarios.

Heterogeneity has been studied by Spyropoulos et al. [20], who propose a utility function for the selection of appropriate custodians. However, the work mentioned above does not consider accounting for resources in general, or bandwidth in particular, which leads to the questionable assumption that a message may be conveyed over a meeting regardless of its size and appropriateness of the contact window.

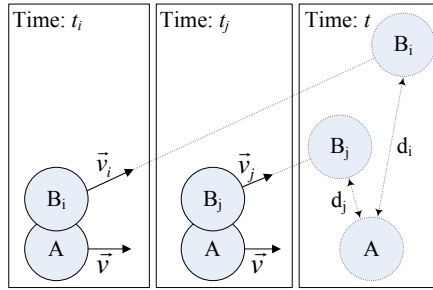
### 3. RESOURCE AVAILABILITY IN PARTITIONED NETWORKS

In this section we propose a distributed scheme for calculating the level of resources available in the vicinity of a node. We start by developing a generic model, which we then apply to individual network assets, such as buffer space, energy, and bandwidth.

We consider a sparse network where resources are potentially not uniformly distributed. As a consequence, there may be pockets of resources in the network, in the form of energy or buffer space, which may vary both in time and space. Incidentally, the way in which resources are distributed in a network may also be the consequence of mobility, but we are not assuming any particular mobility pattern for our model. As an effect of sparsity, we will assume that most meetings happen between two nodes.

### 3.1. Resource variation in time and space. The generic model

We consider a generic resource  $R$  for a given node. Obviously, every node knows its own resource level. Let us denote with  $R_A^o$  the own resource level available at node  $A$  and with  $R_A^v$  the estimated resource available in the vicinity of node  $A$ . Applying the scheme below, every node will be able to evaluate the expected virtual value of resource  $R$  available in its vicinity.



**Figure 1.** Node  $A$  meetings history within observation time span

Let us consider a node  $A$  moving with velocity  $\vec{v}$ , and a set of nodes  $B_i \dots B_j$  moving with velocities  $\vec{v}_i \dots \vec{v}_j$ , respectively. Every node knows its own velocity and resource level, and communicates them at each meeting. For instance, when a meeting happens between node  $A$  and node  $B_i$ , node  $A$  receives vector  $I_i$  from node  $B_i$ :

$$I_i = (\vec{v}_i, R_i^o)$$

Consider a sequence of encounters between node  $A$  and nodes  $B_i, \dots, B_j$  at times  $t_i, \dots, t_j$  at which  $A$  collects resources  $R_i^o, \dots, R_j^o$  from those nodes. We assume that  $A$  maintains a log of maximum  $j - i + 1$  encounters during a sliding time window of size  $\tau$ . After meeting the last node ( $B_j$ ), node  $A$  will have the following information available:

$$M_A = \begin{pmatrix} t_i & t_{i+1} & \dots & t_j \\ \vec{v}_i & \vec{v}_{i+1} & \dots & \vec{v}_j \\ R_i^o & R_{i+1}^o & \dots & R_j^o \end{pmatrix}$$

$M_A$  is determined by adding local time information  $t_k$  to the vector  $I_k$  received at each meeting with node  $B_k$ . We note that the information the node needs to store is limited for two reasons: first, because the maximum number of columns in the matrix ( $j - i + 1$ ) is limited, and second, because all information older than  $\tau$  is discarded (such as  $t_j - t_i < \tau$  at any point in time).

Considering that velocities  $\vec{v}$  and  $\vec{v}_i, \dots, \vec{v}_j$  remain constant for the short observation time span  $\tau$ , we can calculate the Euclidean distance between node  $A$  and nodes  $B_i, \dots, B_j$  at a later point in time than the encounter time, as shown in Figure 1. We can also calculate the contribution of node  $B_k$  to the estimated virtual resources of  $A$  as being inversely proportional to the distance from

$B_k$  to  $A$ , denoted by  $d_k$ . We denote the resource footprint in the vicinity of  $A$  as  $R_A^v$ , and compute it as a function of resources met by node  $A$  during the time span  $\tau$ . Finally,  $R_A^v$  is proportional to the number of meetings  $A$  has over the observation time  $\tau$ . As we do not want to limit our findings to one particular mobility model, we do not use meeting probabilities. Consequently, we need to rely on some derived measures. Putting all the information together, we can apply the formula below to calculate the estimated available resource in the vicinity of node  $A$ :

$$R_A^v = \underbrace{\frac{n_\tau}{\omega \times \tau}}_{c_A} \times \frac{\sum_{k=i}^j \frac{R_k^o}{d_k}}{\underbrace{\sum_{k=i}^j \frac{1}{d_k}}_{\bar{R}_A}} \quad (1)$$

where:  $\tau$  = observation time span  
 $\omega$  = node's average meeting frequency  
 $n_\tau$  =  $j - i + 1$ , number of nodes  $A$  actually met during observation time  $\tau$

In other words, as meetings occur and resource information is exchanged, each node builds up its own map of virtual resources, assigning greater weight to those at a shorter distance. Equation (1) can be decomposed into:

- an element  $c_A$  reflecting the density of meetings in the given region. This acts as a generic factor irrespective of resource type  $R$ .
- an element  $\bar{R}_A$  representing the average availability of resource  $R$  weighted by the inverse of the distance between  $A$  and the nodes met.

The validation of this formula is done in extensive scenarios in Sections 4 and 5 but here we consider the following special cases for discussion:

- if node  $A$  had no meetings over the  $\tau$  time span:  $\lim_{n_\tau \rightarrow 0} R_A^v = 0$  because  $n_\tau$  respectively  $c_A$  is 0
- if the node actually had an average number of meetings over time span  $\tau$ :  $\lim_{c_A \rightarrow 1} R_A^v = \bar{R}_A$

### 3.2. Modelling individual network assets

After having developed a generic model for calculating resources available in node vicinity, let us now move on to the second step in our modelling exercise, and refine this equation for individual network assets: buffer space, energy, and bandwidth. While buffer space and energy are properties related to one node, bandwidth is a property linking together two or more nodes. We treat the three categories of resources in an increasing order of complexity.

#### 3.2.1. Buffer space

The buffer space case is straightforward. Equation (1) can be used directly for calculating buffer space by simply

replacing the generic resource with buffer space in the formula. This is possible because buffer space remains constant as long as no messages are exchanged between nodes. That is, for short observation times  $\tau$  and low network load, we can approximate a node's buffer space at time  $t$  with the buffer space we have observed at time  $t_i > t - \tau$ . Denoting the available buffer size with  $S$ , we can directly replace  $R$  by  $S$  in Equation (1).

### 3.2.2. Energy

The energy model is more complex, because energy levels do not remain constant, even in the absence of message exchange. In case there is traffic, energy is depleted by the sending and receiving of messages at a rate approximately proportional to the size of messages exchanged. In case there is no traffic, node energy decreases simply due to network sensing. We can approximate the energy level at a node, at one particular timepoint  $t$ , by relating it to the relevant factors, as follows:

$$E^o(t) = E_{in} - \underbrace{e_s \times t}_{\text{Energy for sensing}} - \underbrace{e_m \times m}_{\text{Energy for message exchange}} \quad (2)$$

where:  $E^o(t)$  = node's own energy at time  $t$   
 $E_{in}$  = initial (maximum) energy available for this type of node  
 $e_s$  = energy factor for sensing  
 $e_m$  = energy factor used for message exchange  
 $m$  = total size of exchanged messages

Factors  $e_s$  and  $e_m$  can both be measured for different types of nodes in a laboratory setup [21]. In a simpler setup, every node can measure energy depletion as a function of time just by retrieving battery levels at 2 different times. Denoting this attenuation rate by  $e$ , the above equation is simplified as follows:

$$E^o(t) = E_{in} - \underbrace{e \times t}_{\text{attenuation with time}}$$

Thus, time variable  $E^o(t)$  replaces the constant  $R_k^o$  in Equation (1) as a node's estimate for own energy.

### 3.2.3. Bandwidth

For the purposes of this model, we define bandwidth as the maximum volume of data  $D_N$  that a node  $N$  can exchange at one meeting. Our model is meant to achieve a twofold objective: first, to provide an estimate for a node's capacity to send and receive messages (at one meeting or over a given time span); and second, to help determine which message to send, depending on message size and the estimated probability of success.

The theorem we propose below will allow us to estimate the volume of data exchanged between two nodes, as well

as the probability of a message to pass, taking contact window estimation as a basic factor. A contact window denotes the time during which two nodes are in radio range of each other, and represents a critical factor for realistically evaluating bandwidth in mobile networks. As mentioned earlier, we consider that most encounters will happen between only two nodes, as a result of network sparsity.

#### Theorem 1

For a meeting between two nodes (disk radio range with radius  $r$ ), characterised by a relative velocity of  $\vec{v}_{rel}$  and communicating over a protocol with nominal bandwidth  $b_n$  we can calculate:

- I. the maximum volume of data exchanged during the meeting as:

$$D_{max} = \frac{2r}{|\vec{v}_{rel}|} \times b_n$$

- II. the expected volume of data exchanged during the meeting as:

$$D_{exp} = \frac{\pi r}{2|\vec{v}_{rel}|} \times b_n$$

- III. the probability of an exchange exceeding a given size:

$$Pr\{D \geq pD_{max}\} = \sqrt{1 - p^2} \text{ where } p \in [0, 1]$$

#### Proof

As shown in Figure 2, a node crosses the radio range of another on a trajectory  $PQ$ . This trajectory is covered at a velocity  $\vec{v}_{rel}$  and  $P$  is the incidence point between the nodes' radio ranges. Using geometry we can calculate the trajectory between  $P$  and  $Q$  as a function of  $x$ :

$$f(x) = \overline{PQ} = 2\sqrt{x(2r - x)}$$

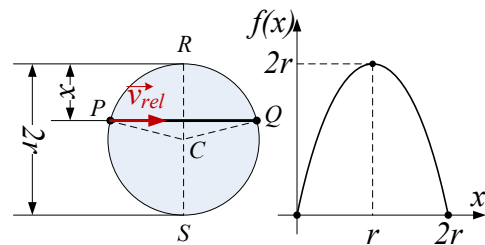


Figure 2. Contact window as a function of incidence point

If we start out from the assumption that nodes meet,  $x \in [0, 2r]$  and contact point  $P$  may be anywhere on the circle arc  $\widehat{RPS}$ . Moreover, in the generic case,  $x$  is a random variable uniformly distributed over the interval  $[0, 2r]$ . Thus,  $D_{max}$  can be calculated as:

$$D_{max} = \frac{\overline{PQ}_{max}}{|\vec{v}_{rel}|} \times b_n = \frac{2r}{|\vec{v}_{rel}|} \times b_n \quad (I)$$

$$\begin{aligned}
D_{exp} &= \frac{\overline{PQ}_{exp}}{|\vec{v}_{rel}|} \times b_n = \frac{\int_0^{2r} f(x) dx}{|\vec{v}_{rel}|} \times b_n = \\
&= \frac{2r^2 \arctan\left(\frac{\sqrt{x}}{\sqrt{2r-x}}\right) + \sqrt{x(2r-x)}(x-r)\Big|_0^{2r}}{2r|\vec{v}_{rel}|} \times b_n = \\
&= \frac{\pi r}{2|\vec{v}_{rel}|} \times b_n \quad (II)
\end{aligned}$$

Now, we calculate the probability of having an exchange exceeding a given fraction  $p \in [0, 1]$  of the maximum  $D_{max}$  as:

$$\begin{aligned}
Pr\{D \geq pD_{max}\} &= Pr\{f(x) \geq p2r\} = \\
Pr\{2\sqrt{x(2r-x)} \geq 2pr\} &= \frac{|x_1 - x_2|}{2r} = \sqrt{1 - p^2} \quad (III)
\end{aligned}$$

where  $x_1$  and  $x_2$  are the solutions of the quadratic equation  $2\sqrt{x(2r-x)} = 2pr$   $\square$

Theorem 1 allows us to estimate one particular network asset – bandwidth – characterised at one node, namely the expected data volume that this node can exchange at one particular meeting. We can now easily integrate bandwidth into our model for generic resources in order to estimate the expected bandwidth value in the vicinity of a node, as follows. At each meeting time  $t_i, \dots, t_j$  a node knows its own velocity as well as that of its recently encountered neighbours, so it can calculate the relative velocities  $v_{rel,i}, \dots, v_{rel,j}$ . In a given scenario,  $r$  and  $b_n$  are approximated as known constants. Therefore, by using Theorem 1.II, each node can calculate the array of expected data to be exchanged ( $D_{exp,i}, \dots, D_{exp,j}$ ) and, by using Equation (I), it can estimate the expected amount of data  $D_A^v$  that can be exchanged by node  $A$  over the observation time  $\tau$ .

One additional application of Theorem 1 is that, if node  $A$  actively manages its message queue, it can also choose which message to send by evaluating the probability of successful message transmission given by Theorem 1.III.

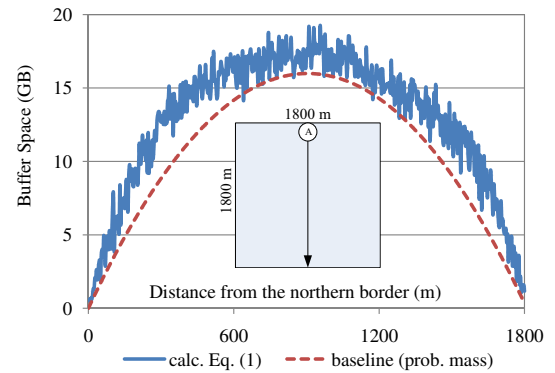
## 4. VALIDATION OF THE MODEL

In this section we demonstrate how the virtual resources in the neighbourhood of a node can be estimated by applying Equation (I) to exchanges between nodes, as proposed in Section 3. The goal of validation is to show that these estimates are indeed close enough to the real resource levels in the network. We have organised our validation exercise in two sections: in this section we validate our model applied to buffer space in three different scenarios,

while in Section 5 we validate our holistic approach considering all three network resources.

### 4.1. Validation in a space-varying environment

As shown by Hyytiä et al. [22], when nodes move according to random waypoint mobility in a square, node density is maximum in the middle of the area and decreases to 0 towards the borders. This observation allows us to validate Equation (I) in a simple, yet revealing experiment where we can isolate the time-varying element from the space-varying element. Assuming constant buffer space per node (i.e. nodes do not exchange large messages, but only small amounts of information, as required for Equation (I)), virtual buffer distribution is given predominantly by node densities and is constant in time.



**Figure 3.** Available buffer space in the vicinity of a node, space-varying scenario

We used the ONE simulation environment [23] where we considered a set of 100 nodes moving at 20 m/s according to random waypoint mobility within a space defined by a square with an edge length of 1800 m, as well as a "spy" node  $A$  moving very slowly at 0.04 m/s on a rectilinear path starting from the middle of the northern edge and ending in the middle of the southern edge. Initially, all nodes have a fixed amount of buffer space (500 MB) and, since no message is delivered between nodes, the resource map is defined exclusively by node density. We modelled the random waypoint movement by creating 100 different movements and computing the average value of  $R_A^v$  over time, according to Equation (I). We considered an observation period of  $\tau = 5$  minutes over a 12-hour scenario, while limiting the  $M_A$  columns to 40. In Figure 3 we plot the values calculated for  $R_A^v$  (as recorded by the "spy" node  $A$  using Equation (I) for buffer space) versus the values given by the baseline (node density probability mass computed according to the formula presented in [22]), for the same "spy" node. As can be seen in Figure 3, Equation (I) closely follows the baseline values for buffer space in the vicinity of the "spy" node. An explanation for the slight difference between the baseline and the calculated values may be that while baseline values consider a 0 radio range (calculating

only node densities), values calculated using Equation (1) consider a radio range of 20 m.

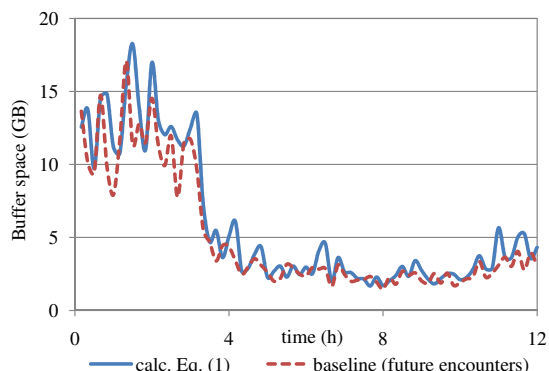
#### 4.2. Validation in a space- and time-varying environment

In the previous subsection, the spatial distribution of resources provided us with a simple baseline. However, if we move on to a space- and time-varying model, or a non-synthetic mobility model, choosing a baseline becomes more complicated. In this subsection we propose two scenarios, with two different baseline alternatives:

- **future encounters**, calculated as the sum of own resources ( $R^o$ ) of all peers that will be actually met by the observed node over a reference timespan\*  $\tau_f$ .
- **cell resources**, calculated as the sum of own resources ( $R^o$ ) of all nodes sharing the same cell as the node itself at a given time. (Cells are obtained by dividing the simulation playground into a number of equal squares.)

##### 4.2.1. Random waypoint scenario

In this scenario, our setting was again a 100-node network performing random waypoint movement in the 1800 m  $\times$  1800 m square playground. We then injected a large number of messages over a short period of time in this network, which reduced buffer space in most nodes to a minimum level. This was followed by a period of slow recovery in buffer space, as the messages were gradually delivered and therefore deleted from the buffers.



**Figure 4.** Available buffer space in the vicinity of a node, time- and space-varying scenario, random waypoint

Figure 4 shows the evolution in time of  $R_A^v$  for a representative node, using buffer space data, as compared to the future encounters baseline. However, comparable accuracy can be found for all the other 99 nodes. If we define inaccuracy  $I$  as the distance between the calculated curve  $c(t)$ , and the baseline curve  $b(t)$ , we can come up with the following formula:

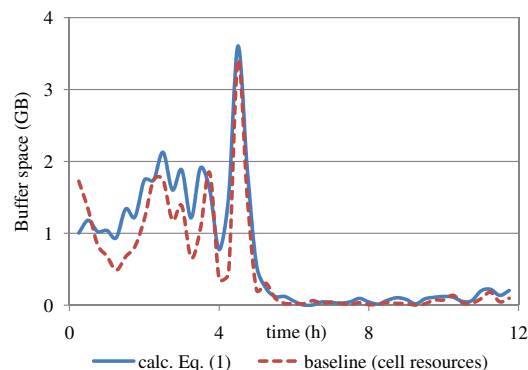
\*For the sake of simplicity we will use  $\tau_f = \tau$

$$I = \frac{\int_0^T |c(t) - b(t)| dt}{\int_0^T b(t) dt} \quad (3)$$

where  $T$  is the simulation time (12h). By applying this formula, we find that inaccuracy  $I$  is between 2% and 6% for all 100 nodes in this scenario, for both baselines proposed: future encounters and cell resources.

##### 4.2.2. Disaster area scenario

We consider a 150-node network moving according to a disaster management scenario as described earlier [9], known as Bonn Motion. In order to create some heterogeneity in the system, nodes were divided into 3 groups, each including 50 nodes. Only the first group of nodes were injecting messages over the first half of the 12 h scenario, addressing them uniformly to all nodes. Buffer space allocation was also uneven: it was 500 MB for the first two groups of nodes, and only 50 MB for the third one. Playground was 360 m  $\times$  170 m and the cell used to calculate the cell resource baseline was 10 m  $\times$  10 m.



**Figure 5.** Available buffer space in the vicinity of a node, time- and space-varying scenario, disaster area

In Figure 5 we present the evolution of  $R_A^v$  for one particular node, calculated for buffer space, compared with the cell resources baseline. For this scenario, maximum inaccuracy as compared to cell resource baseline is 10% and typically below 6%.

We have now demonstrated that the generic notion of virtual vicinity resource can be used in two mobility contexts to estimate buffer resource with a low level of inaccuracy. We have done this considering both time-varying parameters (load and movement) and space-varying parameters (movement).

## 5. EXPLOITING RESOURCE HETEROGENEITY

In Section 3 we proposed the idea that every node can estimate the resource level available in its vicinity

by keeping a small matrix derived from the history of its previous encounters. These were shown to be quite accurate for estimating vicinity resources. Whenever a reference node  $A$  meets a node  $B$ , information about the resources in  $B$ 's vicinity, such as energy ( $E_B^v$ ), buffer space ( $S_B^v$ ), and bandwidth ( $D_B^v$ ) becomes available to  $A$ .

Our next goal is to show how this information can be used to optimise the store-carry-forward scheme in intermittently connected networks. Since our strategy is based on an analysis of information available about three resource types, we have called our approach holistic. We propose here custodian election and queue management policies, as elements of strategy that can contribute to improving overall network performance.

### 5.1. Custodian election and message queue management policies

In a routing or dissemination context, in order to increase the probability of successful message delivery, nodes need to choose the right custodians in the network. For example, choosing a custodian with insufficient amounts of energy may lead to imminent battery depletion, causing not only loss of connectivity in the network, but also loss of all the messages stored in that node's buffer at that particular time.

Custodian election refers to a nodes decision to choose one or more custodians as encounters take place. They should be nodes with the highest probability to carry a message towards its destination. An efficient custodian election policy can then be set up if custodian candidates can signal to elector nodes both the amount of resources found in their vicinity, and their own resource levels. By comparing these two values – own resources versus resource levels available in the vicinity – a node  $A$  will choose node  $B$  as a custodian if and only if  $B$ 's own energy, available buffer space, and bandwidth are relatively abundant as compared to the resources expected in  $B$ 's vicinity. The idea is that a node should choose a particular custodian when there is little evidence that the node may find a better opportunity later.

On the other hand, signalling the amount of resources expected to be available in the neighbourhood of a node may also improve message *queue management*. It is highly important for a node to choose the right-sized message from its message queue in order to actually be able to convey it over the estimated contact window. Starting to transmit a message whose size is uncorrelated with the contact window may increase the risk of partial transmission. Partial transmissions are unwanted phenomena as they imply retransmissions, thus wasting bandwidth and increasing energy requirements [24].

These policies can be viewed from two different perspectives: a *basic* perspective, where the number of custodians chosen cannot be defined and varies according to fluctuating system conditions, and a *controlled* perspective, where the number of custodians is determined as a fraction of the total number of encounters.

### 5.2. The basic perspective

The basic approach has been presented earlier [10], and was demonstrated by implementing a set of custodian election (CE) and queue management (QM) policies on top of Epidemic routing. We can formalise these policies as shown below:

- (CE) When a reference node  $A$  encounters a node  $B$ , node  $A$  elects node  $B$  as a custodian only if its relative strength (own resources versus vicinity resources) is above a specific threshold value. The whole set of resources, energy, buffer space, and bandwidth, are taken into account, as follows: select node  $B$  if:

$$\left(\frac{E_B^o}{E_B^v} > T_E\right) \wedge \left(\frac{S_B^o}{S_B^v} > T_S\right) \wedge \left(\frac{D_B^o}{D_B^v} > T_D\right) \quad (4)$$

where  $T_E, T_S, T_D$  are threshold values for energy, buffer space, and bandwidth respectively.

- (QM) Once custodian  $B$  has been selected according to condition (CE) above, send message  $m$  of size  $s_m$  only if: (1) available energy at  $B$  is above the necessary level required for transmitting  $s_m$  bytes (considering no overhead), (2) available buffer space at  $B$  exceeds  $s_m$ , and (3) available bandwidth, given a particular contact window, allows sending  $s_m$  bytes. Note that queue management policies can be described exclusively as a function of message size  $s_m$  and available resources at node  $B$ . In case one of these conditions does not hold for a message  $m$ , a smaller message should be sent to  $B$ .

Although we will show in Section 5.4 that combining queue management policies with the basic form of custodian election policies has significant benefits, we should not overlook the fact that there are also two main disadvantages associated with this solution:

1. *No control over custodian election ratio.* There is no control over how many custodians are elected out of a (given) number of total meetings. This scheme will elect as many or as few custodians as required, according to the thresholds chosen and the resources available in the vicinity.
2. *Difficulty in setting threshold values.* The question remains open on how to set threshold values ( $T_E, T_S, T_D$  in Equation 4). Of course, these values can be set experimentally, based on a particular system. However, the system may decide at some point to change its initial behaviour. For instance, it suffices for only one of these threshold values to be set too low for a node to potentially defer choosing a custodian indefinitely. It is obvious, however, that such an indefinite deferral would not be a good option for the store-carry-forward process.



### 5.3. The controlled perspective

The original idea presented in this section consists in limiting the number of custodians a reference node  $A$  elects to a specific *fraction* of the nodes it meets. In this approach, only the custodian election policies are modified, while the queue management policies are kept unchanged.

The controlled perspective can also be built on top of Epidemic routing, a typical case of greedy replication. However, since the number of custodians is restricted, replication in this case is no longer greedy. Intuitively, we expect this to result in lower resource usage, as controlled replication is usually seen as an effective means for reducing overhead and still achieving adequate performance.

The objectives of this controlled perspective are the following:

1. To elect as custodians a fraction  $\lambda$  of nodes out of all node encounters.
2. To replicate the message only to custodians holding the best cumulative resources.

We recall from Section 3 that, during an observation time frame  $\tau$  (sliding window), a reference node  $A$  will have approximately  $n_\tau$  encounters with nodes from  $B_i$  to  $B_j$  ( $n_\tau = j - i + 1$ ). Information on the available level of resources at nodes met is stored by node  $A$  in a matrix  $M_A$ . Provided that  $n_\tau$  is big enough, we can also consider that the resource distribution of the nodes encountered by node  $A$  will not vary significantly before and after a new node  $B_{j+1}$  is encountered. Before meeting node  $B_{j+1}$ , and based on  $M_A$ , node  $A$  calculates a resource threshold value, such that a given fraction  $\lambda$  of the  $n_\tau$  nodes fall above this threshold value, while the rest  $(1 - \lambda)$  stay below. When nodes meet, node  $B_{j+1}$  becomes a custodian for node  $A$ , provided that the own resources of  $B_{j+1}$  are above this calculated threshold value. The objective is to make sure that, over any given  $\tau$  period,  $A$  will choose only  $\lambda * n_\tau$  custodians,  $\lambda$  being a protocol parameter, set in advance. Therefore, even though node  $A$  will meet potential custodian nodes one at a time, it will be able to calculate the selection criteria before each encounter, and check them upon encounter with any custodian candidate.

The question remains on how to combine all relevant resources into one abstract and comparable value, thereby capturing our holistic approach to resources. For the generic case, let  $u(E_B^o, S_B^o, D_B^o)$  be a function representing the utility<sup>†</sup> of choosing node  $B$  as a custodian for node  $A$ , where  $E_B^o, S_B^o, D_B^o$  are node  $B$ 's own available levels of energy, storage space, and bandwidth respectively. The utility function represents the benefit an elector node  $A$  obtains from choosing a particular custodian  $B$  instead of some other. Of course,  $u$  should

monotonically increase as a function of the variables  $E_B^o, S_B^o, D_B^o$ , thus indicating the fact that the choice of a node with better energy, buffer space, or bandwidth levels is more beneficial for node  $A$ . On the other hand,  $u$  should also take account of the holistic perspective, and provide adequate weight to the relative importance of energy, buffer space, and bandwidth for the system. For the performance evaluation presented in Section 5.4, we have used a utility function that: 1) gives equal weighting to energy, buffer space, and bandwidth, and 2) increases linearly for each of these resources:

$$u(E_B^o, S_B^o, D_B^o) = E_B^o \times S_B^o \times D_B^o \quad (5)$$

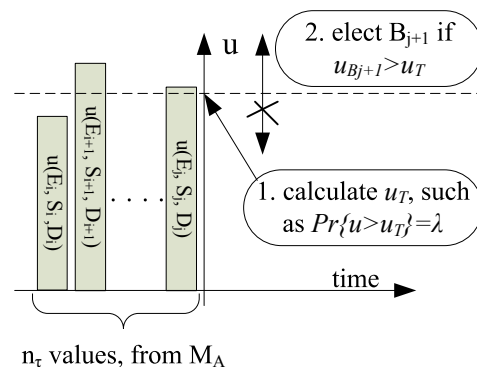


Figure 6. Custodian election in a controlled scenario

As shown in Figure 6, node  $A$  knows the last  $n_\tau$  resource values corresponding to the last nodes met. Knowing function  $u$  and applying it for every node last encountered,  $A$  can determine the threshold value  $u_T$  such that, if the distribution of past resources were to be similar in the near future, approximately a fraction  $\lambda$  of all nodes would be elected. When  $A$  meets a potential custodian node  $B_{j+1}$ ,  $A$  checks if  $u_{B_{j+1}}$  is above the threshold value  $u_T$ . If it is,  $A$  will elect  $B_{j+1}$  as a custodian, otherwise it will not. Matrix  $M_A$  also provides the data structure needed to keep track of the nodes elected and ensures that a fraction close to  $\lambda$  is elected. As  $M_A$  is refreshed at every new encounter, and only a sliding window of size  $n_\tau$  is kept, the system is capable of adapting dynamically to variable network conditions.

Note that the utility function may be more complex in different practical scenarios. For example, either 1) some resources may be given a higher weight as compared to others, for instance nodes with energy resources may be preferred over nodes with buffer space, and/or 2) the function may vary in a nonlinear fashion, for instance utility may decrease more abruptly for resources close to depletion. Under such a scenario, at some point in time a node may value the availability of energy more (as it is aware that its battery is running down, and the next custodian may be its only hope to send a message on), while at some other point in time the same node may

<sup>†</sup> Utility is considered here only from a resource perspective; this approach does not take into account other criteria, such as proximity to destination, social impact, etc.

value the availability of bandwidth more (since it has a particularly heavy payload in its buffer).

#### 5.4. Performance evaluation and discussion

In order to establish the end-to-end impact of the policies presented in Sections 5.2 and 5.3, we have implemented them on top of the Epidemic routing protocol [11]. We have chosen Epidemic routing due to its simplicity and the absence of any a priori queue management and custodian election policies, which makes differences easier to spot. However, it would also be possible to modify some other, potentially more complex, store-carry-forward protocols, as long as the original protocol policies can be combined with integrated information about resource availability.

We used Bonn motion mobility traces [9] and the network configurations described in Table I below. Messages originate from only one of the 50-node groups and are intended for all the 150 nodes in the setup, and they are injected into the system only during the first hour of simulation in order to create the preliminary heterogeneity that is then exploited using the proposed scheme.

Table I. Simulation setup

Group	# nodes	Initial energy $E_{in}$ (kJ)	Initial Buffer (MB)	Initiate messages
1	50	20	500	no
2	50	50	500	yes
3	50	20	50	no
Energy factor for sensing ( $e_s$ ):				0.1J/s
Energy factor for message exchange ( $e_m$ ):				0.1J/kB
Message size (fixed):				10MB
Transmission speed:				0.1MB/s
Simulation period:				12h
Node's radio transmission range:				10m
Playground size:				360m x 170m

The energy model implemented in this simulation corresponds to the theory presented in Section 3.2.2. Available energy at each node diminishes in time as an effect of 1) radio scanning, and 2) message transmission. While the former factor is proportional to elapsed time, the latter is proportional to message size, as described by Equation 2. Initial energy for one node is set between 20kJ and 50kJ, which corresponds roughly to the fully charged battery of a modern smartphone. Because batteries are not recharged during the 12h run, some nodes will be exhausted, and therefore become useless for transmitting messages.

Note that the energy related parameters in this evaluation, presented in Table I, are not modelled based on measurements performed on a specific device. However, the parameters we have selected are corroborated by findings in some recent studies [25]. Of course, device characteristics, radio interface used, and data bursts will

affect the depletion model. Yet, we found that running the same scenario with various radio and energy parameters in several runs largely reproduces the same qualitative results.

In our simulation environment we have compared several curves that are detailed in Table II below.

Figure 7 shows how nodes gradually become useless for the network as their energy is depleted over time. As a consequence of using the highest energy footprint, the Epidemic protocol quickly exhausts most nodes, and the simulation ends with only about 15% of the initial set of nodes. This figure also shows that implementing queue management and custodian election policies is a good way to reduce resource consumption (energy, in this particular case).

When comparing a set of store-carry-forward protocols applied to a scenario where nodes have various levels of available resources, the ideal protocol should satisfy the following intuitive requirements: 1) it should use a small resource footprint to deliver a large number of messages with low latency, and 2) during operation, it should

Table II. Various curves in the simulation environment

id	description
E	= Epidemic (used as baseline)
b:C	= Epidemic + custodian election policy, as described in the <i>basic</i> perspective (Section 5.2). $T_E = T_S = T_B = 1.5$ have been used for this simulation
b:CQ	= Epidemic + custodian election + queue management as described in the <i>basic</i> perspective (Section 5.2)
$c : \lambda$	= Epidemic + custodian election policy, as described in the <i>controlled</i> version (5.3). It elects as custodians a $\lambda$ fraction of nodes out of all nodes met. For instance, c:9 represents the case where only $\lambda = 9\%$ custodians out of all meetings are elected. $\lambda$ is varied in a set of values of 9%, 15%, and 35%

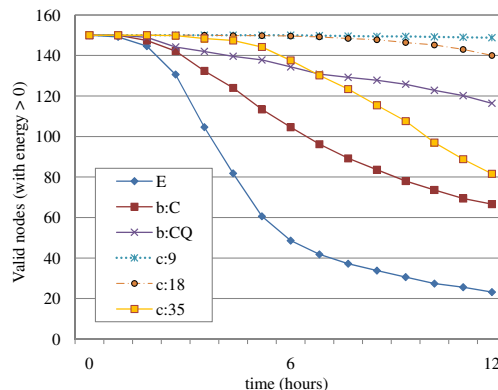
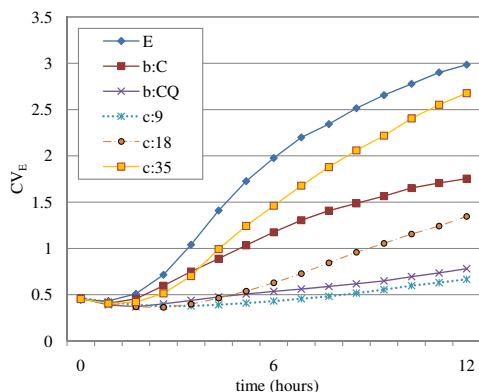


Figure 7. Valid and exhausted nodes over time



**Figure 8.** Variation coefficient for energy available to nodes

attempt to level out disparities between left-over resources available to various nodes. While the first requirement is systematically taken into account in most related studies, the second requirement is somewhat more subtle. The suggestion would be that an ideal protocol should do its best to preserve as many nodes available in the network as possible. In other words, it should make careful use of nodes with scarce resources, and make preferential use of nodes with abundant resources, thereby maximising the number of valid nodes available in the network. On the other hand, the network may suffer if resources are concentrated only in a few nodes, while the remaining nodes undergo exhaustion as an effect of battery depletion, buffer exhaustion, or network congestion. Formally, in order to meet the second requirement, the distribution of available resources should be maintained as uniform as possible across the node population.

In order to understand the various behaviours of our proposed schemes, we have introduced the coefficient of variation (CV) on resources. Let us analyse the case of available energy first, knowing that other resources, such as buffer space and bandwidth, can receive a similar treatment.

$$CV_E = \frac{\sqrt{\frac{1}{N} \sum_1^N (E_i - \bar{E})^2}}{\bar{E}}$$

where:  $CV_E$  = coefficient of variation for energy  
 $N$  = number of nodes (150 in our case)  
 $E_i$  = energy of node  $i$   
 $\bar{E}$  = energy mean value

In Figure 8 we plot the coefficient of variation for energy as a function of time, for all the six curves. This figure shows that custodian election policies, particularly in the controlled version, show smaller energy variance across the node population. Although not shown here, similar shapes can also be found when plotting the coefficient of

variation for buffer space and bandwidth across the node population. This means that custodian election policies will use network resources especially from nodes with above average amounts of resources, and deal gently with nodes whose available assets are below average.

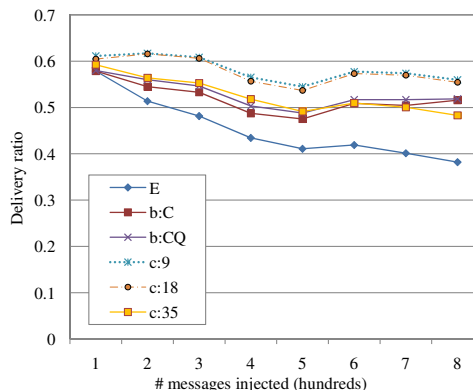
From Figure 7 and Figure 8 we can conclude that our proposed schemes not only use fewer resources, but also maintain a narrow variation of resources in the node population by avoiding node starvation as a consequence of resource exhaustion.

Finally, other network metrics also show significant improvements when queue management and custodian election schemes are used. In order to validate our approach at various loads, we gradually increased the number of messages sent by a factor of 1 to 8, and plotted the results in Figure 9. The delivery ratio is significantly better (around 30%) and there is no significant increase in latency (less than 3%) when best performing curves are used (c:9 and c:18).

A legitimate question concerns the costs incurred (i.e. added overhead) for information exchange between nodes. Costs can be approximated as follows:

- **network costs.** In order to implement custodian election and queue management policies, custodian candidates need to send  $E_A^o/E_A^v$ ,  $S_A^o/S_A^v$  and  $D_A^o/D_A^v$  at each meeting. For calculating the resource level available in the vicinity ( $E_A^v$ ,  $S_A^v$  and  $D_A^v$ ), nodes need to exchange the vector  $I = (\vec{v}, E^o, S^o)$  at each meeting. Considering that each scalar value is expressed by 2 bytes, and each vectorial value by 4 bytes, we can consider that data exchanged by one node at each encounter is 14 bytes.
- **storage costs.** Each node stores a matrix sized  $n_r \times size_I$ . This translates into storage requirements of 320 bytes in the current setting.

These are simple metrics that can be easily compared with aggregated network workload (for all nodes and the whole duration of the experiment) and buffer size. In our settings, aggregated network workload was about  $10^6$  times above



**Figure 9.** Delivery ratio on load

network costs and node average buffer was about  $10^6$  times above storage costs.

## 6. CONCLUSION

In this paper we have proposed a distributed scheme allowing nodes in a partitioned (DTN) network to estimate the energy, buffer space, and bandwidth levels available in their vicinity. Since this strategy is based on an analysis of information available about three resources, the approach is called holistic. We have tested the validity of the proposed model by simulations and have demonstrated a good accuracy of values calculated using the proposed scheme as compared to different baselines.

The optimisation approach we have proposed uses this resource-related information to implement custodian election and message queue management policies. Due to their versatility, these policies may be easily adapted to various store-carry-forward protocols. We have then demonstrated that these policies yield substantial benefits when combined with an Epidemic routing baseline in a disaster management scenario. We have also shown that, by exploiting this information, a routing protocol may not only use up fewer resources overall, but may also consume resources preferentially from nodes with higher resource levels, sparing those with limited supplies when possible. As a result, disparities in available resources across the node population can be significantly reduced, and nodes are less likely to leave the network as a consequence of resource depletion. The solutions we have proposed are particularly beneficial in networks with heterogeneously distributed resources.

Future work includes refinements of the policies by integrating delivery constraints related to message time-to-live versus estimated delivery latency. Moreover, validating this collaboration scheme in a real-life delay-tolerant network would be of great interest. This would also require more extensive physical energy measurements, and obtaining a more accurate picture of idle state energy consumption as well as transmission power. We think that a more detailed simulation of these physical characteristics based on realistic models would be interesting for a more in-depth justification of these techniques.

## REFERENCES

1. Fall K, Farrell S. DTN: an architectural retrospective. *IEEE Journal on Selected Areas in Communications* June 2008; **26**(5):828–836, doi:10.1109/JSAC.2008.080609.
2. Spyropoulos T, Psounis K, Raghavendra CS. Spray and wait: an efficient routing scheme for intermittently connected mobile networks. *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking (WTDN'05)*, 2005; 252–259, doi: 10.1145/1080139.1080143.
3. Lindgren A, Doria A, Schelén O. Probabilistic routing in intermittently connected networks. *Lecture Notes in Computer Science* January 2004; **3126**:239–254.
4. Yoneki E, Greenfield D, Crowcroft J. Dynamics of Inter-Meeting Time in Human Contact Networks. *Proceedings of the 2009 International Conference on Advances in Social Network Analysis and Data Mining (ASONAM'09)*, 2009, doi:10.1109/ASONAM.2009.42.
5. Daly E, Haahr M. Social network analysis for routing in disconnected delay-tolerant manets. *Proceedings of the 8<sup>th</sup> ACM international symposium on Mobile ad hoc networking and computing (MobiHoc'07)*, 2007, doi:10.1145/1288107.1288113.
6. Hossmann T, Spyropoulos T, Legendre F. Know Thy Neighbor: Towards Optimal Mapping of Contacts to Social Graphs for DTN Routing. *Proceedings of the 29<sup>th</sup> IEEE International Conference on Computer Communications (InfoCom'10)*, 2010, doi:10.1109/INFCOM.2010.5462135.
7. Piórkowski M, Sarafijanovic-Djukic N, Grossglauser M. A Parsimonious Model of Mobile Partitioned Networks with Clustering. *Proceedings of the International Conference on Communication Systems and Networks (COMSNETS'09)*, IEEE, 2009, doi: 10.1109/COMSNETS.2009.4808865.
8. Tournoux PU, Leguay J, Benbadis F, Conan V, de Amorim MD, Whitbeck J. The Accordion Phenomenon: Analysis, Characterization, and Impact on DTN Routing. *Proceedings of the 28<sup>th</sup> IEEE International Conference on Computer Communications (InfoCom'09)*, 2009, doi:10.1109/INFCOM.2009.5062024.
9. Aschenbruck N, Gerhards-Padilla E, Gerharz M, Frank M, Martini P. Modelling Mobility in Disaster Area Scenarios. *Proceedings of the 10<sup>th</sup> International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ACM, 2007; 4–12, doi:10.1145/1298126.1298131.
10. Sandulescu G, Schaffer P, Nadjm-Tehrani S. Vicinity Resource Cartography for Delay-Tolerant Networks: a Holistic Perspective. *Third IFIP Wireless Days*, IEEE, 2010.
11. Vahdat T, Becker D. Epidemic routing for partially connected ad hoc networks. *Cs-2000-06*, Duke University July 2000.
12. Grossglauser M, Tse DNC. Mobility Increases the Capacity of Ad Hoc Wireless Networks. *IEEE/ACM Transactions on Networking* August 2002; **10**(4):477–486.
13. Zhang X, Neglia G, Kurose J, Towsley D. Performance modeling of epidemic routing. *Computer Networks: The International Journal of Computer*

- and *Telecommunications Networking* July 2007; **51**:2867–2891.
14. Jacquet P, Mans B, Rodolakis G. Information Propagation Speed in Mobile and Delay Tolerant Networks. *IEEE Transactions on Information Theory* October 2010; **56**(10):5001–5015, doi:10.1109/TIT.2010.2059830.
  15. Balasubramanian A, Levine BN, Venkataramani A. DTN Routing as a Resource Allocation Problem. *ACM SIGCOMM Computer Communication Review* August 2007; **37**(4):373–384, doi:10.1145/1282427.1282422.
  16. Krifa A, Barakat C, Spyropoulos T. An optimal joint scheduling and drop policy for Delay Tolerant Networks. *Proceedings of International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM'08)*, 2008, doi:10.1109/WOWMOM.2008.4594889.
  17. Ramanathan R, Hansen R, Basu P, Rosales-Hain R, Krishnan R. Prioritized Epidemic Routing for Opportunistic Networks. *Proceedings of the First International Mobile Opportunistic Networking MobiOpp (MobiOpp'07)*, 2007, doi:10.1145/1247694.1247707.
  18. Yuan Q, Cardei I, Wu J. Predict and Relay: An Efficient Routing in Disruption-Tolerant Networks. *Proceedings of the 10<sup>th</sup> ACM international symposium on Mobile ad hoc networking and computing (MobiHoc'10)*, 2009, doi:10.1145/1530748.1530762.
  19. Leguay J, Friedman T, Conan V. Evaluating Mobility Pattern Space Routing for DTNs. *Proceedings of the 25<sup>th</sup> IEEE International Conference on Computer Communications (InfoCom'06)*, 2006.
  20. Spyropoulos T, Turletti T, Obraczka K. Routing in Delay Tolerant Networks Comprising Heterogeneous Populations of Nodes. *IEEE Transactions on Mobile Computing* August 2009; **8**(8):1132–1147, doi:10.1109/TMC.2008.172.
  21. Rice A, Hay S. Decomposing power measurements for mobile devices. *Proceedings of the 8<sup>th</sup> IEEE Conference on Pervasive Computing and Communications (PerCom'10)*, 2010, doi:10.1109/PERCOM.2010.5466991.
  22. Hyytiä E, Lassila P, Virtamo J. Spatial node distribution of the random waypoint mobility model with applications. *IEEE Transactions on Mobile Computing* June 2006; **5**(6):680–694.
  23. Keränen A, Ott J, Kärkkäinen T. The ONE Simulator for DTN Protocol Evaluation. *Proceedings of the 2<sup>nd</sup> International Conference on Simulation Tools and Techniques (SIMUTools'09)*, ICST, 2009, doi:10.4108/ICST.SIMUTOOLS2009.5674.
  24. Sandulescu G, Nadjm-Tehrani S. Adding Redundancy to Replication in Window-aware Delay-Tolerant Routing. *Journal Of Communications, Special Issue in Delay Tolerant Networks, Architecture, and Applications* March 2010; **5**, no. 2.

25. Perrucci GP, Fitzek FH, Sasso G, Kellerer W, Widmer J. On the Impact of 2G and 3G Network Usage for Mobile Phones' Battery Life. *Proceedings of the 15<sup>th</sup> European Wireless Conference (EW'09)*, IEEE, 2009; 255–259, doi:10.1109/EW.2009.5357972.

## AUTHORS' BIOGRAPHIES



**Gabriel Sandulescu** received his Ph.D. in Computer Science from the University of Luxembourg in 2011 and a Diploma in Electrical Engineering from the Politehnica University in Bucharest in 1992. He is employed with the European Fund Administration Luxembourg since 1999, where he has been involved in software engineering. His current research interests include new service paradigms in

conjunction with network convergence, resource allocation in mobile networks, and distributed systems.



**Péter Schaffer** received his M.Sc. and Ph.D. degrees in Computer Science from the Budapest University of Technology and Economics (BME) in 2005 and 2009, respectively. During his Ph.D. studies, he was member of the Laboratory of Cryptography and Systems Security (CrySyS). Since October 2009, Péter holds a Research Associate position at

the Interdisciplinary Centre for Security, Reliability and Trust (SnT) at University of Luxembourg. Péter's research interests are in computer security, privacy and reliability, especially in relation to large-scale networks and systems.



**Simin Nadjm-Tehrani** is a full professor at Linköping University (LiU), Sweden. She leads the real-time systems laboratory (RTSLAB) at the department of Computer and Information Science, LiU, since 2000. She obtained her BSc from Manchester University, England, and her PhD from Linköping University in 1994. Her current

research interests lie in the area of dependable distributed systems and networks, including analysis of safety and fault tolerance, reliable communication and measuring availability in presence of failures, attacks and overloads.