Routing metrics in Delay Tolerant Networks

Hugo Cruz Sánchez, Laurent Franck and André-Luc Beylot

***Abstract*—Routing in telecommunication networks rely on the definition of performance indicators called routing metrics. Popular routing metrics are the route throughput, end-to-end delay and jitter. However, in a delay tolerant network displaying link disruptions, these metrics may be questionable. This contribution proposes six metrics for characterising routes in a delay tolerant network. These metrics are: route lifetime, end-to-end delay, capacity, synchronicity and simultaneousness. Some of these metrics are borrowed (and possibly adapted) from classical networks, others are new. Each metric is formalised and examplified by showing how the use of these metrics helps to compute routes which are adequate for applications that may be deployed over delay tolerant networks.**

***Index Terms*—Routing, metrics, delay tolerant networking.**

I. Introduction

Routing consists in computing a path from a source to a (set of) destination(s). Each path displays characteristics called metrics. During route computation, these metrics are optimised in order to fulfill the requirements of the supported services. In today's networks (that we shall call "classical networks"), the metrics considered are mainly the end-to-end delay, the route length expressed as a number of hops and the throughput.

This contribution addresses routing metrics for Delay Tolerant Networks (DTN) [1] which - unlike classical networks - allow the storage of messages in intermediate nodes for an arbitrary duration. Delay tolerant networks are an instantiation of store and forward networks. Examples of networks based on a delay tolerant architecture [2] may include: interplanetary communication, low earth orbiting satellites [3], sensor networks with data mules [4]. In such environments, storing messages contributes to mitigate the adverse effects of intermittent connectivity (ex. due to mobility), limited resources or very long propagation delays. A general list about delay tolerant network projects can be found at the IRTF-Delay Tolerant Network Research Group web site [5].

The specificities of delay tolerant networks call for revisiting routing [6] and associated metrics. Some metrics used in classical networks may not longer hold and it is likely that new metrics will be required. For example, in classical networks, the number of hops is used as an indicator of the end-to-end delay: the more hops, the larger the end-to-end delay. In a delay tolerant network, this approximation is flawed: some hops may take a long duration (due to storage) while other hops are quasi immediate.

H. Cruz-Sánchez and L. Franck are with ENST Bretagne - site of Toulouse, 10 avenue E. Belin, F-31028 Toulouse Cedex. Corresponding author: Laurent.Franck@enst-bretagne.fr

A-L. Beylot is with ENSEEIHT. E-mail: Andre-Luc.Beylot@enseeiht.fr

The analysis and evaluation of the routing policies and performance within delay tolerant networks may be extended from classical metrics scheme [7], [8] in order to taken into account the store and forward behavior of the networks. This idea has been partially developped in some recent works [9], [10], [11].

The contribution is organised as follows: Section II describes the metrics commonly used in classical networks, their purpose and possible shortcomings when applied to delay tolerant networks. Section III introduces the concepts required for defining routing metrics applicable to delay tolerant networks. Section IV defines these metrics, providing an informal and then formal definition according to the framework referenced above. Examples of routing policies based on the metrics described in this contribution are given in Section V. The contribution ends with a conclusion and perspectives on how to extend this study.

II. Routing metrics in classical networks

In classical networks, the following routing metrics are popular: number of hops, end-to-end delay, delay jitter, route lifetime and route throuput [12]. These metrics are optimised during route computation in order to fulfill the requirements of the supported services. While other metrics do exist (the Internet Type of Service field covers also monetary and reliability aspects) there a not often used. The following paragraphs browse through these common metrics and see whether they are directly applicable to delay tolerant networks.

A. Number of hops

The number of hops corresponds to the number of nodes minus one on the path from the source to the destination. In classical networks, minimising the number of hops results in end-to-end delay optimisation with the advantage of using a straightforward metric. Relying on the actual end-to-end delay would require periodic measurements. In delay tolerant networks this shortcut no longer holds: because of possible storage, some hops may take significantly more time. As a result, two routes with the same number of hops do not always display similar end-to-end delays.

B. End-to-end delay

The end-to-end delay is the time required for a packet to reach the destination assuming it is measured from the source to the destination application layer. The end-to-end delay is made of propagation delays and processing delays on the route. The processing delay includes the time needed to switch a message inside an equipment plus the time required to send messages through the outgoing interface. In delay tolerant networks, the end-to-end delay also encompasses the storage duration [11] and depends, as demonstrated in Section IV-B, on the timing of issue of the messages.

C. Delay jitter

The delay jitter measures the difference in end-to-end delay for subsequent messages on a given path. Delay jitter is important for real time services because a high jitter requires buffering at the destination in order to deliver data in a smooth (jitter-less) way. Because delay tolerant networks are not meant to support real time services, delay jitter is not relevant in this context.

D. Route lifetime

The route lifetime expresses the duration a route can be properly used. In classical networks, route lifetime is usually high. On the contrary, in an environment where link and node outages are possible such as in a delay tolerant network, the route lifetime may be significantly shortened. However, as it will be described in Section IV-A, the notion of lifetime has to be refined.

E. Route throughput

The route throughput denotes the number of bytes transported from the source to the destination per unit of time. It is based on the end-to-end delay and the throughput offered by the least capable link. Route throughput in delay tolerant networks may not be representative, because of possible asynchronous operations. Section IV-C proposes an alternate metric.

The previous paragraphs give hints that classical routing metrics are not best suited to delay tolerant networks. Furthermore, it is expected that new metrics are required in order to capture the asynchronous specificities of delay tolerant networks. Similar issues exist in wireless sensor networks where the energy used on a path is sometimes more relevant than the end-to-end delay [13]. In the next Section, we present the terms and concepts required for (re)defining routing metrics applicable to delay tolerant networks.

III. Terms and concepts

Several assumptions are needed in order to delimit the scope of this study. First, only unicast routes are considered. Second, the network nodes are fixed or mobile and may be switched on and off for power saving purposes. Partitioning is therefore possible during the network lifetime and is tackled by message storage in network nodes.

Within that context, a route $R(0,n)$ from the source (node 0) to the destination (node n) is a sequence of $hop(i)$ with $i : 1...n$. Each $hop(i)$ is said to be *feasible* only during the time window $[t_i, t'_i]$. *Feasible* means that if the message leaves node $i-1$ at an instant comprised in $[t_i, t'_i]$, it will be successfully received at node i after a constant propagation delay p_i (ie. between $[t_i + p_i, t'_i + p_i]$).

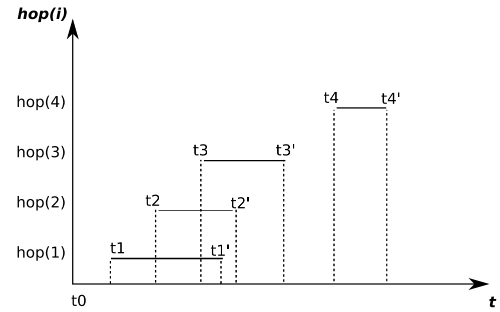


Fig. 1. An example of hop vs time diagram

For $hop(i)$ with $i : 1...n-1$:

1) $t'_i + p_i \leq t'_{i+1}$. If not, t'_i can be set to $t'_{i+1} - p_i$ without loss of generality.
2) $t_i + p_i \leq t_{i+1}$. If not, t_{i+1} can be set to $t_i + p_i$ without loss of generality.

The p_i term can also encompass transmission delays when necessary (i.e. when the transmission delay is not negligible). The hop definition also considers that *for a given route* a hop does not repeat over time (i.e. there is a single time window). While it does not compromise the results, it yields metrics that are easier to formalise and then use. Figure 1 shows an example of a hop vs time diagram as used throughout the contribution. Time t_0 is the start of the session, ie. the time the route is computed. It may be possible that $t_0 < t_1$ because the first hop is not yet feasible at the start of session. In such a case, storage takes place in the source node. As far as routing metrics are concerned, t_1 (corresponding to the first instant the route is used) is the reference time throughout this document. Indeed, t_0 depends on the behaviour of the user and this aspect is decoupled from the route computation.

IV. Routing metrics for delay tolerant networks

This section defines six metrics applicable to delay tolerant networks: route lifetime, end-to-end delay, capacity, synchronicity, simultaneousness and discontinuity. Each description covers an informal presentation, a discussion on the difference with classical networks and a formal definition according to the concepts above. Among these, three metrics have no equivalent in classical networks and their complementarity is shown. The section ends with a discussion on the relation between the route metrics and route computation.

A. Lifetime

The route lifetime denotes the time during which a route can be used without re-computation. Because classical networks display short end-to-end delays (hundreds of milliseconds at most), interactive protocols can be deployed and the validity of a route assumes the simultaneous feasibility of all hops. On the contrary, the failure of a single hop yields the failure of the whole route. In a delay tolerant network, the failure of an egress hop is harmless, provided all messages have already gone past that node. The classical definition of route lifetime must therefore be revisited in order to introduce a location dimension (i) in addition to the existing time dimension (t).

The route lifetime $RLF_i^t(R)$ is defined as:

$$RLF_i^t(R) = t'_{i+1} - t \quad (1)$$

where:

$$\begin{cases} t_i + p_i \leq t \leq t_{i+1} & i : 1...n-1 \\ t_1 \leq t \leq t'_1 & i = 0 \end{cases}$$

For commodity, the end-to-end route lifetime corresponds to the cumulated lifetime of all hops:

$$RLF_{E2E}(R) = t'_n - t_1 \quad (2)$$

It is used as normalising factor in subsequent metrics.

B. End-to-end delay

The end-to-end delay measures the time taken for a message to journey from the source to the destination. This contribution considers the interface to interface delay, that is duration between transmission by the source interface and the reception by the destination interface. In a classical network, variations of the end-to-end delay are bounded by the variations of the propagation and/or processing delays. In a delay tolerant network, the time of departure of the message also impacts the experienced end-to-end delay, regardless of the propagation and processing delays.

Figure 2 illustrates this effect: the time of issue t for a message at the source is comprised between $[t_1, t'_1]$. Propagation delays p_i are assumed to be small compared to $[t_i, t'_i]$ for the simplicity of the example. This interval can be split into subintervals yielding different end-to-end delays:

1) $t \in [t_1, t_3 - p_1 - p_2[$ yields an end-to-end delay of $p_3 + (t_3 - t)$ and includes some mandatory storage.
2) $t \in [t_3 - p_1 - p_2, t'_1]$ yields an end-to-end delay equal to $p_3 + (p_1 + p_2)$ which is minimal. No storage is required.

The minimum $E2E^m(R)$ and maximum $E2E^M(R)$ end-to-end delays are defined as:

$$E2E^m(R) = \begin{cases} p_1 + ... + p_n & \text{if } t'_1 \geq t_n - (p_1 + ... + p_{n-1}) \\ p_n + (t_n - t'_1) & \text{otherwise} \end{cases}$$

$$E2E^M(R) = p_n + (t'_n - t_1)$$

Minimum and maximum mean that considering the constraints expressed and whatever the timing of message issue, a lower or larger end-to-end delay, respectively, cannot be achieved.

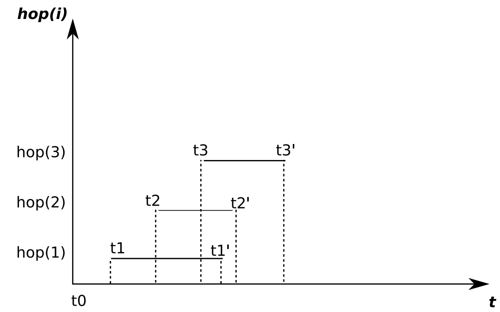


Fig. 2. A route where the time of message issue impacts the end-to-end delay.

C. Capacity

The route capacity denotes the amount of data a route is able to transfer from the source to the destination during its lifetime.

Route capacity $CAP(R)$ is defined as:

$$CAP(R) = min_i(CAP_i(R)) \ ; \ i : 1...n \quad (3)$$

where $CAP_i(R)$ is the capacity of $hop(i)$ computed as the amount of data that can be sent between node $i-1$ and node i during $[t_i, t'_i]$.

Considering that route lifetime is usually high in classical networks, it is not surprising that throughput (an instantaneous capacity) is chosen as routing metric. On the other hand, route capacity in a delay tolerant network provides two hints: (a) if a source is issuing an amount of data larger than the route capacity, a subsequent route will be required and (b) the route capacity sets an upper bound on the storage capability required in every node of the route.

D. Synchronicity

Delay tolerant networks mitigate outages of connectivity through the storage of messages. Still, there are times where connectivity between a source and a destination is complete and a “classical” communication scheme possible. Provided propagation delays are within service limits, real-time traffic is also supported. Route synchronicity denotes the - normalised - duration where classical operations are possible.

Route synchronicity $SYN(R)$ is defined as:

$$SYN(R) = \frac{GTWO(R)}{RLF_{E2E}(R)} \quad (4)$$

with $SYN(R) \in [0, 1]$.

$GTWO(R)$ is the duration of time window overlap, if any, for all $hop(i)$

$$GTWO(R) = \begin{cases} t'_1 - t_n + \sum_{j=1}^{n-1} p_j & \text{if } t_n + \sum_{j=1}^{n-1} p_j < t'_1 \\ 0 & \text{otherwise} \end{cases}$$

Taking advantage of synchronicity by issuing messages during $[t_n, t'_1]$ ensures that:

1) No mandatory storage takes place.
2) The route may be "reversible" up to r round trips if (a) $t_n + 2 * r * \sum_{i=1}^{n} p_i \leq t'_1$ and (b) the underlying technology permits it.

For example, in Figure 2, $SYN(R) = (t'_1 - t_3)/(t'_3 - t_1)$ considering negligible propagation delays p_i. Classical networks have a synchronicity always set to 1. If not, the route is considered broken.

E. Simultaneousness

Synchronicity measures how hops overlap on an end-to-end basis. Simultaneousness measures how subsequent hops overlap. As such, it might be considered as a downgraded synchronicity.

Route simultaneousness $SIM^2(R)$ is defined as:

$$SIM^2(R) = \frac{\sum_{i=1}^{n-1} SIM_i^2(R)}{(n-1)RLF_{E2E}(R)} \tag{5}$$

with $SIM^2(R) \in [0, 1]$.

$\sum SIM_i^2(R)$ is the sum of overlap durations when two hops are not disjoint: $\sum_{i=1}^{n-1}(t'_i + p_i - t_{i+1})$ for i fulfilling $t'_i + p_i > t_{i+1}$

In classical networks, all routes have a $SIM^2(R) = 1$. A delay tolerant network routing algorithm might only select routes with a $SIM^2(R) > 0$ in order to support some hop by hop reliable bundle transfer (subject to p_i).

F. Higher order simultaneousness

The simultaneousness defined previously takes into account sequences made of two hops. This concept can be extended to k subsequent hops, n-th order simultaneousness corresponding to the synchronicity metric defined before.

Simultaneousness of order k, $SIM^k(R)$ is defined as:

$$SIM^k(R) = \frac{\sum_{i=1}^{n-(k-1)} SIM_i^k(R)}{(n-(k-1))RLF_{E2E}(R)} \tag{6}$$

with $2 \leq k \leq n$ and $SIM^k(R) \in [0, 1]$.

$SIM_i^k(R)$ is the overlap duration for k hops starting from node i.

$$SIM_i^k(R) = \begin{cases} t'_i - t_{i+(k-1)} + \sum_{j=0}^{k-2} p_{i+j} & \text{if } t_{i+(k-1)} + \sum_{j=0}^{k-2} p_{i+j} < t'_i \\ 0 & \text{otherwise} \end{cases}$$

High order simultaneousness helps to implement efficiently reliable transfer. Indeed, in a DTN, reliability is supported by means of storage in the transmitting node until acknowledgement by the receiving node. In the worst case, there is no simultaneousness and storage must happen in every node but the last one. However, as soon as second order simultaneousness is available (i.e. simultaneousness among three nodes), the middle node is not required to store the bundle. The larger the simultaneousness, the least number of nodes required to store bundles. If storage capacity is a concern and/or memory writing operations are slow compared to the network speed, the routing algorithm has to favour routes with high order simultaneousness. On the other hand, if network links are unreliable and likely to fail unexpectedly, high order simultaneousness should not be exploited.

G. Discontinuity

Discontinuity is the dual metric of simultaneousness and expresses the normalised duration of mandatory storage. Figure 3 shows two situations where mandatory storage occurs: fully and partially disjoint hops.

Similarly to end-to-end delay, the duration of mandatory storage depends on the time of issue of messages.

The minimum $DSC^m(R)$ and maximum $DSC^M(R)$ route discontinuities are defined as:

$$DSC^m(R) = \frac{\sum_{i=1}^{n-1} MSTO_i^m(R)}{RLF_{E2E}(R)}$$

where:

$$MSTO_i^m(R) = \begin{cases} t_{i+1} - (t'_i + p_i) & \text{if } t'_i + p_i < t_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$DSC^M(R) = \frac{\sum_{i=1}^{n-1} MSTO_i^M(R)}{RLF_{E2E}(R)}$$

where:

$$MSTO_i^M(R) = t_{i+1} - (t_i + p_i)$$

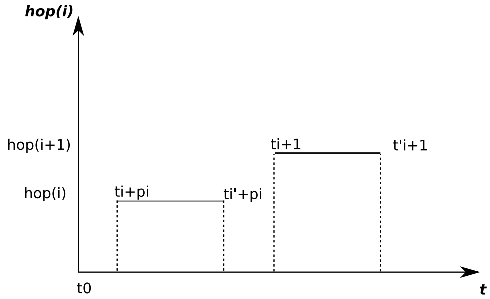
In classical networks, all valid routes have a $DSC_{m,M}(R) = 0$. In a delay tolerant network, a resource saving routing algorithm might optimise routes according to the $DSC^m(R)$ metric.

H. Relation between synchronicity, discontinuity and simultaneousness

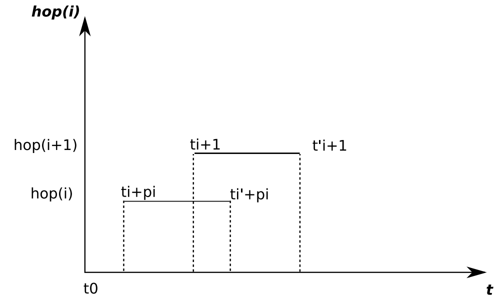
Synchronicity, discontinuity and simultaneousness are tightly interlaced metrics. Their relations shown in Table I help to refine the continuum existing between classical networks and delay tolerant networks.

Classical networks feature only fully connected routes: $SYN(R) = 1$ and storage is not mandatory (Figure 4a). In a delay tolerant network ($SYN(R) < 1$), three additional categories of routes may also be found:

- Fully disconnected routes require storage for every hop: $SYN(R) = 0$, $SIM^2(R) = 0$ (Figure 4b).
- Partially disconnected routes require storage for some but not all hops: $SYN(R) = 0$, $SIM^2(R) > 0$, $DSC^m(R) > 0$ (Figure 4c).

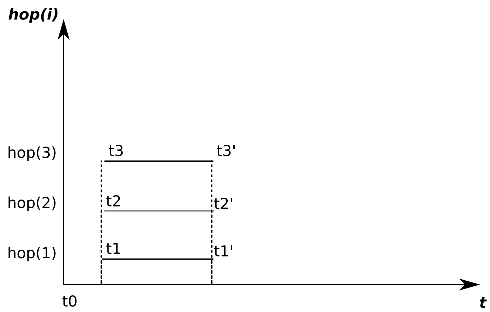


(a) Fully disjoint: $t_i + p_i < t_{i+1}$ and $t'_i + p_i < t_{i+1}$

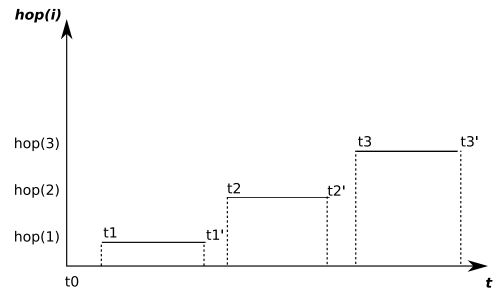


(b) Partially disjoint: $t_i + p_i < t_{i+1}$

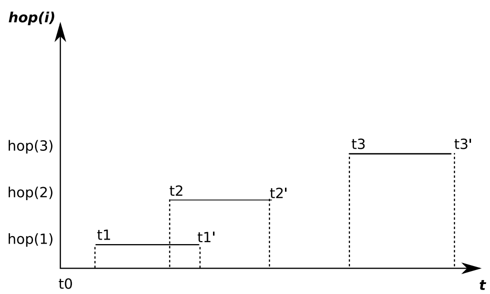
Fig. 3. Fully and partially disjoint hops



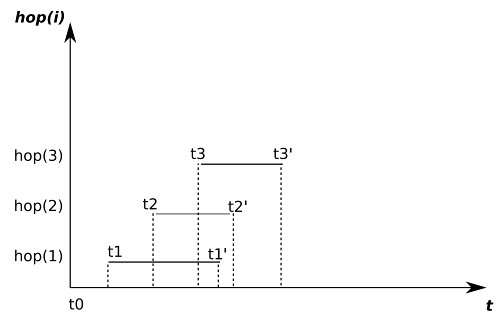
(a) Fully connected



(b) Fully disconnected



(c) Partially disconnected



(d) Partially connected

Fig. 4. Fully connected, fully disconnected and partially disconnected routes. Propagation delays are considered negligible compared to $[t_i, t'_i]$.

- Partially connected routes may require storage depending on the timing of message issue: $SYN(R) > 0$, $SIM^2(R) > 0$, $DSC^m(R) = 0$ (Figure 4d).

The use of synchronicity, simultaneousness, discontinuity and route capacity for storage dimensioning purposes has still to be investigated.

I. Route metrics and route computation

All routing metrics have an end-to-end scope. Proper routing decisions based on these metrics must therefore benefit from a complete knowledge of the route [14]. Routing protocols fall in two categories: source routing where the route is fully computed in the source node and hop by hop routing where the route computation is updated at each hop. When an end-to-end optimisation of the metric is a strong requirement, source routing is mandatory. However, source routing, in order to be effective, requires the source node to have an accurate perception of the state of the network. When the network topology is dynamic and non deterministic, this is usually achieved at the expense of signalling. For this reason, hop by hop routing is often deployed, resulting in sub-optimal routing decisions (because of subsequent hop-by-hop approximations).

As far as delay tolerant networks are concerned, the same problem exists. Either the network state is deterministic or at least computable in advance: low earth satellite constellations, deep space links, sensor networks with known duty cycle are examples of such networks. If not, a signalling scheme or hop by hop routing is required. The transposition of the delay tolerant routing metrics to hop by hop routing is still an open issue.

V. EXAMPLES OF ROUTING POLICIES

This section describes how routing metrics are used to set up routing policies which in turn are required to properly support given services. Figure 5 shows a scenario of possible network connectivity where several subsequent hops numbered from 1

TABLE I
RELATIONS AMONG SYNCHRONICITY, DISCONTINUITY AND SIMULTANEOUSNESS

$SYN(R) = 0$	$\Rightarrow SIM^2(R) \in\ [0,1[$ $\Rightarrow DSC^m(R) \in\ [0,1[$ $\Rightarrow DSC^M(R) \in]0,1[$
$SYN(R) = 1$	$\Rightarrow SIM^2(R) = 1$ $\Rightarrow DSC_{m,M}(R) = 0$
$SIM^2(R) = 0$	$\Rightarrow DSC^m(R) \in]0,1[$ $\Rightarrow DSC^M(R) \in]0,1[$ $\Rightarrow SYN(R) = 0$
$SIM^2(R) = 1$	$\Rightarrow DSC_{m,M}(R) = 0$ $\Rightarrow SYN(R) = 1$
$DSC^m(R) = 0$	$\Rightarrow SYN(R) \in\ [0,1]$ $\Rightarrow DSC^M(R) \in [0,1[$ $\Rightarrow SIM^2(R) \in]0,1]$
$DSC^M(R) = 0$	$\Rightarrow SYN(R) = 1$ $\Rightarrow DSC^m(R) = 0$ $\Rightarrow SIM^2(R) = 1$
$DSC^m(R) \rightarrow 1$	$\Rightarrow SYN(R) = 0$ $\Rightarrow DSC^M(R) \rightarrow 1$ $\Rightarrow SIM^2(R) \rightarrow 0$
$DSC^M(R) \rightarrow 1$	$\Rightarrow SYN(R) = 0$ $\Rightarrow DSC^m(R) \in]0,1[$ $\Rightarrow SIM^2(R) \rightarrow 0$

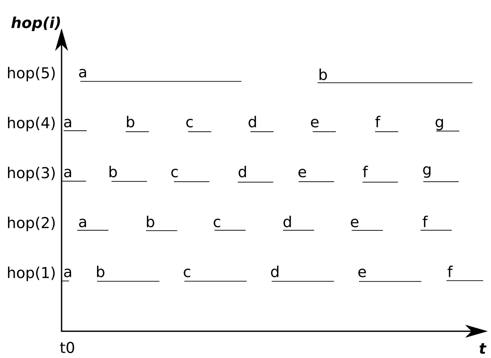


Fig. 5. Subsequent hops yield different possibilities for route selection.

to 5 are available intermittently. Hop number i denotes the transition from node $i-1$ and i as before. Propagation and transmission delays are considered negligible for the ease of the examples. Considering a given application and service to deploy, a routing algorithm will select the right sequence of hops according to metrics which ensure proper operation of the service.

A. Emergency beacons

Emergency beacons are used in the maritime and mountain environments. Upon triggering, they send a signal intended to (a) call for rescue and (b) help localising the user. The amount of information carried in such signals is usually low, the primary constraint is to reach the co-ordination centre rapidely. Figure 6a shows a route where the end-to-end delay metric is minimized (the sequence of hops is $1a-2a-3a-4a-5a$).

B. Software radio

The advent of powerful DSPs makes now possible the processing in software of traditionnaly hardware based operations like demodulation. This is called software (or cognitive) radio. A major advantage of such systems is the possibility to change demodulation algorithms through a simple software (firmware) upgrade. Uploading a new firmware requires to transfer large block of data hence the need for high capacity routes in order to avoid harmful route handovers during firmware transmission. Figures 6b and 6c show two routes maximising the capacity ($1b-2b-3e-4f-5b$ and $1b-2b-3c-4c-5a$). However, the route of Figure 6c has a significant advantage: it presents a lower discontinuity hence reducing the amount of mandatory storage. Considering that possibly large amount of data are involved, this is a plus. Capacity optimisation is strongly linked to discontinuity optimisation.

C. Reliable data transmission

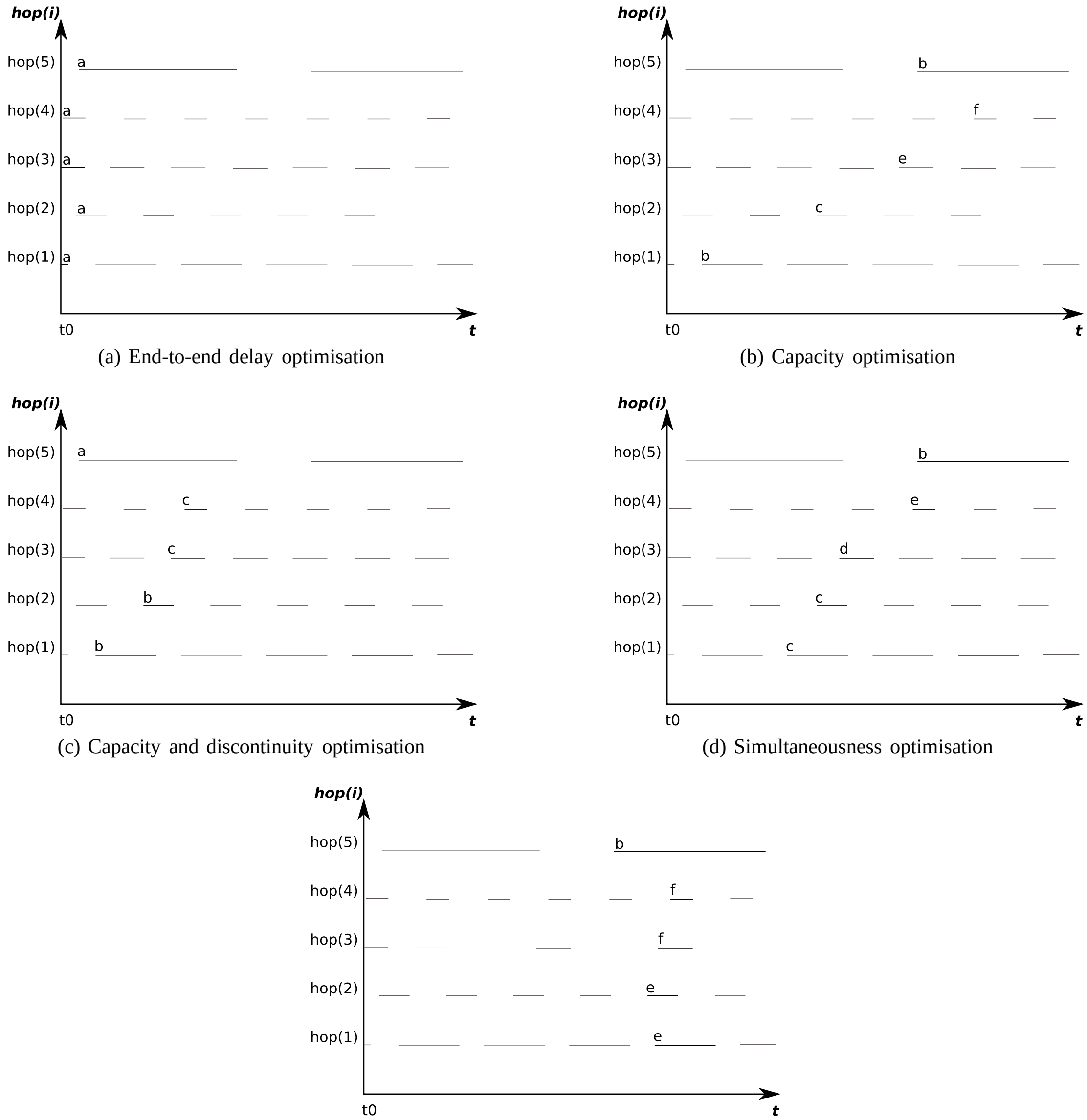
Reliability in a DTN is handled by means of custody and acknowledged exchanges among subsequent custodian nodes. This mechanism assumes that the sending node does not release custody until the data is successfully transmitted to the next custodian. Assuming that a route displays significant discontinuity (like in Figure 6b) storage is necessary in every node with the expected impact on memory requirements. Having routes featuring k^{th} order simultaneousness may help with that respect since storage is only required in the first and last node of the simultaneous segment. Figure 6d shows an example of 3^{rd} order simultaneousness where the route is made of two segments: $1c-2c-3d$ and $3d-4e-5b$. Custody is only required in nodes 0 and 3.

D. Session key exchange

Implementing secure communication between two nodes often requires a shared and secret session key. Protocols based on the Diffie-Hellman algorithm achieve mutual knowledge of a shared secret without the need to openly transmit the secret on the supposedly insecure communication channel. Using routes with synchronicity ensures a timely exchange of information during the Diffie-Hellman based protocol. Figure 6e shows an example of route ($1e-2e-4f-4f-5b$) displaying synchronicity (that is 5^{th} order simultaneousness).

VI. CONCLUSION

This contribution has investigated different routing metrics that impact route computation in a delay tolerant network. Because of the specificities of delay tolerant networks, a simple investigation reveals that (a) metrics used in classical networks are not always applicable to delay tolerant networks and (b) classical networks miss metrics important to delay tolerant networks. The use of traditional metrics to define routing strategies obfuscates some phenomena affecting the performance. Designing and evaluating routing must include



(a) End-to-end delay optimisation

(b) Capacity optimisation

(c) Capacity and discontinuity optimisation

(d) Simultaneousness optimisation

(e) Synchronicity optimisation

Fig. 6. Examples of routes according to different metrics optimisations.

the metrics derived from the delay tolerant networks nature.

Six metrics are identified and formalised: route lifetime, end-to-end delay, capacity, synchronicity, simultaneousness and discontinuity. These metrics are used to measure the inherent characteristics of a route. Other subjective metrics may also be defined (those related to energy, monetary cost and security for example), however they do not fall in the scope of this contribution.

The contribution ends with four examples showing how these metrics can be used to devise routing policies suitable for applications like emergency beacons, software radio, reliable data communication and session key exchange.

Future work covers the use of these metrics and its influence in applying them in present delay and tolerant mechanisms [15] and routing algorithms [16]. More specifically, an area worth investigating is how these metrics can be applied to environments where the communication opportunities are stochastic. As far as deterministic topologies are concerned, we started a study where the proposed metrics are applied to routing in a network of store and forward low earth orbit satellites with routing paradigms from the transportation networks field [17], [18].

References

[1] S. Farrell and V. Cahill, *Delay -and Disruption- Tolerant Networking*. Norwood, MA, USA: Artech House, Inc., 2006.

[2] V. G. Cerf, S. C. Burleigh, R. C. Durst, K. Fall, A. J. Hooke, K. L. Scott, L. Torgerson, and H. S. Weiss, "Delay-tolerant networking architecture (rfc 4838)," April 2007.

[3] L. Wood, W. M. Eddy, W. Ivancic, J. McKim, and C. Jackson, "Saratoga: a delay-tolerant networking convergence layer with efficient link utilization," in *IWSSC 2007, Third International Workshop on Satellite and Space Communications*, September 2007.

[4] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: modeling a three-tier architecture for sparse sensor networks," in *Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on*, 2003, pp. 30–41.

[5] "Delay tolerant networking research group: http://www.dtnrg.org/. viewed on june 15th 2007." [Online]. Available: http://www.dtnrg.org/wiki

[6] Z. Zhang, "Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: Overview and challenges," *IEEE Communications Surveys & Tutorials*, vol. 8, no. 1, pp. 24–37, 2006.

[7] R. Draves, J. Padhye, and B. Zill, "Comparison of routing metrics for static multi-hop wireless networks." in *SIGCOMM*, 2004, pp. 133–144.

[8] J. Leguay, T. Friedman, and V. Conan, "Evaluating mobility pattern space routing for dtns," Nov 2005.

[9] D. Marasigan and P. Rommel, "Mv routing and capacity building in disruption tolerant networks," in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol. 1, 2005, pp. 398–408 vol. 1.

[10] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *MobiHoc '04: Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*. New York, NY, USA: ACM Press, 2004, pp. 187–198.

[11] E. P. C. Jones, L. Li, and P. A. S. Ward, "Practical routing in delay-tolerant networks," in *WDTN '05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. New York, NY, USA: ACM Press, 2005, pp. 237–243.

[12] M. E. Steenstrup, Ed., *Routing in communications networks*. Hertfordshire, UK, UK: Prentice Hall International (UK) Ltd., 1995.

[13] T. Small and Z. J. Haas, "Resource and performance tradeoffs in delay-tolerant wireless networks," in *WDTN '05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. New York, NY, USA: ACM Press, 2005, pp. 260–267.

[14] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," in *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, vol. 34, no. 4. ACM Press, October 2004, pp. 145–158.

[15] K. Scott and S. Burleigh, "Bundle protocol specification," jan 2006. [Online]. Available: http://tools.ietf.org/id/draft-irtf-dtnrg-bundle-spec-03.txt

[16] A. Lindgren, A. Doria, and O. Schel, "Probabilistic routing in intermittently connected networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 7, no. 3, pp. 19–20, 2003.

[17] G. Desaulniers and D. Villeneuve, "The shortest path problem with time windows and linear waiting costs," *Transportation Science*, 2000.

[18] H. Cruz-Sanchez, L. Franck, and A.-L. Beylot, "Precomputed routing in a store and forward satellite system," in *VTC Fall 07: Proceeding of the 2007 IEEE 66th Vehicular Technology Conference*, 2007.