Performance of Distributed Algorithms in DTNs: Towards an Analytical Framework for Heterogeneous Mobility

Andreea Picu
Communication Systems Group
ETH Zürich, Switzerland
Email: lastname@tik.ee.ethz.ch

Thrasyvoulos Spyropoulos
Mobile Communications
EURECOM, France
Email: firstname.lastname@eurecom.fr

***Abstract*—*Opportunistic* or *Delay Tolerant Networks* (DTNs) are envisioned to complement existing wireless technologies (cellular, WiFi). Wireless peers communicate when in contact, forming a network "on the fly", whose connectivity graph is highly dynamic and only partly connected. Because of this stringent environment, solutions to common networking problems (routing, congestion control, etc.) in this context are greedy, choosing the best solution among the locally available ones. This shared trait motivates the common treatment of such greedy algorithms for DTNs and raises some interesting questions: Do they converge? How fast are they? Yet, existing models study individual solutions. Moreover, they often assume homogeneous node mobility. The study of real world traces reveals considerable heterogeneity and non-trivial structure in human mobility. While algorithms have been proposed, accounting for this heterogeneity, their analytical tractability is still a challenge. In this paper, we propose a new *model for greedy DTN algorithms*, supporting the full *heterogeneity of node mobility*. We provide closed form solutions for crucial performance metrics (delivery probability and delay) and prove necessary and sufficient conditions for algorithm convergence. For illustration, we apply our model to the content placement problem, a variant of distributed caching. We use real and synthetic mobility traces to validate our findings and examine the impact of mobility properties in depth.**

I. Introduction

Opportunistic or *Delay Tolerant Networks* (DTNs) are envisioned to augment existing wireless infrastructure-based services (e.g., offload cellular data traffic), and enable novel applications. Nodes harness unused bandwidth by exchanging data whenever they are in proximity (*in contact*), with the goal to forward data probabilistically closer to destinations. By introducing redundancy (e.g., coding or replication) and intelligent mobility prediction algorithms, data of interest can be delivered or retrieved over a sequence of such contacts, despite the lack of end-to-end paths.

Many challenging problems arise in this context: unicast and multicast routing [1], [2], resource allocation [3], [4], content placement [5], etc. Considering the disconnected and highly dynamic nature of the connectivity graph of Opportunistic Networks, these problems are substantially more difficult here, than in traditional connected networks. As a result, most solutions proposed for each problem are intuitive greedy algorithms that deterministically choose the best available action locally. This underlines the need for a deeper, joint analysis of greedy schemes for DTNs. Yet, almost all existing studies deal with individual problems and solutions.

What is worse, these studies rely on simple mobility models (e.g., Random Walk, Random Waypoint, Random Direction), for tractability reasons [4], [6]–[8]. In these models, node mobility is stochastic and *independent identically distributed* (IID). Then, Markov chain theory or fluid approximations are used to model node movements and derive performance metrics like packet delivery probability and delay.

However, studies of real mobility scenarios [9], [10] reveal more complex structure, comprising heterogeneity and correlation in nodes' mobility. Protocol design has incorporated these findings, devising more sophisticated, albeit still locally greedy, solutions [5], [11]. In contrast, the analysis of DTN protocols still relies on overly simple mobility assumptions. A recent study [12] departs from the IID assumption by introducing mobility classes, but is still not fully flexible.

To this end, this paper proposes a unified analytical framework for greedy, distributed optimization algorithms in DTNs. We propose that algorithms for optimization problems in this context be modeled as a stochastic traversal of the solution space. This traversal can be described by a Markov chain whose transition matrix depends on two key elements: (i) the contact probability of a given node pair (*mobility component*), and (ii) an acceptance/rejection variable for a "proposed" transition (*algorithmic component*). The decoupling of mobility and algorithmic influence allows us to examine the mobility conditions under which local greedy algorithms are correct (i.e. converge). Moreover, it enables the use of transient Markov analysis to calculate convergence probability and convergence delay of an algorithm in *generic mobility settings*.

While we believe that the proposed framework can be easily adapted to a large variety of optimization problems for DTNs[1], throughout the paper, we use the content placement or relay selection problem [5], as a case study: New content is injected into the network, in which a large subset of nodes (e.g., a multicast group) may be interested over time. To make the content easily reachable by interested nodes, L replicas are pushed from its source to L "carriers", who will make it most available to everyone (e.g., minimizing the expected meeting time of an interested node and a carrier). We chose this "group" communication problem, as we believe that group communication (e.g., multicast, anycast, publish/subscribe) will be more relevant than unicast routing to content dissemination applications envisioned for these networks.

Our main contributions are summarized below:

- We propose a Markov chain model that combines the

[1]Some additional examples can be found in [13]. Note also that our framework naturally applies to stochastic optimization algorithms (e.g. Metropolis-Hastings Monte Carlo algorithms [5]), in addition to locally greedy algorithms. Due to space limitations, we choose to focus on greedy algorithms here.

heterogeneous mobility properties of a scenario and the actions of an algorithm into an appropriate transition matrix over a problem's solution space (Section II).
- We prove necessary and sufficient conditions for an algorithm's correctness and examine whether and when these conditions are met, using an ample set of mobility traces (Section III).
- Based on the absorption properties of the above Markov chain, we derive closed form results for the convergence probability and convergence delay of an algorithm in generic mobility scenarios (Section IV).

II. Model and Problem Definition

Mobility and Contact Model: Let $\mathcal{N}$ be the set of all nodes in our Opportunistic Network, $|\mathcal{N}| = N$. Each of the N nodes has a unique ID. A *contact* occurs between two nodes who are in range to setup a bi-directional wireless link to each other. We assume that contacts last for a negligible time compared to that between two successive contacts. We also assume that contacts happen in sequence, according to some contact arrival process, whose intensity is related to network density[2]. A given mobility scenario with heterogeneous node mobility can then be described through its pairwise contact probabilities p^c_{ij}, that the next contact in the sequence is between nodes i and j, and the respective ***contact probability matrix***:

$$\mathbf{P}^c = \{p^c_{ij}\}. \tag{1}$$

This implies that, at any moment, the remaining inter-contact time between nodes i and j has a geometric distribution with parameter p^c_{ij}. This probability is akin to the (relative) frequency or intensity of contacts between a given pair of nodes. It can be either measured directly from a given real or synthetic mobility trace (e.g. fitting the inter-contact time distribution for every pair); or it can be calculated using the pair's contact statistics (e.g. frequency and/or duration [14]). Finally, though the geometric distribution assumption might not always hold in practice, it is a useful approximation that we have validated against a number of real traces.

Solution Space for Content Placement Problem: Having described how nodes move and meet each other, we now describe the content placement or relay selection problem. This problem can be formally defined as follows: *A source node is given a finite budget L of replicas*[3] *for some content it creates or obtains. Find an optimal subset of nodes $\mathcal{L}^\star \subset \mathcal{N}$ who will store the content, so as to maximize some accessibility utility* U. The content may be a popular video, for example, that is expected to be heavily requested in the future.

Let us assume an initial network configuration is achieved, with the L content copies at L different nodes (the source can do this through e.g., spraying [2]). This network state can be formally expressed using an N-element vector:

$$\mathbf{x} = (x_1, x_2, \ldots, x_N), \qquad x_i = \begin{cases} 1 & i \text{ is a relay,} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

[2]For our analysis, we require that the probability of simultaneous contacts be small. This is the case, e.g., if the arrival process of contact events is Poisson. In general, the assumption just implies a relatively sparse network.

[3]The parameter L achieves a trade-off between performance and cost.

Thus, our problem is defined by the following state spaces:

$$x_i \in \mathcal{S} \qquad \mathcal{S} = \{0,1\} \qquad \textbf{node state space}^4, \tag{3}$$

$$\mathbf{x} \in \Omega \qquad |\Omega| = \binom{N}{L} \qquad \textbf{network state space}^5. \tag{4}$$

Solution Space Traversal: Every contact between a relay i ($x_i = 1$) and another node j ($x_j = 0$) offers the chance of moving to a new state or solution $\mathbf{y} \in \Omega$. In other words, *the sequence of solutions presented to a distributed optimization algorithm in this context are dictated by the contact probability matrix* $\mathbf{P}^c$. This is in contrast with traditional optimization algorithms, where a local neighborhood of solutions to be compared with the current one in the next step is defined/chosen by the designer.

If a content replica is transferred from relay i to j, then $y_i = 0$ and $y_j = 1$. Figure 1 provides an example of *potential* state transitions. Network configurations $\mathbf{x}$ and $\mathbf{y}$ only differ in the states of nodes 2 and 4; for any other node k, $y_k = x_k$. Defining the ***configuration difference*** as

$$\delta(\mathbf{x}, \mathbf{y}) = \sum_{1 \leqslant i \leqslant N} \mathbb{1}\{x_i \neq y_i\}, \tag{5}$$

we say two states are *adjacent* if and only if $\delta(\mathbf{x}, \mathbf{y}) = 2$. Then, a transition $\mathbf{xy}$ is possible.

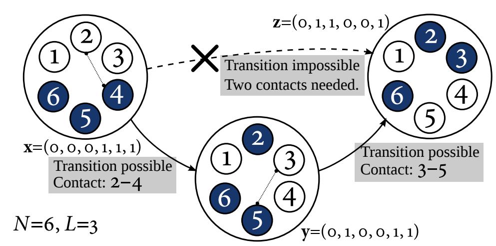


Fig. 1. Example state transitions for Content Placement. Note that $\delta(\mathbf{x}, \mathbf{y}) = \delta(\mathbf{y}, \mathbf{z}) = 2$, while $\delta(\mathbf{x}, \mathbf{z}) = 4$. Thus, transition $\mathbf{xz}$ is not possible.

Local Optimization Algorithm: Node contacts only *propose* new candidate solutions. Whether the relay i in fact hands over its content replica to j (or, in the more general case, whether a new allocation of objects between i and j is chosen), is decided by the *algorithm*. In greedy schemes, a *possible* state transition occurs only if it improves a utility function U, defined over Ω. Here, let $U_\mathbf{x}$ express the accessibility offered by configuration $\mathbf{x}$ (e.g., expected meeting delay between a node and any relay). Then, a possible transition $\mathbf{xy}$ occurs with **acceptance probability**:

$$A_{\mathbf{xy}} = \mathbb{1}\left\{U_\mathbf{x} < U_\mathbf{y}\right\}.$$

While, $A_{\mathbf{xy}} \in \{0,1\}$ for greedy (utility-ascent) algorithms, generally, the acceptance probability may be any function of the two utilities: $A_{\mathbf{xy}} \in [0,1]$. This allows us to model stochastic utility-ascent algorithms (e.g. simmulated annealing).

[4]A binary node state space works for denoting single items at each node, as in unicast and multicast routing. For different optimization problems in DTNs, a larger node state space may be needed (see [13] for an example). Nevertheless, the basic framework remains the same.

[5]The network state space may become large, depending on the problem (e.g., 2^N in epidemic routing [1]). This can be handled using state lumping, Petrinet-based models, etc. Due to space limitations, we defer the treatment of these issues to future work.

In general, the utility U to maximize is: (i) node mobility-related, e.g. node degree or contact probabilities [11], (ii) node features-related, e.g. buffer space or battery, or (iii) content-related, e.g. demand for content. Without loss of generality, for our content placement problem we will assume that a node's utility is directly proportional to the number of unique nodes it encounters per time unit, and that the utility of a given solution (i.e. assignment of available replicas to relays) is the sum of individual relay utilities. This is a reasonable, greedy metric towards maximizing each node's accessibility to the content. We refer the interested reader to [5] for a more detailed discussion of utility functions for this problem. (We note that the upcoming analysis has been validated against different utility functions in [13].)

A Markov Chain Model for Distributed Optimization: Summarizing, the transition probability between *adjacent* network states $\mathbf{x}$ and $\mathbf{y}$ can be expressed in function of the contact probability and the acceptance probability as:

$$p_{\mathbf{xy}} = p^c_{ij} \cdot A_{\mathbf{xy}}, \tag{6}$$

where nodes i and j are the two nodes whose encounter provokes the state transition.

Finally, note that the transition from any state $\mathbf{x}$ to any other state $\mathbf{y}$ only depends on these two states and not on past states. This means that our system has the Markov property, therefore we model it with a time-homogeneous discrete-time Markov chain $(\mathbf{X}_n)_{n\in\mathbb{N}_0}$ over the solution space Ω. From above, the transition probabilities of the Markov chain are:

$$\begin{aligned} p_{\mathbf{xy}} &= \mathbb{P}\left[\mathbf{X}_{n+1} = \mathbf{y} \,|\, \mathbf{X}_n = \mathbf{x}\right] \\ &= \begin{cases} 0, & \delta(\mathbf{x},\mathbf{y}) > 2 \\ p^c_{ij} \cdot A_{\mathbf{xy}}, & \delta(\mathbf{x},\mathbf{y}) = 2 \\ 1 - \sum\limits_{\substack{1 \leqslant \mathbf{z} \leqslant |\Omega| \\ \mathbf{z} \neq \mathbf{x}}} p_{\mathbf{xz}}, & \mathbf{x} = \mathbf{y}. \end{cases} \end{aligned} \tag{7}$$

where, i and j are the two nodes whose encounter provokes the state transition. p^c_{ij} is the ***mobility component*** of the transition probability and $A_{\mathbf{xy}}$ is the ***algorithm component***.

To apply this model to other DTN problems solved by utility-ascent algorithms, one needs at most to redefine the state space Ω and the utility function U appropriately. Some problems can even be directly mapped to the above content placement model: e.g., for single-copy routing simply set $L = 1$ and assign the highest utility to the destination. Due to space limitations, we defer the demonstration of our model's flexibility to future work.

III. Correctness Analysis and Usage

A crucial point that arises with the use of greedy distributed optimization in DTNs is the (in)ability of such algorithms to efficiently navigate the solution space. Specifically, *what properties of the mobility model or the utility function make simple utility ascent algorithms applicable?* In this section, we illustrate the value of the presented Markov model in answering this questions, for the case of Greedy Content Placement. To our best knowledge, this is the first *correctness* analysis for DTN algorithms for generic mobility scenarios.

A. Correctness Analysis for Greedy Content Placement

Our goal is to prove necessary and sufficient conditions for the correctness of gradient-ascent algorithms for the content placement problem, that is guaranteed discovery of the optimal solution $\mathbf{x}^\star$. This solution amounts to "pushing" a replica to each of the L highest utility nodes. We denote this set of nodes as $\mathcal{L}^\star$. By definition, the network state $\mathbf{x}^\star$ is *absorbing* in the Markov chain of our problem. Depending on the mobility scenario and the chosen utility function, there may be more absorbing states (i.e., local maxima of U). In that case, the algorithm is not guaranteed to converge to the optimal solution from every initial copy assignment, and thus is not correct.

Thm. 1 derives necessary and sufficient conditions on the *contact probability matrix* $\mathbf{P}^c$ and *utility function* U, for the correctness of the greedy algorithm. This is shown to require the existence of an increasing utility path from any node in $\mathcal{N} \setminus \mathcal{L}^\star$ (nodes outside the L highest utility ones) and any node in $\mathcal{L}^\star$. Due to space limitations, the proof is found in [13].

Theorem 1: For all source nodes and all initial copy allocations, *Greedy Content Placement* is correct if and only if for all $i \in \mathcal{N} \setminus \mathcal{L}^\star$ there exist at least L nodes $j_1, \ldots, j_L \in \mathcal{N}$ with $U_i < U_{j_{(.)}}$, such that $p^c_{ij_{(.)}} > 0$.

Put differently, given a DTN distributed optimization problem (mapped into a utility function), and a mobility scenario (captured in a contact probability matrix), Thm. 1 converts the (often hard) task of deciding whether a simple greedy algorithm would suffice, into the (often easier) task of checking a matrix (the contact matrix) for enough non-zero entries.

B. Correctness Conditions in Realistic Mobility Scenarios

Having derived conditions for the correctness of Greedy Content Placement, we investigate here whether and when these conditions actually hold in realistic mobility scenarios. To cover a broad range, we use five real contact traces and one synthetic mobility trace for validation. Their characteristics are summarized in Table I: (i) the *Reality Mining* trace (MIT) [9], (ii) the Infocom 2005 trace (INFO) [15], (iii) the ETH trace [10], (iv) a Swiss military trace from an outdoor training scenario (ARMA), (v) the San Francisco taxis trace (SFTAXI) and (vi) a synthetic scenario created with a recent mobility model (TVCM) [16].

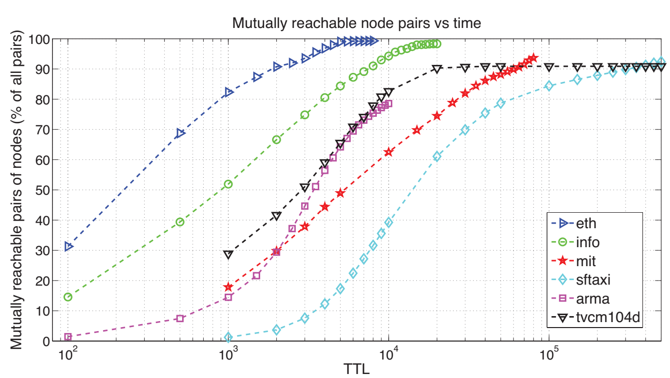
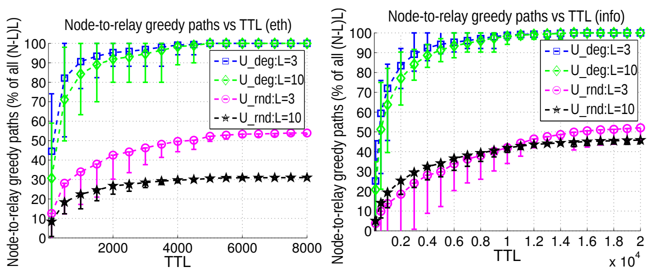


Fig. 2. Multihop greedy paths in traces

Figure 2 shows the percentage of node pairs that are mutually reachable by a utility ascent path as a function of TTL. While for the whole trace duration, over 90% of the node

	MIT	INFO	ETH	ARMA	SFTAXI	TVCM
Scale and context	92 campus students & staff	41 conference attendees	20 lab students & staff	152 people	536 taxis	24/104 nodes, 2/4 disjoint communities
Period	9 months	3 days	5 days	9 hours	1 month	11 days
Scanning Interval	300s (Bluetooth)	120s (Bluetooth)	0.5s (Ad Hoc WiFi)	30s (GPS)	30s (GPS)	N/A
# Contacts total	81 961	22 459	22 968	12 875	1 339 274	1 000 000

TABLE I
MOBILITY TRACES CHARACTERISTICS.



(a) Utility Ascent paths (ETH) (b) Utility Ascent paths (Infocom)

Fig. 3. Comparison of several utilities

pairs are greedily reachable (note that the whole trace duration exceeds one week worth of contacts for some of the traces), for smaller, more realistic TTL values, many node pairs do not have paths for the greedy algorithm to use. This is due either the *nodes being unreachable* or to existence of *absorbing local maxima*. This can be determined by comparison to the number of mutually reachable pairs when allowing non-greedy paths. In ETH, the number of paths does not increase significantly, making this scenario suitable for greedy solutions. In contrast, in the other scenarios, the improvement ranges from double to fivefold, making greedy solutions waste potential.

In addition to the contact pattern of a mobility scenario, the conditions of Thm. 1 (and thus algorithm correctness) may be affected by the following factors: (a) the value of L, (b) the choice of utility function. In Figure 3, we inspect more closely the effect of L and the relation between a mobility-related utility and a random utility. It provides a comparison of two relevant utilities: (i) the mobility-related node degree, and (ii) a random utility, which has no correlation to mobility.

Two observations ensue from Figure 3. First and foremost, there do not always exist utility ascent paths leading to optimal solutions. This means local maxima are present even for simple utility functions. Larger L may exacerbate this problem (Figure 3(a)). Second, the correlation or lack thereof between a node's utility rank and its mobility rank considerably affects the navigability properties of the contact graph. This stresses the need to choose utility functions carefully.

Summarizing, we have used the correctness conditions derived in Section III-A, to analyze the worst case behavior of the *Greedy Content Placement* algorithm in a variety of real and synthetic mobility scenarios. We conclude that this algorithm is relatively fragile in the face of parameters like TTL and the choice of utility function. As mentioned earlier, our model can adapt to study other algorithms and optimization problems in a similar way. When an algorithm's worst case behavior is important, this is an essential tool for its analysis.

IV. CONVERGENCE ANALYSIS AND USAGE

In the previous section, we proved necessary and sufficient conditions for the greedy content placement algorithm to reach the globally optimal configuration (Markov chain state) from *any* initial network state. In addition to *worst case performance*, in practice we are also interested in the following performance metrics for an algorithm:

i) *The convergence probability* of the algorithm to the global optimum (less than 1 in the presence of local maxima).
ii) *The convergence time* of algorithms. This translates to quantities of practical importance, e.g. the delivery delay.

Existing analytical models for DTNs also treat these quantities, but they assume unrealistic mobility and are not easily applicable to complex algorithms where decisions are utility-based [5], [11]. Here, we use heterogeneous mobility (pairwise contact probabilities) in our Markov chain model and show that convergence probability and mean convergence time of each algorithm can be mapped to statistics of *absorption* quantities on the Markov chain. These statistics can often be derived in closed form after some matrix algebra.

A. Convergence Analysis

We study the performance of our *Greedy Content Placement* algorithm. The Markov Chain $(\mathbf{X}_n)_{n\in\mathbb{N}_0}$ described by $\mathbf{P} = \{p_{\mathbf{xy}}\}$ (Eq. (7)), corresponding to the solution space traversal for the *Greedy Content Placement* algorithm is *absorbing*. We use the theory of absorbing Markov chains to characterize the algorithm's performance [17].

The maximum utility network state $\mathbf{x}^\star$ is, by definition, an absorbing state in the Markov chain. Additional local maxima correspond to other absorbing states in $\mathbf{P}$. We denote the set of such states

$$\mathcal{LM} \subset \Omega \quad \text{(local maxima).}$$

$\mathcal{LM}$ contains all solutions $\mathbf{x} \in \Omega \setminus \{\mathbf{x}^\star\}$, such that for all states $\mathbf{y} \in \Omega$ with $\delta(\mathbf{x},\mathbf{y}) = 2$, either $p^c_{ij} = 0$ or $U_\mathbf{y} < U_\mathbf{x}$ (as in Thm. 1). Every other solution in Ω is a transient state. Denote by TR $\subset \Omega$, the set of transient states. Then, $\Omega = \{\mathbf{x}^\star\} \cup \mathcal{LM} \cup \text{TR}$.

In order to derive absorption related quantities, we write the matrix $\mathbf{P}$ in *canonical form*, where states are re-arranged such that transient states (TR) come first, followed by absorbing states corresponding to local maxima ($\mathcal{LM}$), followed by the maximum utility state $\mathbf{x}^\star$:

$$\mathbf{P} = \begin{array}{c} \begin{array}{ccc} \text{TR} & \mathcal{LM} & \mathbf{x}^\star \end{array} \\ \left(\begin{array}{c:c:c} \mathbf{Q} & \mathbf{R_1} & \mathbf{R_2} \\ \hdashline \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \hdashline \mathbf{0} & \mathbf{0} & 1 \end{array}\right) \end{array} \begin{array}{c} \text{TR} \\ \mathcal{LM} \\ \mathbf{x}^\star \end{array}$$

Let $|\mathcal{LM}| = r_1$ and $|\text{TR}| = t$. That is, there are r_1 local maxima and t transient states. Then, $\mathbf{I}$ is the $r_1 \times r_1$ identity matrix, $\mathbf{Q}$ is a $t \times t$ matrix, $\mathbf{R_1}$ is a non-zero $t \times r_1$ matrix, $\mathbf{R_2}$ is a non-zero t-element column vector.

We can now define the *fundamental matrix* $\mathbf{N}$ for the absorbing Markov chain as follows:

$$\mathbf{N} = (\mathbf{I} - \mathbf{Q})^{-1} = \mathbf{I} + \mathbf{Q} + \mathbf{Q}^2 + \cdots \tag{8}$$

The last equality is easy to derive (see [17], page 45). $\mathbf{N}$ is a $t \times t$ matrix whose entry $n_{\mathbf{xy}}$ is the expected number of times the chain is in state $\mathbf{x}$, starting from state $\mathbf{y}$, before getting absorbed. Thus, the sum of a line of the fundamental matrix of an absorbing Markov chain is the expected number of steps until absorption, when starting from the respective state.

Theorem 2 (Success Probability): The probability that *Greedy Content Placement* succeed in finding the optimal solution, starting from any initial state with equal probability, is

$$p_g = \frac{1}{t} \cdot \sum_{\mathbf{x} \in TR} b_{\mathbf{xx}^\star}, \tag{9}$$

$\mathbf{B}^\star = \{b_{\mathbf{xx}^\star}\}$ is a t-element column vector with $\mathbf{B}^\star = \mathbf{NR_2}$.

Proof: Starting from transient state $\mathbf{x}$, the process may be captured in the optimal state, $\mathbf{x}^\star$, in one or more steps. The probability of capture on a single step is $p_{\mathbf{xx}^\star}$. If this does not happen, the process may move either to an absorbing state in $\mathcal{LM}$ (in which case it is impossible to reach $\mathbf{x}^\star$), or to a transient state $\mathbf{y}$. In the latter case, there is probability $b_{\mathbf{yx}^\star}$ of being captured in the optimal state. Hence we have:

$$b_{\mathbf{xx}^\star} = p_{\mathbf{xx}^\star} + \sum_{\mathbf{y} \in TR} p_{\mathbf{xy}} \cdot b_{\mathbf{yx}^\star}, \tag{10}$$

which can be written in matrix form as $\mathbf{B}^\star = \mathbf{R_2} + \mathbf{QB}^\star$. Thus $\mathbf{B}^\star = (\mathbf{I} - \mathbf{Q})^{-1}\mathbf{R_2} = \mathbf{NR_2}$. $\mathbf{B}^\star$ is the vector of success probabilities starting from each of the t transient states. We obtain the probability of success starting from any state uniformly, as follows:

$$\frac{1}{t} \cdot b_{\mathbf{x}_1\mathbf{x}^\star} + \cdots + \frac{1}{t} \cdot b_{\mathbf{x}_t\mathbf{x}^\star} = \frac{1}{t} \cdot \sum_{\mathbf{x} \in TR} b_{\mathbf{xx}^\star}. \tag{11}$$

■

Theorem 3 (Convergence Delay): The expected time for *Greedy Content Placement* to find the optimal solution, starting from any initial state with equal probability, given that it does not get absorbed in any local maximum is:

$$\mathbb{E}[T_g] = \frac{1}{t} \cdot \sum_{\mathbf{x} \in TR} \tau_{\mathbf{x}}, \tag{12}$$

where $\boldsymbol{\tau} = \{\tau_{\mathbf{x}}\}$ is a t-element column vector with $\boldsymbol{\tau} = \mathbf{D}^{-1}\mathbf{NDc}$. $\mathbf{c}$ is a t-element column vector with ones, and $\mathbf{D}$ is a diagonal matrix with entries $b_{\mathbf{xx}^\star}$ for $\mathbf{x} \in TR$.

Proof: Assume we start in a non-absorbing state $\mathbf{x}$ of our Markov chain $(\mathbf{X}_n)_{n\in\mathbb{N}_0}$ and compute all probabilities relative to the hypothesis that the process ends up in the optimal state, $\mathbf{x}^\star$. Then we obtain a new absorbing chain $(\mathbf{Y}_n)_{n\in\mathbb{N}_0}$ with a single absorbing state $\mathbf{x}^\star$. The non-absorbing states will be as before, except we have new transition probabilities. We compute these as follows. Let $\mathbf{a}$ be the statement "$(\mathbf{X}_n)_{n\in\mathbb{N}_0}$ is absorbed in state $\mathbf{x}^\star$". Then if $\mathbf{x}$ is a non-absorbing state, the transition probabilities for $(\mathbf{Y}_n)_{n\in\mathbb{N}_0}$ are:

$$\begin{aligned}
\mathbb{P}[\mathbf{Y}_{n+1} = \mathbf{y} \,|\, \mathbf{Y}_n = \mathbf{x}] &= \mathbb{P}[\mathbf{X}_{n+1} = \mathbf{y} \,|\, \mathbf{a} \wedge \mathbf{X}_n = \mathbf{x}] \\
&= \frac{\mathbb{P}[\mathbf{X}_{n+1} = \mathbf{y} \wedge \mathbf{a} \,|\, \mathbf{X}_n = \mathbf{x}]}{\mathbb{P}[\mathbf{a} \,|\, \mathbf{X}_n = \mathbf{x}]} \\
&= \frac{\mathbb{P}[\mathbf{a} \,|\, \mathbf{X}_{n+1} = \mathbf{y}] \cdot \mathbb{P}[\mathbf{X}_{n+1} = \mathbf{y} \,|\, \mathbf{X}_n = \mathbf{x}]}{\mathbb{P}[\mathbf{a} \,|\, \mathbf{X}_n = \mathbf{x}]} \\
\hat{p}_{\mathbf{xy}} &= \frac{b_{\mathbf{yx}^\star} p_{\mathbf{xy}}}{b_{\mathbf{xx}^\star}}.
\end{aligned}$$

The standard form for $\hat{\mathbf{P}}$, the transition matrix of $(\mathbf{Y}_n)_{n\in\mathbb{N}_0}$, may be obtained as follows. The matrix $\hat{\mathbf{R}}$ is a column vector with $\hat{\mathbf{R}} = \{\frac{p_{\mathbf{xx}^\star}}{b_{\mathbf{xx}^\star}}\}$. Let $\mathbf{D}$ be a diagonal matrix with diagonal entries $b_{\mathbf{xx}^\star}$, for $\mathbf{x}$ non-absorbing. Then $\hat{\mathbf{Q}} = \mathbf{D}^{-1}\mathbf{QD}$ and consequently, $\hat{\mathbf{N}} = \mathbf{D}^{-1}\mathbf{ND}$. $\hat{\mathbf{B}}^\star$ is now a t-element column vector of ones.

For the derivation of $\boldsymbol{\tau}$ as a function of $\hat{\mathbf{N}}$ see [17], page 51. The initial state is chosen uniformly in TR. Hence, using $\boldsymbol{\tau}$ and the law of total expectation, we obtain equation 12. ■
The variance of the convergence delay is derived in the same manner. The interested reader is referred to [13].

Corollary 4: The expected time for the *Greedy Content Placement* algorithm to converge to any solution, locally or globally optimal, is given by $\frac{1}{t} \cdot \sum_{\mathbf{x} \in TR} T_{\mathbf{x}}$, where $\mathbf{T} = \{T_{\mathbf{x}}\}$ is $\mathbf{T} = \mathbf{Nc}$. $\mathbf{c}$ is a t-element column vector with ones.

B. Validation of Convergence Results

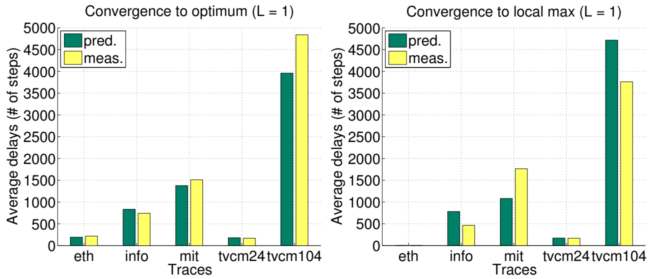
We compare the analytical results from the above theorems to simulation results, using the same traces as previously. We show results for $L = 1$ and $L > 1$. In $L > 1$ results, for every mobility scenario, L has the largest value we could handle in our Matlab code, without any state space reduction techniques (usually $L < 5$). For $L > 5$ the (multi-dimensional) Markov chain makes it hard to calculate the inverses needed in Thm. 2 and 3. However, since this state space is approximately the product of L identical Markov chains, we expect that this complexity could be efficiently tackled, using approximations, state lumping, etc. We defer this to future work.

	L	Optimum pred.	Optimum meas.	Local max. pred.	Local max. meas.
ETH	1	1.0000	1.0000	N/A	N/A
Infocom	1	0.5267	0.5486	0.4132	0.3893
MIT	1	0.7586	0.7625	0.2044	0.2005
TVCM24	1	0.6087	0.5972	0.3913	0.4028
TVCM104	1	0.7862	0.6971	0.2138	0.3029
ETH	4	1.0000	1.0000	N/A	N/A
Infocom	3	0.3299	0.3001	0.1601	0.1775
MIT	2	0.5612	0.5658	0.2964	0.2889
TVCM24	4	0.0987	0.0967	0.3770	0.3623
TVCM104	2	0.6040	0.4686	0.3609	0.4582

TABLE II
ABSORPTION PROBABILITIES

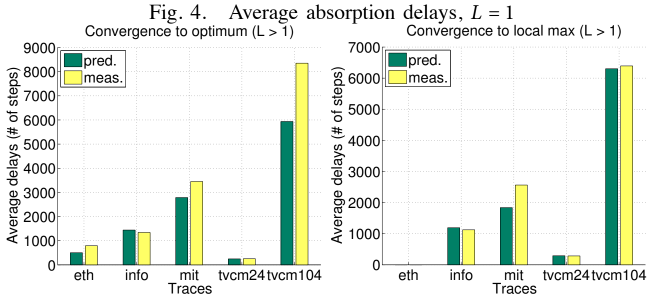
Table II shows absorption probabilities predicted using Thm. 2 and measured in the traces. The first two columns give the probability of absorption by the global optimum (theory and simulation), the second two – the probability of absorption by a local maximum. In the majority of cases, the prediction is sufficiently accurate, both with a single absorbing state, the global maximum (in ETH) and when local maxima are present, resulting in multiple absorbing states (in Infocom, MIT, TVCM24, TVCM104).

Fig. 4 and Table 5 compare the predicted and measured values of the absorption delay averaged over all initial states for the greedy algorithm, with $L = 1$ and respectively $L > 1$. These results correspond to Thm. 3. (In Fig. 4(b) some traces, e.g. ETH, do not have any local maxima with our utility.) The theoretical results coincide once again well with the measured delays, both for absorption by the optimum state and for absorption by a local maximum.



(a) Absorption by optimum (b) Absorption by local max.

Fig. 4. Average absorption delays, $L = 1$



(a) Absorption by optimum (b) Absorption by local max.

Fig. 5. Average absorption delays, $L > 1$

C. Discussion and Implications

Summarizing, despite simplifying assumptions required for tractability (e.g. geometrically distributed time to next contact, contact independence), the proposed framework is able to predict relevant performance metrics for Content Placement fairly accurately under a large variety of real and realistic mobility scenarios. To our best knowledge, this is the first analytical work that can accurately predict performance metrics for general, heterogeneous mobility. Moreover, while our preliminary results have focused on the Content Placement problem, the main components of our framework are generic and should enable accurate performance predictions for any problem that fits this framework. As an immediate example, utility-based single-copy routing schemes can be directly mapped to Content Placement with $L = 1$ and an appropriate utility ranking (with the destination as the optimum). In future work, we intend to conduct a similar performance analysis for this and other problems (buffer management, anycast/multicast, etc.) to further validate the merit of this framework.

Beyond the theoretical aspects of our analysis, we believe the presented model can be useful to protocol or system designers. New protocols must be evaluated prior to implementation, and an idea of their sensitivity to various parameters and mobility scenarios is necessary. While trace-driven or synthetic simulations can often provide good estimates, they usually require large amounts of computing resources and time, in order to obtain good confidence intervals for all possible parameter values and scenarios. In contrast, given a trace of the target environment, our model allows for a quick evaluation of the algorithm with reasonable accuracy. It can also be used to achieve trade-offs (e.g., delay vs. number of copies) and to tune parameters (e.g., of the utility function [11]). Finally, our model also offers important insight into the structure of the problem and the interplay among its various parts (mobility, algorithm). Thus, a system designer could easily assess, for example: whether a simple greedy algorithm is sufficient for the targeted environment or a more sophisticated solution is needed; whether a certain utility function is appropriate for the mobility scenario at hand (both shown in Section III).

V. Conclusion

In this paper, we presented a generic model and analytical framework for DTN algorithms. Unlike earlier analytical research work, our model captures the full heterogeneity of node mobility, which has been observed in real scenarios [9], [10]. Moreover, our framework allows the examination of a larger class of algorithms, instances of which are very frequently proposed as solutions to DTN problems: utility-ascent/descent algorithms, be they probabilistic of deterministic.

We illustrated the use of our model and framework, by applying it to a deterministic utility-ascent algorithm for the Content Placement problem in DTNs. Specifically, we proved necessary and sufficient conditions for its correctness, and derived closed form solutions for its performance (convergence probability and delay), in various mobility scenarios. We used real and synthetic mobility traces to verify our findings, and found a close match between predictions and measurements.

In the future, we plan to demonstrate and evaluate the applicability of our model to the other DTN problems, such as routing, resource allocation etc. Moreover, we will address the state space explosion problem, which will extend the usability of the model.

References

[1] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Tech. Rep., 2000.

[2] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Spray and Wait: an efficient routing scheme for intermittently connected mobile networks," in *WDTN 2005*.

[3] A. Balasubramanian, B. Levine, and A. Venkataramani, "DTN routing as a resource allocation problem," *SIGCOMM*, 2007.

[4] A. Krifa, C. Barakat, and T. Spyropoulos, "Optimal buffer management policies for delay tolerant networks," in *IEEE SECON*, 2008.

[5] A. Picu and T. Spyropoulos, "Distributed stoch. optimization in Opportunistic Nets: The case of optimal relay selection," in *CHANTS 2010*.

[6] A. A. Hanbali, A. Kherani, and P. Nain, "Simple models for the performance evaluation of a class of two-hop relay protocols," in *IFIP Networking 2007*.

[7] R. Groenevelt, P. Nain, and G. Koole, "The message delay in mobile ad hoc networks," *Perform. Eval.*, vol. 62, 2005.

[8] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Performance analysis of mobility-assisted routing," in *MobiHoc 2006*.

[9] N. Eagle and A. Pentland, "Reality mining: sensing complex social systems," *Pers. Ubiq. Comput.*, vol. 10, no. 4, pp. 255–268, 2006.

[10] V. Lenders, J. Wagner, and M. May, "Measurements from an 802.11b mobile ad hoc network," in *WOWMOM 2006*.

[11] E. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant MANETs," in *MobiHoc 2007*.

[12] T. Spyropoulos, T. Turletti, and K. Obraczka, "Routing in delay-tolerant networks comprising heterogeneous node populations," *IEEE Trans. Mob. Comput.*, vol. 8, 2009.

[13] A. Picu and T. Spyropoulos, "Distributed optimization in DTNs: Towards understanding greedy and stochastic algorithms," ETH Zürich, Tech. Rep. TIK Report Nr. 326, 2010.

[14] T. Hossmann, T. Spyropoulos, and F. Legendre, "Social network analysis of human mobility and implications for dtn performance analysis and mobility modeling," ETH Zürich, Tech. Rep. TIK Report Nr. 323, 2010.

[15] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot, "Pocket switched networks and human mobility in conference environments," in *WDTN 2005*.

[16] W.-J. Hsu, T. Spyropoulos, K. Psounis, and A. Helmy, "Modeling spatial and temporal dependencies of user mobility in wireless mobile networks," *IEEE/ACM Trans. Netw.*, 2009.

[17] J. Kemeny and L. Snell, *Finite Markov Chains*, 1960.