Institut Eurécom
Networking and Security Dept.
2229, route des Crêtes
B.P. 193
06904 Sophia-Antipolis
FRANCE

Research Report RR-06-172

Performance Analysis of Cooperative Content Distribution for Wireless Ad hoc Networks

Pietro Michiardi, Guillaume Urvoy-Keller

Tel : (+33) 4 93 00 81 45
Email : {Pietro.Michiardi, Guillaume.Urvoy-Keller}@eurecom.fr

[1]Institut Eurécom's research is partially supported by its industrial members: Swisscom, Thales, SFR, France Téécom, Hitachi, STMicroelectronics, Bouygues Téécom, Sharp, Cisco Systems, BMW Group Research & Technology BMW Group Company.

Abstract

In this paper we focus on the problem of content distribution in wireless ad hoc networks. Our goal is to come up with a fully decentralized mechanism to distribute content from one source to a potentially large number of destinations. Despite the large literature on content distribution schemes available for wired settings we argue that the very nature of the underlying ad hoc network poses new challenges that cannot be addressed with current schemes. We propose a cooperative peer-to-peer scheme that allows parallel download of the content based on swarming protocols. Our scheme builds a distribution overlay network that takes into account traffic locality and allows peers to trade parts of the content while sustaining cooperation. We evaluate through simulations the performance of our scheme for different static scenarios using a variety of metrics that allows characterizing the impact of our solution at different layers of the system stack. Our results highlight the great benefits of our solution in terms of system fairness, achievable throughput and energetic consumption. We also study the scalability properties of our solution under the extended network model and discuss on per-peer capacity when the size of the network exhibit a realistic growth.

1 Introduction

In this paper we focus on the problem of efficient content distribution for (mobile) ad hoc networks (MANETs). The founding principle that characterizes a MANET is the cooperative nature of the network. The lack of pre-deployed components that control the network operation is alleviated by node cooperation whereby every node has to bear the costs of network functioning. In a MANET, nodes act both as data terminals and routers, so that every node in the network is responsible for forwarding packets on behalf of other nodes. The lack of a routing infrastructure is alleviated by capitalizing on nodes' willingness to share their capacity for the benefit of the network[1].

In the decentralized and dynamic setting offered by MANETs, traditional mechanisms for efficient content distribution designed for the Internet cannot be directly used. In general, structured approaches whereby a static distribution topology is built to transport data from the source(s) of the content to all potential destinations suffers from high node churn rates [8]. Approaches such as Akamai [1] that rely on dedicated servers/proxies do not match the requirements of the ad hoc paradigm because a pre-deployed infrastructure cannot be assumed and because the cost of content distribution should be evenly shared among the nodes.

With the aim of improving the performance of content distribution by removing the bottleneck associated with a single source, researchers have recently revealed the benefits of cooperative content distribution (CCD) schemes. Mechanisms such as BitTorrent [3] and Slurpie [21] are based on peer cooperation whereby un-used upload capacity available at every recipient is shared for the benefit of other peers, reducing the capacity needs at the source. CCD schemes assume the content to be split in pieces in order to allow parallel download. They are based on two key algorithms: one algorithm which we call the *peer selection algorithm* drives the selection of peers with whom to cooperate; the second algorithm, namely the *piece selection algorithm*, is used to select the pieces of the content that will be exchanged.

Alternative CCD schemes have been recently proposed to minimize the costs (in terms of message exchange) associated to the piece scheduling strategy that affect BitTorrent-like systems [12]. These techniques rely on the novel concept of *network coding* whereby every peer of the system is able to inject new pieces of the content obtained through random linear combinations of the pieces (or combinations of the pieces) already in its possession. The randomization introduced by the coding process eases the scheduling of piece propagation, and, thus, makes the distribution process more efficient. This is particularly important in large unstructured overlay networks, where the nodes need to make piece forwarding decisions based on local information only. However, the computational costs associated to network

[1] Note that, in general, node cooperation cannot be assumed a priori. To solve the problems introduced by node selfishness several approaches have been studied in the literature. For sake of simplicity, in this paper we take node cooperation as granted. We will address the issues raised by selfish nodes in our future work.

coding might be prohibitive in the context of ad hoc networks, where mobile devices are generally assumed to have limited capabilities in terms of computational resources.

A common feature of CCD schemes designed for the Internet is that peers are connected to each other through a random distribution overlay built with the help of a centralized component.

The case of MANET introduces new problems that are not present in the Internet. Analytical studies [14] show that the per-node capacity of a static MANET does not scale well with the number of nodes in the network. Even if this pessimistic result has been relaxed for the mobile case in [13], the number of hops traversed by a packet from its source to the destination can have a dramatic impact on the achievable throughput [18]. Furthermore, in a pure ad hoc environment, no centralized component can be assumed to be deployed for bootstrapping the content distribution overlay. Peers must self-organize and form the dissemination structure through a distributed algorithm.

In this paper we propose a swarming mechanism that mitigates the effects of the aforementioned issues. Our solution is inspired by the BitTorrent protocol [2] and relies on an original mechanism to build an overlay network to distribute the content from one source to a potentially large set of destinations. The main goal of the paper is to provide a thorough simulation-based analysis of the achievable performance when using cooperative content distribution schemes in a static MANET scenario. It is outside the scope of this work to present a solution for content localization and to discuss on security issues (mainly related to content integrity) that affect cooperative content distribution schemes.

The remainder of the paper is organized as follows: in Section 2 we illustrate the key ideas behind CCD schemes and discuss on the impact of a shared medium and the decentralized nature of ad hoc networks. In Section 3 we present our CCD mechanism, restricting our attention to a *static* ad hoc network. In Section 4 we describe the simulation environment used to assess the performance of our scheme and discuss on the results for a static network. We conclude with Section 6 where we also sketch our future research directions.

2 Cooperative Content Distribution

In this Section we present the terminology used in this paper and we introduce the CCD problem. We focus on the transfer of a single commodity (i.e., the content) from one source, namely the *seed*, to a potentially large number of destinations, namely the *leechers*. We assume the content to be confined to the network, that is, seeds and leechers are nodes of the ad hoc network. We assume the content to be in the form of unit-sized tokens (or pieces). A peer has two states: the *leecher state*, when it is downloading a content but does not yet have all tokens, and the *seed state*, when the peer has all the tokens. Tokens start out at the seed and the goal is to transfer them to a set of leechers that can store, forward and duplicate tokens at will.

The CCD problem consists in finding an optimal distribution schedule whereby all tokens are transferred to all receivers in the shortest time. A recent work by Killan *et. al.* [16], propose a rigorous analysis of the CCD problem setting aside the engineering challenges of protocol design to focus on the fundamental graph problem in a wireline context. In [17] the authors formulate the CCD problem assuming global knowledge and show that variants that attempt to optimize for either speed or bandwidth utilization are NP-complete.

Practical cooperative schemes have emerged as efficient solutions to cope with *flashcrowd scenarios* whereby peers exhibit a high demand rate for the commodity distributed by the initial seed. For example, in mechanisms such as BitTorrent [3] and Slurpie [21] pieces of the content are replicated as quickly as possible in order to create new sources that help the original seed in distributing the content.

In this paper we address the CCD problem drawing practical heuristics from the BitTorrent protocol [2]. Practical CCD schemes are driven by two key algorithms, the *peer selection algorithm* and the *piece selection algorithm*. Even if tightly coupled, piece and peer selection strategies have distinct objectives. The goal of a piece selection strategy is to maximize the entropy of the information distributed in the network, *i.e.* maximize the diversity of the tokens replicated by the peers [7]. An optimal replication strategy avoids bottlenecks at peers holding rare tokens, ideally sustaining a constant demand rate for each source of tokens in the overlay. The peer selection goal is to determine which peers are eligible for receiving tokens. Though various criteriae can be devised, peer selection is mainly driven by the requirement to sustain peer cooperation and achieve optimal performance.

The scope of the peer selection algorithm is restricted by the number of connections a peer maintains in the system. We term the graph resulting from the connections between each peer in the system the *distribution overlay*. Current schemes designate a centralized component to build the distribution overlay using a randomized approach. The resulting distribution overlay take the form of a random k-regular graph, in which an edge (the connection) exists between two vertices (the peers) with a probability that depends on the number of peers in the system and on the parameter k, which represents the number of connections per peer.

The underlying network affects the behavior of CCD mechanisms. As opposed to the Internet, where relatively static conditions can be met[2], a wireless and mobile scenario impose new constraints that are not present in the wireline case. We highlight the impact of a MANET environment on the two key algorithms that characterize a CCD mechanism and on the technique used to build the distribution overlay in the following sections.

[2]Nonetheless, the peculiarities of Internet graph structures, varying bandwidth and latency characteristics, failures and TCP congestion control behavior promoted a significant number of distinct system architectures, see [17] for an overview.

2.1 Issues due to a shared medium

The shift to a distributed and dynamic scenario introduces practical and conceptual issues that impact both the design of CCD mechanisms and their implementation in a MANET environment. Integrating two constructs (the distribution overlay and the physical network) with different scaling properties is not straightforward. In the wireline case, the typical assumption is that the bottleneck for the capacity of peers to serve a content is due to a limited access link capacity, whereas the capacity of the core network is assumed to be unconstrained. Under this assumptions, analytical studies [23] show that the service capacity of CCD schemes exhibits an exponential growth with the demand in a transient regime, *i.e.,* when the system bootstraps. In a stationary regime, the system performance degrades with early departures of peers that completed the download and don't contribute to the distribution process.

These results might not hold when the underlying network is a MANET. Indeed, it has been showed by the seminal work of Gupta *et. al.* [14], that the per-node capacity in a *static* n-node random ad hoc network scales as $\Theta(1/\sqrt{n \log n})$ under the assumption of random but dense traffic patterns. Other studies [11, 18] show the inverse relation that exists between the achievable throughput and the number of hops from a source to a destination of a data flow. The amount of spatial reuse, which translates into traffic locality, determines to a large extent the capacity scaling of ad hoc networks [18].

Moreover, the key idea of capitalizing on un-used peer capacity for the benefit of the distribution scheme with the ultimate goal of overcoming performance bottlenecks becomes questionable in a MANET context. The multi-hop nature of a MANET has the consequence that the upload capacity available to a peer is shared with other peers both for delivering application data and for relaying their traffic. In contrast, in the Internet the routing infrastructure bears the cost of traffic relaying while end systems can effectively trade their upload capacities for useful data. From the perspective of the content distribution problem, the case of MANET introduces the step of *content relaying* which can be considered suboptimal with respect to the step of *content replication*. Hence, traffic locality not only accounts for increased performance at the network level but also at the application level, as content replication becomes predominant on content relaying.

2.2 Issues due to a distributed environment

Besides theoretical considerations drew from analytical models, the decentralized nature of a MANET combined with a lossy environment that characterizes wireless networks, introduces practical constraints that affect the design of CCD schemes. Current CCD schemes designed for the Internet assume the presence of a central entity used to build the distribution overlay. This is in contrast with the nature of ad hoc networks whereby no centralized components can be assumed. In Section 3, we define a decentralized approach that builds the distribution overlay taking into account the constraints imposed by traffic locality. As opposed to [20], we do not modify the piece selection strategy by imposing an additional metric representing node vicinity since we believe that this alternative strategy would have unpredictable effects on the entropy of the piece replication process.

CCD schemes owe their performance to an effective use of the upload capacity available at each peer, which is sustained by the high entropy that characterizes the piece replication process. Piece replication decisions are based on a peer's local vision of the piece distribution within her neighborhood. However, this information can be affected by the unreliability that characterizes wireless networks and the loss of protocol messages used to keep fresh a peer's local vision of its neighborhood could deform the results of the piece selection strategy. We address this issue using a reliable transport protocol for sensitive information exchanged between peers, as explained in Section 3.

3 The protocol

In this section we describe the network model and define the building blocks of our scheme.

We assume the underlying ad hoc network to be composed by a set of N nodes which we assume homogeneous in terms of computation, energetic and transmission capabilities. We assume bidirectional wireless links between each node and a 802.11 MAC layer protocol supporting RTS/CTS. A unicast routing protocol is available to support packet transmission between nodes. In this work we use the OLSR [4] routing protocol; it is outside the scope of this paper studying the effects of different routing protocols on the performance of our scheme. Peers use TCP for reliable transfer of data and control messages while UDP is used by the mechanism responsible for building the overlay topology. Based on this network model we define an overlay construction mechanism which we describe in Section 3.1.

While in our original protocol we take into account the effects of mobility, in this paper we only focus on static scenarios.

3.1 Overlay construction

The distribution overlay defines the interconnections between peers, which self-organize in groups that we call *neighborhoods*. Neighborhoods define the visibility perimeter of each peer: the exchange of pieces and the knowledge of piece distribution is restricted to neighborhoods. In our scheme, the capacity constraints that affect the underlying network determine to a large extent the distribution overlay topology, which loosely matches the network topology. In the following, we propose a deterministic mechanism that limits the perimeter of a neighborhood. We define a neighbor set (NS) by the number of outgoing connections (OC) and the number of incoming connections (IC) a peer can establish. OC are locally initiated by a peer when bootstrapping the distribution overlay, while IC are remote connections corresponding to requests to join a neighborhood. The cardinality of a NS and its composition, i.e. the ratio of OC and IC connections, play an important role in determining the graph properties of the corresponding distribution overlay [22].

Our overlay construction mechanism is based on the expanding ring search technique. Every peer (including the seed) that joins the network unicasts query messages to build a NS, using the UDP transport protocol. Query messages include a sequence number, used to discard duplicate query messages, and a TTL field. The TTL is initially set to one. If the number of peers reachable within one hop from the requesting peer is less than the parameter OC, the TTL value is increased by one and the querying process starts over until OC connections can eventually be established. Note however that the TTL value cannot be increased arbitrarily: we use a threshold value (MAX_TTL) that defines the horizon of the search process. As a result, it is possible that the targeted value of OC connections cannot be reached. Our mechanism prioritizes traffic locality on the cardinality of the NS. While the results presented in Section 5.1 support our choice, the optimal choice of parameters calls for an analytical framework that we will address in our future work.

The NS should be robust against node mobility, peer departures/arrivals and peer failures. For example, due to mobility, the NS of a peer could still be reachable but with a significantly higher average hop count. Our mechanism detects any *significant* change in the 1-hop neighborhood of a peer and initiate a new discovery to adapt to a different topology. However, since the focus of this paper is on a static ad hoc network, we omit further details concerning the behavior of the protocol in a mobile scenario. Note that we assume every peer to be interested in the commodity delivered by the seed, that is, we assume the content popularity to be equal to 100

Once the distribution overlay is in place, peers can start downloading pieces of the content within the scope of their neighborhoods. In the following sections we describe how peers trade pieces of the content.

3.2 Peer selection algorithm

The bandwidth being a limited resource, a single node cannot serve every peer interested in pieces it holds at the same time. Thus, only a subset of the neighborhood of a peer, namely the active neighbor set (ANS), can be *unchoked* and receive data. All other peers which are not being served are said to be *choked*. The way the ANS is selected follows a rate-level tit-for-tat strategy (TFT) [6] in which peers will preferably serve cooperating peers. Each peer measures how fast it can download from each other peer in her NS and, in turn, serves those from whom it has better download rates. Note that if implemented strictly, the TFT strategy would lead the distribution process to stall for new peers joining the system as those peers have nothing to share. Thus, the TFT strategy is implemented for all but one slot in the ANS which is attributed to any peer in the NS, regardless of its upload rate. This so-called *optimistic unchoke,* which is the technique implemented in BitTorrent [2], allows peers to greedily sweep throughout the NS to discover faster peers than those currently belonging to the ANS (i.e., those with higher upload rates). Moreover, the optimistic unchoke phase is used to bootstrap new peers: peers that join a torrent are given the chance to download their first pieces without contribution.

Note that we make an important distinction between the leecher and the seed state. Peers that are going to be served by a seed are selected based on the last time they were unchoked. Instead of giving preference to peers with a high download rate, our algorithm favors peers that have received the least service [19].

Our peer selection technique has the goal of guaranteeing a reasonable level of upload and download reciprocation. Indeed, a peer-to-peer session consists of seeds, leechers, and free-riders, i.e., leechers that never upload data. Even if we do not explicitly address the selfishness problem that affect P2P systems in this paper, we consider that free-riders might be a subset of the leechers. With a byte-level tit-for-tat strategy, when there is more capacity of service in the system than the actual request for this capacity (this might be the case for heterogeneous ad hoc networks), the excess capacity will be lost even if slow leechers or free-riders could benefit from it. With our technique, peers are allowed to use the excess capacity eventually available in the system, but not at the expense of leechers with a higher level of contribution. Reciprocation is fostered and free-riders are penalized. Note that seeds do not make a distinction between contributing leechers and free-riders. However, free-riders cannot compromise the stability of the system because the more contributing leechers, the less the free-riders can exploit the system.

In the following section, we describe the trading strategy used by peers to decide which pieces of the content will be selected for replication.

3.3 Piece selection algorithm

In our scheme we introduce a further level of content fragmentation. The content is split in *pieces* and each piece is split in *blocks*. Blocks are the transmission unit while pieces are the replication unit on the distribution overlay. Our CCD protocol only accounts for transfered pieces: partially received pieces cannot be served by a peer.

The piece selection algorithm allows a peer to decide which pieces it wants to receive from remote peers. In our protocol we use the *local rarest first* strategy [2], whereby the rarest piece(s) in the NS of a peer is replicated. The statistical distribution of pieces in a NS is maintained with a signaling protocol similar to the one used by BitTorrent [2]. There are some exceptions to the normal execution of the piece selection algorithm that are used to bootstrap peers and to optimize the piece replication process. If a peer has downloaded strictly less than k pieces (this is the case for new peers joining the network), it chooses the next piece to request at random. In our implementation we set $k = 4$, as in [3]. Our mechanism uses a *strict priority policy*: when one block of a piece has been requested, the other blocks of the same piece are requested with the highest priority. This is done in order to increase the pace of the replication process, as only complete pieces can be exchanged.

4 Performance evaluation

In this section we examine the impact on the performance of our CCD scheme of the main system parameters. Furthermore, we study the scalability properties of our scheme when the number of peers in the network increases reasonably.

4.1 Simulation set-up

We use the *Qualnet* [5] network simulator. In our simulations we use the CSMA/CA 802.11a MAC protocol and use the RTS/CTS-Data/ACK mechanism. We set the data rate at 36Mbps, which leads to a $230m$ data radio range in free-space[3].

In our simulations we use the unicast proactive ad hoc routing protocol OLSR [4]. Data packets carrying pieces of the content as well as signaling messages are sent using the TCP transport protocol while control messages exchanged by peers in the overlay construction phase use the UDP transport protocol. We motivate our choice (see Section 2.2) by the need for reliable transfer of pieces and some sensitive control messages of our CCD scheme.

In the first part of our evaluation we focus on different topologies whereby nodes are placed uniformly over a square area whose size determines the connectivity properties of the network. The first network we focus on is characterized by a high average node degree: an instance of such a topology is depicted in Figure 1. Every node has a number of neighboring nodes that let peers reach a relatively high cardinality of the NS without the need to expand the search perimeter by more than 2 hops.

The second network we study is characterized by a low node degree: Figure 2 provides a snapshot of such a topology. In this case, peers are able to reach a relatively high cardinality of their NSs at the cost of expanding the search perimeter by more hops as compared to the network in Figure 1.

In our experiments the size of the content distributed by the source is set to 5MB[4]. The content is split into pieces of 16 blocks, each block being of size 16KB. The choice of piece and block sizes follow the heuristic proposed by BitTorrent and by the work presented in [20]. We run several experiments to understand the impact of different piece and block sizes: due to space limitations we do not present these results as the impact on performance is negligible.

[3] This value has been determined using the Qulanet utility to calculate the radio range based on the channel model, the antenna model and receiver sensitivity.

[4] The reason why we do not focus on larger content sizes follows what has been observed in [22], where the authors collected data on torrents advertised by the isohunt website. Small sized files are not uncommon in the Internet and are more realistic in the MANET context.

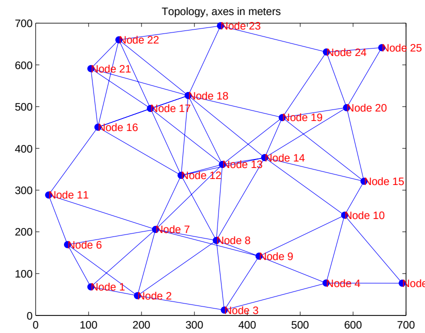


Figure 1: 25-node topology with high node degree.

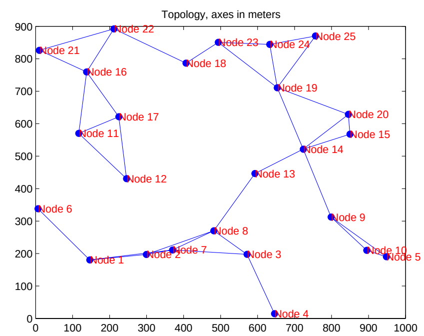


Figure 2: 25-node topology with low node degree.

We assume peer arrival rates to be representative of a *flashcrowd* scenario: peers bootstrap the content distribution overlay and fetch the content at the same time t.

Each point in the following plots is the average result over 5 independent simulation runs for every simulation setting.

4.2 Performance metrics

The cooperative content distribution problem, as defined in Section 2, consists in finding a distribution schedule that minimizes the time to distribute all the pieces of the content to all receivers. We define the *time to download* metric (TTD) that indicates the time at which a peer received the whole content. This metric is useful to discern which system configuration, *e.g.* the parameters that characterize the overlay construction technique, provides better results. We study the cumulative distribution of TTD for every peer and compare results focusing on the median and variance of the TTD distribution. If it is straightforward to assign a *preference* *order* on lower median values, we argue that also a lower variance in the TTD distribution is preferable. A low variance indicates that all peers roughly complete the download of the content at the same time, leading to a system-wide fairness.

In Section 5.3 we discuss on the performance of our scheme using measures taken at different layers of the protocol stack. We define an *energy consumption* metric that indicates the energy consumed at the physical layer. The energetic model implemented in Qualnet follows the one presented in [10]: only the transmission and reception of data consumes energy while no energy is consumed in idle state. We also focus on the number of retransmissions at the MAC layer and at the TCP layer. At the MAC level, we measure the aggregate number of *RTS retransmissions* due to the expiration of the timeout for the reception of the CTS and the aggregate number of *Packet retransmissions* due to the timeout for the reception of the corresponding ACK message. At the transport level, we measure the aggregate number of *message retransmissions* and the aggregate number of *message fast-retransmissions*. We also evaluate the *Average Download Rate* metric derived from the content size and the TTD distribution.

5 Simulation results

5.1 Impact of the overlay structure

In this section we examine the impact of the distribution overlay structure on the performance of our CCD mechanism in terms of the TTD metric. We focus on the topologies showed in Figures 1 and 2 in which one source peer distributes the content to 24 peers. The overlay structure is determined by the union of the neighbor sets (NS) of all peers: we thus focus on the parameters that define the NS. The NS is characterized by its cardinality, *i.e.*, the number of peers in the neighborhood which is equal to $OC+IC$, and by the average hop count of the peers in the NS, which is determined by the search horizon (*i.e.,* MAX_TTL). For a static network, the node degree (which is proportional to the node density) determines to a large extent the need for expanding the search perimeter in order to reach the desired NS cardinality.

We define two sets of reference parameters, one for the topology depicted in Figure 1 and one for the topology in Figure 2. For the topology characterized by a high node degree (Figure 1) we define two configurations, one in which only neighboring nodes are selected as part of the distribution overlay, and one in which we allow the expanding ring search technique to seek for nodes which are at most 2 hops away from a peer constructing its neighborhood, see Table 1.

(a) Search ring = 1 hop

OC =	3
MAX_TTL =	1

(b) Search ring = 2 hops

OC =	5
MAX_TTL =	2

Table 1: Overlay network parametersfor the topology in Figure 1

For the topology characterized by a low node degree (Figure 2) we define three configurations. In the first case we only select neighboring nodes to be part of the distribution overlay. We then allow the expanding ring search technique to seek for nodes which are at most 2 hops away and at most 3 hops away, see Table 2.

(a) Search ring = 1 hop

OC =	2
MAX_TTL =	1

(b) Search ring = 2 hops

OC =	3
MAX_TTL =	2

(c) Search ring = 3 hops

OC =	4
MAX_TTL =	3

Table 2: Overlay network parametersfor the topology in Figure 2

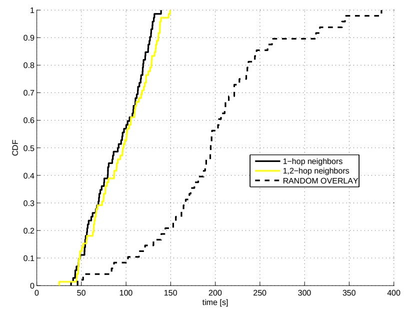
Note that we use IC=4 for all experiments. While we examined the effects of a large number values, in this paper we only report values that provide insights on the role of distribution overlay. For each configuration we evaluate the cumulative distribution of the TTD, and derive statistical information on the median and variance of the TTD. As explained in Section 4.2 we look for the configuration leading to lower median and variance values of the TTD.

Figures 3(a) and 3(b), show the CDF of the TTD for the two reference networks, emphasizing the configuration used to build the distribution overlay.

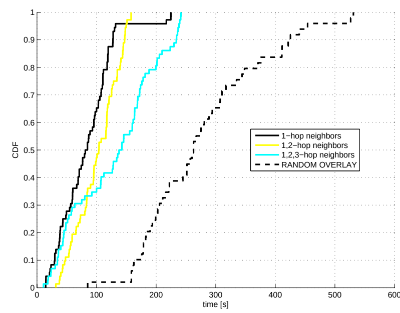
The results in Figure 3(a) show that expanding the search boundaries to obtain a NS of higher cardinality has a negligible effect on the TTD cumulative distribution[5]. This is mainly due to interference and collision issues at the MAC level that we will analyze in more detail in Section 5.3. As opposed to what has been observed for CCD mechanisms used in a wireline context such as the Internet (see for example [22]) the benefits that derive from a ”rich” neighbor set (both in terms of peers and piece diversity) are neutralized by the shared nature of the wireless medium. However, Figure 3(b) shows that the simple heuristic of constructing only 1-hop NS might be sub-optimal. On one hand, for topologies with a low average node degree (Figure 2) a 1-hop NS might result in an unfair distribution of the content since nodes are penalized by exponentially increasing timeout values at the MAC level, as shown by the inactivity period in Figure 3(b). On the other hand, performance degrade when the search perimeter is increased too much because of poor TCP performance on long routes [15]. For the sake of completeness, we compare the technique described in Section 3.1 with a centralized approach equivalent to the *tracker* component in BitTorrent [3]. We simulate the presence of a central entity to bootstrap the distribution overlay in a random fashion, as described in Section 2. Figures 3(a) and 3(b) show that traffic locality determines to a large extent the performance of our CCD mechanism. Both the median and the variance of the TTD increase when a randomized approach is used.

We also studied the impact of the position of the initial seed in the network, using as a reference the topology in Figure 1. In general, the seed position determines the length of the routes used to reach all potential destinations. However,

[5]Figure 3(a) shows that a 1-hop NS is slightly preferable, following the preference order given in Section 4.2.



(a) Reference topology in Figure 1



(b) Reference topology in Figure 2

Figure 3: Impact of the overlay structure: distribution of the time to download the content for different network topologies. With our technique an increase of the number of outgoing connections (OC) can improve performances at the cost of expanding the search ring. System performance severely degrade with a randomized approach to build the distribution overlay.

the route length could adversely impact the performance of the distribution process in a MANET context. Figure 4 shows that a central or peripheral position have a little impact on the performance of our solution. Indeed, because of our particular distribution overlay, the content propagates along routes with a bounded hop count.

Note that an analytical model to study the service capacity of CCD schemes for visibility graphs could help in finding the *optimal* configuration for our technique, and we will consider this research direction as part of our future work.

5.2 Detailed protocol behavior

In this section we explore the behavior of the proposed CCD mechanism in more details. We chose the parameters of the overlay construction technique such that OC=3, IN=4 and MAX_TTL=2. We analyze the behavior of the CCD protocol using the following evaluation criteriae for both networks represented in Figures 1 and 2. In this paper we only present results for the topology in Figure 1, as similar results hold for the latter topology.

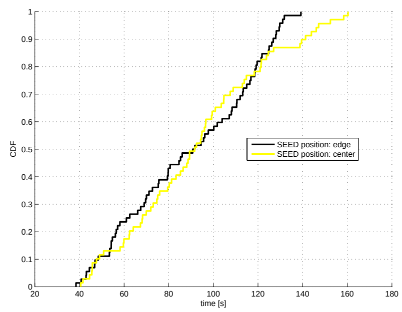


Figure 4: Impact of the position of the initial seed on the cumulative distribution of the time to download the content for the topology in Figure 1. The impact of a central or peripheral position of the content source does not severely impact the solution proposed in this paper.

- *Piece propagation*: for every leecher in the system, we study the reception time of the first and the last piece of the content[6]. This allows to deduce the speed at which pieces propagate in the network at two significative time instants and to observe any variation of that speed in the distribution process.

- *Aggregate upload capacity*: we evaluate the service capacity of our system by summing up the upload rates for every peer. Upload rates are obtained by summing the total bytes uploaded by a peer in a timeslot $t_C = 10sec$.
  This metric allows to deduce the efficiency of our CCD mechanism. However, due to the lack of a theoretical upper bound of the service capacity to which our results can be compared to, our results only show a quantitative measure of what is achievable in terms of service capacity in a static ad hoc network.

- *Number of parallel downloads*: the number of concurrent transmissions in the system is calculated by summing up the number of unique peer identities involved in the reception of a block of the content in $t_{PAR} = 10sec$.
  The number of parallel transmissions is related to the spatial reuse that we achieve in the network due to the distribution overlay structure.

In Figures 5, 6 and 7 we show the piece propagation, the aggregate upload capacity of the system and the number of parallel downloads for the reference topology depicted in Figure 1.
Figure 5 illustrates a relatively high propagation speed for the first piece of the content: in less than 30 seconds all leechers get their first piece. However, the propagation speed rapidly decreases: the high dispertion in the reception time of the last piece show that the initial thrust is not sustained throughout the content

[6]Note that piece are not ordered: the first/last piece received does not correspond to first/last piece index.

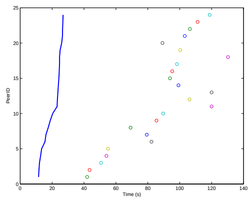


Figure 5: First and last piece download progress for the topology in Figure 1. PeersID are ordered based on the first piece reception time

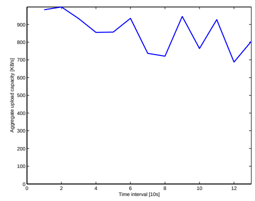


Figure 6: Aggregate upload capacity of the system in a 10 sec. time interval for the topology in Figure 1.

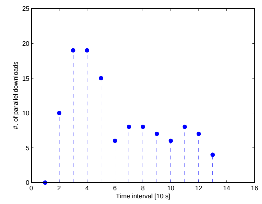


Figure 7: Number of parallel downloads in a 10 sec. time interval for the topology in Figure 1.

distribution process. In Section 5.3 we show that this decrease in performance is mainly due to the shared nature of the wireless medium.

Figure 6 shows that the service capacity of the system, in the simple scenario of a static network, is characterized by a relatively high (around 850 KBps) value. The fluctuating behavior can be partially explained by the fact that the leechers that completed their download and that could serve other peers as new sources, cannot build a new NS due to the distance in hops of the remaining leechers. On one hand, this result suggests a differentiation of the overlay construction technique between the two system states.

In Figure 7 we show a metric related to the spatial reuse of the network. The localized approach used to build the distribution overlay allows multiple communications to take place at the same time, as shown by the high number of parallel downloads. This result is somewhat in contrast with the intuition that suggests a low spatial reuse achievable in networks with a high node degree. We note a sharp drop in the number of parallel downloads as the piece propagation progress in time[7]: when the content distribution process reaches a full regime (*i.e.*, all peers are able to trade pieces) the effects of interference and collisions at the MAC level are preponderant.

The results in Figures 5, 6 and 7 show that the multi-hop nature of the network and the shared medium used to communicate can produce negative effects on the system performance that our solution partially mitigates by appropriately constructing the overlay structure.

It should be noted that our results hold for a system whereby peers follow the prescribed cooperative behavior in that they remain in the overlay to seed the content for leechers and for new peers joining the network. The next step in the analysis of CCD protocols for MANETs will be to tackle the problem of un-cooperative

[7]The decrease in number of parallel downloads is also marginally due to the increasing number of peers that completed their download. However a simple comparison between Figures 5 and 7 shows that at time 60 seconds only 5 peers completed their download while the number of parallel downloads dropped from 15 to 6.

peers that quit the overlay as soon as they finish to download the content. On one hand we showed that new seeds in the system might be ineffective if an appropriate technique to build the distribution overlay is not in place; on the other hand the early departure of leechers that complete the download and that are topologically close to the initial seed might reduce the overall system performance, as the *stretch* between the overlay and the network might increase. The shared nature of the medium complicates the task because the need for new seeds to distribute the content is in conflict with the goal of reducing interference and collision issues at the medium access control layer.

In the next section we further extend our study by inspecting important system parameters at different layers of the protocol stack. Due to the lack of alternative content distribution mechanisms to compare our results to, we use as a reference basis the performance of a simple client-server (FTP) approach.

5.3 Performance comparison: CCD vs. FTP

In this section we compare the performance of our CCD mechanism to that of a traditional client-server approach to transfer a file from one source to all nodes in the network. We use a FTP server running on the source (which is the node with ID=1 in all experiments) of the content and a FTP client running on the remaining nodes of the network. Our experiments are carried out for both the topologies depicted in Figure 1 and Figure 2, however we only present results for the former case due to space limitations. Leechers and FTP clients initiate the download process at the same time.
We assess the properties of our scheme based on two metrics, the total time to transfer the content to all receivers (TTD) and the amount of energy consumed by every peer in the network, as defined in Section 4.2.

Figure 8 and 10 show respectively the cumulative distribution of the TTD for the network in Figures 1 and 2 for the CCD and the FTP schemes. We observe that the CCD mechanism performs better than the FTP scheme: for both network types the median of the TTD is smaller for the CCD case (roughly 100 seconds as compared to 180 seconds) than for the FTP case. Especially, the CCD scheme outperforms a FTP based solution if we take into account the variance of the TTD or the maximum time needed for the content to be distributed to all nodes in the network. The dedicated nature of FTP data flows suffers from the underlying network topology. Indeed, the maximum time for the content to be distributed through a topology with a low node degree (and longer routes) doubles as compared to a topology with a high node degree. In contrast, the experiments we carried out on the network in Figure 2 have an equivalent TTD cumulative distribution for the CCD case, leading to the conclusion that our solution mitigates the impact of the underlying network configuration.

The benefits of our CCD scheme appear also evident if we measure the total amount of energy consumed by the nodes during the content distribution process. As it is possible to see in Figure 9 and 11 the energy consumption in the FTP sce-

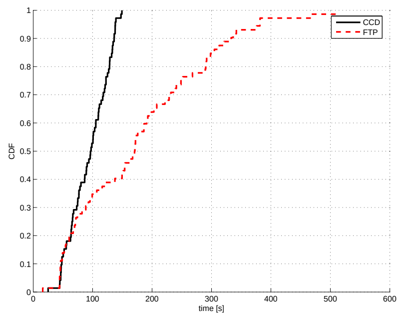


Figure 8: Cumulative distribution of the time to download the content for our CCD scheme and for the FTP mechanism for the topology in Figure 1. The CCD case outperforms the FTP case for both the median and the variance of the TTD distribution.

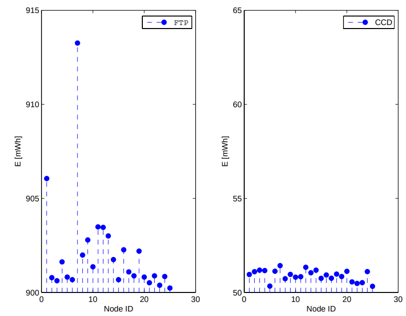


Figure 9: Energy consumption comparison between CCD and FTP case for the topology in Figure 1. The CCD solution remarkably consumes less energy (an order of magnitude) and has the property of equalizing the costs associated to the content distribution among all peers.

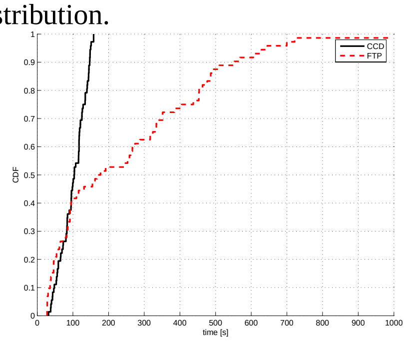


Figure 10: Cumulative distribution of the time to download the content for the CCD and the FTP case for the topology in Figure 2. The maximum time to distribute the content for the FTP case is higher as compared to Figure 8 because of the underlying network topology, while the CCD scheme is not highly impacted.

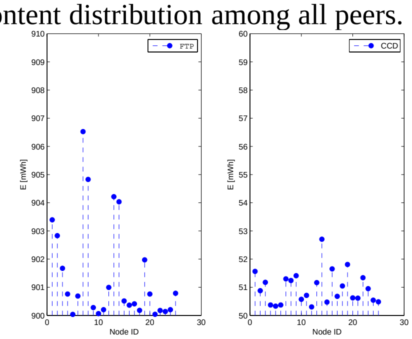


Figure 11: Energy consumption comparison between CCD and FTP case for the topology in Figure 2. The difference in energy consumption is remarkable for the CCD and FTP cases. The consequence of a topology with low node degree and a small number of alternative shortest paths can be seen also for the CCD solution by considering the uneven energetic consumption.

nario differs from the CCD case *by an order of magnitude*. Moreover, it is possible to observe that in the FTP case the source of the content as well as those nodes on the shortest path from the source to all destinations consume a considerably higher amount of energy as compared to other nodes in the (static) network. In the CCD case, the energy consumption is remarkably equalized. The seed does not consume more energy than any other leechers, and the leechers consume roughly the same

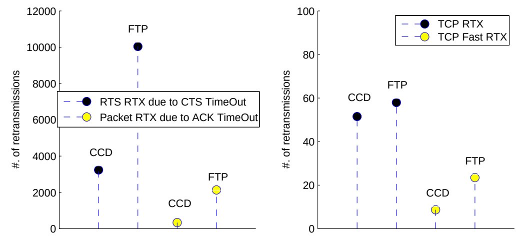


Figure 12: Average number of retransmissions at the MAC and Transport Layers for the topology in Figure 1. The impact of MAC retransmissions due to interference and collisions is remarkable as compared to retransmissions at the transport layer.

amount of energy.

With the aim of explaining the difference in energetic consumption between the CCD and the FTP case, we focus on the topology represented in Figure 1. In Figure 12 we represent the average number of retransmissions measured in the system focusing on MAC and transport layer retransmissions, as explained in Section 4.2. Whereas the number of retransmissions at the transport layer (right plot in Figure 12) does not explain the remarkable difference in energy consumption, the number of retransmissions at the MAC layer can be considered as the main factor that differentiate the CCD and the FTP approach both in terms of energetic consumption and performance. The loss of MAC-level control and data packets triggers the exponential backoff mechanism typical of a legacy 802.11 protocol, which leads a node to wait an increasingly larger amount of time before (re)transmitting a packet. Hence the poor performance and the higher energy consumption in the FTP case.

5.4 Scalability properties

In this section we study the scalability properties of our CCD mechanism. In our experiments we follow the *extended network model* [9] whereby we maintain the density of the nodes in the network constant, while increasing both the number of nodes and the (square) area on which nodes are deployed. We scale the network up to 100 nodes while a control script ensures the resulting network not to be partitioned before executing the simulation.

Results are presented in Figure 13, where we show the average download throughput of a peer, *i.e.,* the *per-peer capacity*, calculated using 5 independent simulation runs.

Figure 13 shows that the per-peer download rate does not scale well with the system size. With the sake of fitting our experimental values to an analytical expression we report a scaled version of the per-node capacity law as introduced by Gupta *et. al.* in [14]. Although the scaling law obtained in [14] refers to a random network deployed on a unit disk area (we deploy our nodes on a square area), opti-

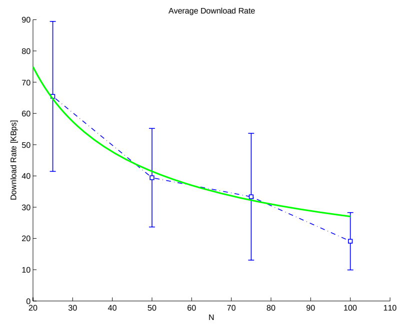


Figure 13: Per node capacity scaling under the extended network model. The network size has a severe impact on the per node capacity. In the Figure we plot a scaled version of the $1/sqrt(nlogn)$ law for sake of comparison with our experimental results.

mal scheduling of packet transmission and random source-destination pairs, in the interval limits of our evaluation study, the experimental and the analytical curves loosely match and differ only by a constant factor.

This result shows that, in the interval of our study, the traffic locality imposed by the overlay structure does not significantly improve the scaling properties inherent to the underlying network, as the result in [18] would have suggested. Nonetheless, our results hold for a realistic packet scheduling; using random source-destination pairs (*i.e.*, a random overlay network) would have yield worse performance; current analytical models neglect the effects of a transport protocol such as TCP on the scaling law of the network. Understanding the resulting combination of constructs with different scaling properties calls for an analytical framework that we will address in our future work.

6 Conclusion and future work

In this paper we address the problem of content distribution in static multi-hop networks. We propose a solution based on swarming protocols and on a decentralized mechanism to build the content distribution overlay. Our technique takes into account traffic locality as a key factor to overcome the limitations due to the underlying multi-hop network and the shared nature of the medium used by wireless nodes to communicate. With our solution the cost associated to the content distribution is evenly shared among the peers and cooperation is fostered while the eventual excess capacity in the system is used for the benefit of the peers with scarce upload capacity.

Our simulation results show that the performance in terms of the total time to distribute the content of our mechanism are reasonably good for scenarios characterized by different network topologies. We show that the energetic consumption associated to the content distribution is evenly shared among the peers and that our scheme mitigates the effects of MAC-level retransmissions due to interference and collisions. We conclude our analysis showing that the per-node capacity scaling law of our solution decreases as the one predicted by analytical models for an idealized system, despite the traffic locality imposed by the overlay network.

In our future research we will present performance result for the mobile case and focus on new scenarios whereby we will vary the content popularity, the peer departure rates and the peer cooperation level. We will also focus on the definition of an analytical framework to assess performance bounds for the general content distribution problem in multi-hop networks.

References

[1] Akamai technologies. http://www.akamai.com/.

[2] Bittorrent protocol. http://wiki.theory.org/BitTorrentSpecification.

[3] Bittorrent web site. http://www.bittorrent.com/.

[4] Optimized link state routing protocol: Internet draft. http://hipercom.inria.fr/olsr/draft-ietf-manet-olsr-11.txt.

[5] QualNet simulation suite. http://scalable-networks.com.

[6] R. Axelrod. *The Evolution of Cooperation*. ISBN: 0465021212, 1984.

[7] B. Cohen. Incentives Build Robustness in BitTorrent. In *Proc. of the 1st Workshop on Economics of Peer-to-Peer Systems*, Berkeley, USA, 2003.

[8] M. Conti, E. Gregori, and G. Turi. A cross-layer optimization of gnutella for mobile ad hoc networks. In *Proc. of the 6th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, New York, NY, USA, 2005.

[9] O. Dousse and P. Thiran. Connectivity vs capacity in dense ad hoc networks. In *Proc. of the 23rd IEEE Conference on Computer Communications (Infocom)*, Hong Kong, China, 2004.

[10] L. M. Feeney and M. Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *Proc. of the 20th IEEE Conference on Computer Communications (Infocom)*, Anchorage AK, USA, 2001.

[11] M. Gerla, K. Tang, and R. Bagrodia. TCP Performance in Wireless Multi-hop Networks. In *Proc. of the 2nd IEEE Workshop on Mobile Computer Systems and Applications (WMCSA)*, Washington, DC, USA, 1999.

[12] C. Gkantsidis and P. Rodriguez. Network Coding for Large Scale Content Distribution. In *Proc. of the 24th IEEE Conference on Computer Communications (Infocom)*, Miami, USA, 2005.

[13] M. Grossglauser and D. N. C. Tse. Mobility increases the capacity of ad hoc wireless networks. *IEEE/ACM Trans. Netw.*, 10(4):477–486, August 2002.

[14] P. Gupta and P. R. Kumar. The capacity of wireless networks. *Information Theory, IEEE Transactions on*, 46(2):388–404, 2000.

[15] G. Holland and N. H. Vaidya. Analysis of TCP Performance Over Mobile Ad Hoc Networks. In *Proc. of the 5th IEEE/ACM International Conference on Mobile Computing and Networking (Mobicom)*, August 1999.

[16] C. Killian, M. Vrable, A. C. Snoeren, A. Vahdat, and J.Pasquale. Brief announcement: the overlay network content distribution problem. In *Proc. of the 24th Annual ACM Symposium on Principles of Distributed Computing (PODC)*, Las Vegas, NV, USA, 2005.

[17] C. Killian, M. Vrable, A. C. Snoeren, A. Vahdat, and J.Pasquale. The Overlay Network Content Distribution Problem. Technical Report CS2005-0824, UCSD, CS Dept., 9500 Gilman Drive, La Jolla, CA 92093-0404, USA, May 2005.

[18] J. Lee, C. Blake, D. S. J. De Couto, H. I. Lee, and R. Morris. Capacity of Ad Hoc wireless networks. In *Proc. of the 7th Annual International Conference on Mobile Computing and Networking (Mobicom)*, Rome, Italy, 2001.

[19] A. Legout, G. Urvoy-Keller, and P. Michiardi. Understanding bittorrent: An experimental perspective. Technical Report INRIA-00000156, November 2005.

[20] A. Nandan, S. Das, G. Pau, and M. Gerla. Co-operative downloading in Vehicular Ad-hoc Wireless Networks. In *Proc. of the 2nd Annual Conference on Wireless On-Demand Network Systems and Services (WONS)*, Washington, DC, USA, 2005.

[21] R. Sherwood, R. Braud, and B. Bhattacharjee. Slurpie: A Cooperative Bulk Data Transfer Protocol. In *Proc. of the 23rd IEEE Conference on Computer Communications (Infocom)*, Hong Kong, China, 2004.

[22] G. Urvoy-Keller and P. Michiardi. Impact of Inner Parameters and Overlay Structure on the Performance of BitTorrent. In *Proc. of the 9th IEEE Global Internet Symposium Workshop, In Conjunction with IEEE Infocom 2006*, Barcelona, Spain, 2006.

[23] X. Yang and G. de Veciana. Service Capacity in Peer-to-Peer Networks. In *Proc. of the 23rd IEEE Conference on Computer Communications (Infocom)*, Hong Kong, China, 2004.