Connectivity-Driven Attachment
in Mobile Cellular Ad Hoc Networks

Julien Boite and Jérémie Leguay

Thales Communications & Security, Gennevilliers, France
{julien.boite,jeremie.leguay}@thalesgroup.com

Abstract. Cellular wireless technologies (e.g. LTE) can be used to build *cellular ad hoc networks*. In this new class of ad hoc networks, nodes are equipped with two radio interfaces: one being a terminal, the other one being an access point. The nodes can initiate only one outgoing attachment towards a neighbor using their terminal interface, while they can receive multiple incoming attachments through their access point. In this context, attachment decisions based on traditional criteria (e.g. signal quality) may lead to network partitions or suboptimal path lengths, thus making access point selection critical to ensure efficient network connectivity. This paper proposes a distributed greedy attachment strategy to reach near optimal network connectivity. Through simulations, we show that our strategy leads to network connectivity almost as good as in pure ad hoc networks, with small impact on path length. We also study the impact of thresholds to avoid low value attachments.

Keywords: Cellular Technologies, Ad Hoc Networks, Attachment Strategy, Network Connectivity.

1 Introduction

Most ad hoc networks have been conceived and deployed with wireless equipments that use a dedicated ad hoc radio mode. This is the case for WiFi community networks such as FunkFeuer in Austria [1] or Guifi [2] in Spain, as well as for public safety [3] and tactical networks [4]. This paper considers a particular class of ad hoc networks, that we call *cellular ad hoc networks*, where wireless technologies in cellular mode such as WiFi (infrastructure) or LTE are used in a specific way to build an ad hoc network. This kind of setup has been shown to be promising [5] as cellular modes offer higher data rates than ad hoc modes, when available, in off-the-shelf products. This paper proposes a distributed attachment strategy to maximize network connectivity for such networks.

In *cellular ad hoc networks*, nodes are composed of two radio interfaces as presented in Fig. 1: one acting as a terminal, the other one acting as an access point. This combination allows each node (i) to attach to one of its neighbors using the terminal interface, and (ii) to receive connections from neighboring nodes thanks to the access point radio interface. The network links are thus built
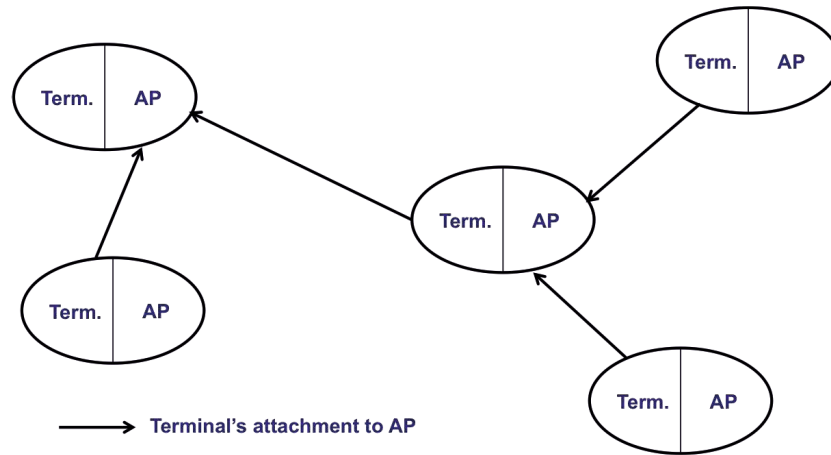
Fig. 1. Cellular ad hoc network

from the attachments that nodes initiate with one another using their terminal interface. A routing protocol makes multi-hop communications possible between connected nodes.

By *mobile cellular ad hoc networks*, we mean that the above-described nodes can move. Due to nodes mobility, attachments may change over time, leading to a dynamic network topology that has to be carefully maintained to ensure that the highest number of nodes can communicate. Attachment decisions from one node to another are the unitary network events that build the network topology. However, while neighbors in the radio range can communicate all together using the traditional ad hoc mode, the cellular mode introduces a particular constraint: the terminal interface of a node can attach to only one neighbor node. Therefore, the attachment decisions have to be made with caution since they directly impact the network topology and, subsequently, the global network connectivity. This paper proposes an attachment strategy to maximize the global network connectivity in this context.

Traditionally, off-the-shelf terminal interfaces apply an attachment strategy which is driven by pre-configured preferences and signal strength maximization. They select the access point to which they attach or towards which they should start a handover based on signal quality, user preferences or historical data. Because these local decisions do not consider connectivity at the network scale, they can create problematic connectivity conditions as depicted in Fig. 2. As shown in the figure, the network may (i) be disconnected while the radio range would allow full connectivity or may (ii) suffer from long routes between nodes that could be closer in the network space. To avoid such situations, we propose to drive attachment decisions by the objective of maximizing connectivity from a global network viewpoint. With this approach, ad hoc nodes dynamically attach or handover to a neighboring node so that connectivity is maximized, i.e. so that the number of node pairs that could exchange data is the highest possible.
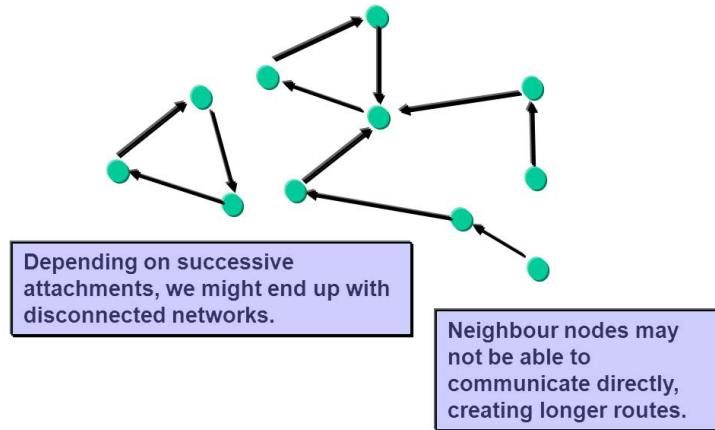
Depending on successive attachments, we might end up with disconnected networks.

Neighbour nodes may not be able to communicate directly, creating longer routes.

Fig. 2. Connectivity issues may arise with signal-based attachment strategies

Other issues should be addressed when deploying a cellular ad hoc network such as the allocation of non-overlapping channels to the different access point interfaces of nodes. Distributed spectrum allocation techniques may be used to solve this issue [6] in case that the number of channels is lower than the number of nodes. In this paper, we assume that this problem has been solved and focus on making good attachment decisions with regards to network connectivity.

We propose a distributed attachment mechanism that significantly improves network connectivity. It adopts a greedy approach by attaching smaller network partitions to larger ones. We show that this strategy can be implemented on top of existing technologies such as WiFi or LTE, and requires only one piece of information to be shared between nodes using existing control channels. We present a performance evaluation in simulation and compare our distributed mechanism with the optimal solution (pure ad hoc) and the state of the art solution (signal strength based attachment strategy). Two main criteria have been considered: (i) the percentage of connected pairs of nodes, and (ii) the average path length between connected pairs. Our mechanism outperforms traditional attachment strategies and offers close to optimal network connectivity while the average path length is just a bit longer. As frequent changes in attachments may engender a high number of handovers and thus cause disruptions to ongoing flows, we introduce thresholds to avoid handovers that do not bring significant gain on network connectivity or would impair too many flows. With identified threshold values, we show that near-optimal results still hold when only the most beneficial handovers are triggered.

The remainder of the paper is structured as follows. Section 2 presents the related work on cellular ad hoc networks. The greedy strategy that we propose and its implementation are presented in Section 3. Section 4 shows the performance evaluation. Finally, Section 5 concludes this paper and gives some perspectives for future work.

2   Related Work

The combination of access point and terminal functions to build a cellular ad hoc networks has been first proposed in 2007 [7]. However, this setup has only received an interest from the research community rather recently. In [5], authors have evaluated this approach using WiFi nodes. They have introduced preference information so that nodes do not oscillate when they have several neighbors, as the signal strength may vary back and forth between one neighbor and another. They have also introduced the possibility to turn off the access point part on nodes in case of high density. Indeed, when all nodes are in range, a good solution can be that everybody attach to the same node. In [8], authors have considered virtualization functionalities on a single WiFi card to allow a node to act as an access point and a terminal at the same time. The paper proposed a tree-based logical structure to perform routing that could be maintained following the attachments.

A number of techniques have been proposed to modify the access point selection procedure in wireless networks. In [9], the authors propose to run a battery of performance tests once being attached. If the connection is not satisfying enough, another access points is evaluated. One of the key feature of our proposition is that the decision can be achieved pro-actively from the information retrieved in the discovery phase. To the best of our knowledge, there is no related work that addresses the problem of making efficient attachments with regards to network connectivity in cellular ad hoc networks.

3   A Greedy Attachment Strategy

The objective of the mechanism we propose is primarily to maximize the number of nodes that can communicate. For this, we adopt a greedy attachment strategy that connects nodes to their best connected neighbors. The decisions are made locally by each node, using two sources of information: (i) the network topology that nodes maintain, limited to the network part they belong to, and (ii) a simple property of the network part behind neighboring nodes. Section 3.2 explains how this property, that we call *connectivity information*, can be exchanged with no overhead. As a secondary purpose, the algorithm tries (i) to minimize the number of flows that may be impaired by reattachment decisions, and (ii) to improve the connectivity of already connected areas by minimizing the average path length. Overall, the attachment decisions trade off between improving connectivity (increasing the number of connected pairs of nodes, reducing the average path length) and reducing flow impairments.

3.1   Architecture

The distributed attachment mechanism we propose relies on a few system modules embedded on each node. Fig. 3 presents the main interactions between the attachment module, that executes the attachment algorithm detailed in

section 3.3, and other system components from both the network side (routing, flow management) and the radio side (wireless interfaces). These interactions and their purpose are described below.
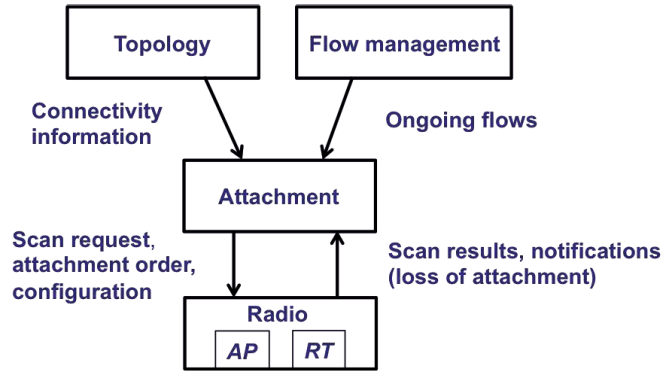

Fig. 3. Main interactions between the attachment module and other system modules

Topology module: the attachment module needs topology information to make decisions. It relies on the topology module to maintain such information, built from the routing protocol exchanges. This information represents the network topology around the node, i.e. a graph with nodes and links between them within the node's connected network area. We call this network area composed of reachable nodes the node's *connected component* in the rest of this paper. From this topology view, a node can thus verify if a neighboring node is already in its connected component and thus determine if it is worth attaching to it (the more new nodes a neighboring node allows to reach, the more interesting the attachment to this node). The attachment module can also compute properties of the node's connected component such as its size (number of nodes) and the average length of paths between nodes. These performance indicators from a connectivity viewpoint are key information used in the attachment decision process.

Radio module: depending on the implementation, this module sits either on top of two radio interfaces (represented in Fig. 3 as the access point AP and the relay terminal RT) or on top of one interface having both functionalities. To discover neighboring nodes, the attachment module can ask the radio module (more precisely the RT part) for neighbors discoveries. Most of cellular technologies support this action which is generally called a scan. Nodes can thus retrieve, for each neighboring node, the frequency used, the signal strength, and other specific information which depends on the technology (e.g. SSID for WiFi). In addition, when an attachment decision is made, the attachment module sends an order to the radio module (RT) so that the relay terminal performs the attachment. The attachment module can also ask for radio reconfigurations such as frequency change or meta-data updates (e.g. the SSID announced by the AP). This allows to take benefit of existing communication channels, used by

access points to announce meta-data, for exchanging *connectivity information* between nodes during neighbors discoveries (cf. next section). Finally, the radio module notifies the attachment module of connectivity events, especially when the terminal losses its attachment.

Flow management module: an attachment decision made while the node is already attached can impact ongoing flows. The number of these flows and their relative importance are thus parameters that the attachment module should take into account. The flow management module maintains a list of flows relayed by the node. The attachment module can gather this list and thus evaluate the impact of a reattachment, e.g. according to QoS or criticity policies.

Note that the solution presented so far can be extended without losing its general principles. For instance, using several terminal interfaces per node is one possible extension.

3.2   Connectivity Information Exchange

In order to make good attachment decisions, a node needs to compare the properties of its connected component with those of candidate neighbors. One node can easily compute parameters relative to its connected component from the graph maintained by the Topology module. However, when a candidate neighbor is not in the connected component of the node, the required information is not available since this neighbor in not in the local graph. For that reason, connectivity information must be exchanged between nodes during a neighbor discovery process.

In the attachment process we present in Section 3.3, the only required connectivity information is the size of each neighbor's connected component. Exchanging this small piece of information can be implemented using several means. It can be achieved using a dedicated communication channel. However, off-the shelf cellular technologies do not support this explicitly. Existing fields can be used instead, such as SSID in WiFi or the PLMN (Public Land Mobile Network) identifier in LTE. To take benefit from these native communication channels, the attachment module in the architecture described above can configure the radio module (AP) so that these text fields are dynamically updated. A node can thus update its SSID AP_ID_X with X the connectivity information announced to other nodes, basically the number of nodes in its connected component.

3.3   Attachment Decision

The attachment procedure evaluates the interest of attaching to a neighbor with regards to the global network connectivity. It considers the following factors:

1. Opportunity to increase the size (number of nodes) of the node's connected component. The attachment strategy firstly looks for a neighbor that would allow to increase the number of nodes that can be reached. The number of nodes and the presence of a neighbor node in the current connected component is an information retrieved from the Topology module. The size of the

connected components to which neighboring nodes belong is obtained during the neighbor discovery process. Since the decision to modify an attachment can disrupt or impact ongoing flows (packet losses, increased path length and delays), the attachment strategy estimates this impact and finally decides whether or not to change the attachment. Information on ongoing flows are retrieved from the Flow management module.

2. Opportunity to reduce paths length inside the node's connected component. When the size of the connected component cannot be increased, the attachment strategy looks for a neighbor that would substantially reduce paths

---

Algorithm 1. Pseudocode for the attachment process.

---

function UPDATEATTACHMENT
    ▷ Retrieve list of neighbors and their respective connectivity information
    triggerNeighborsDiscovery()
▷ Find neighbor with highest Connected Component (CC) that is not in node's CC
    bestNeighbor ← *findBestNeighborOutsideCc*()
    if (! node.*isAttached*()) **then**   ▷ The node is not attached
      if (bestNeighbor != NULL) **then** ▷ A candidate neighbor outside CC exists
        attachTo(bestNeighbor)
      else   ▷ No candidate neighbor outside CC
        optimizePathLengthInsideCc()
      end if
    else   ▷ The node is already attached
      if (bestNeighbor != NULL)   ▷ A candidate neighbor out of CC exists
&& (bestNeighbor.*getCcSize*() ≥ node.*getCcSize*()) **then**   ▷ and has a larger CC
               ▷ Simulate handover to compute flow disruption rate
        flowDisruptionRate ← *computeFlowDisruptionRate*()
        if (flowDisruptionRate < thresholdFlowDisruptions) **then**
          attachTo(bestNeighbor)
        end if
      else   ▷ No candidate neighbor outside CC
        optimizePathLengthInsideCc()
      end if
    end if
end function

function OPTIMIZEPATHLENGTHINSIDECC()
           ▷ Find neighbor that minimizes average path length inside CC
    bestNeighbor ← *findBestNeighborInsideCc*()
    if (bestNeighbor != NULL) **then**   ▷ A candidate neighbor inside CC exists
          ▷ Simulate attachment to compute average path length gain
      pathLengthGain ← *computePathLengthGain*()
      if (pathLengthGain > thresholdPathLengthGain) **then**
        attachTo(bestNeighbor)
      end if
    end if
end function

length in the connected component to optimize network connectivity. The average paths length before and after simulated attachments to candidate neighbors can be computed from the connectivity graph maintained by the Topology module.

The attachment procedure is greedy as it permanently tries to increase the size of nodes' connected component. Algorithm 1 details this procedure that applies in several cases: (i) periodically when nodes are not attached to any neighbor, (ii) immediately after a node loses its attachment, and (iii) periodically when nodes are already attached with the aim to further increase connectivity, reduce path length, or update the topology according to the mobility of nodes.

This attachment strategy makes use of two thresholds to control reattachments:

- *thresholdFlowDisruptions* prevents reattachments if too many flows are impacted (weights can be associated to the importance of flows),
- *thresholdPathLengthGain* is defined to avoid changes of attachment if the gain generated on the average path length is not high enough (it filters low value attachment decisions and increases network stability).

The combination of these two thresholds allows to trade off between the improvement of connectivity and the negative impact that frequent changes of attachments may have.

4   Performance Evaluation

Our distributed attachment mechanism has been evaluated in the simulator shown on Fig 4. This tool that we have developed generates reproducible mobility
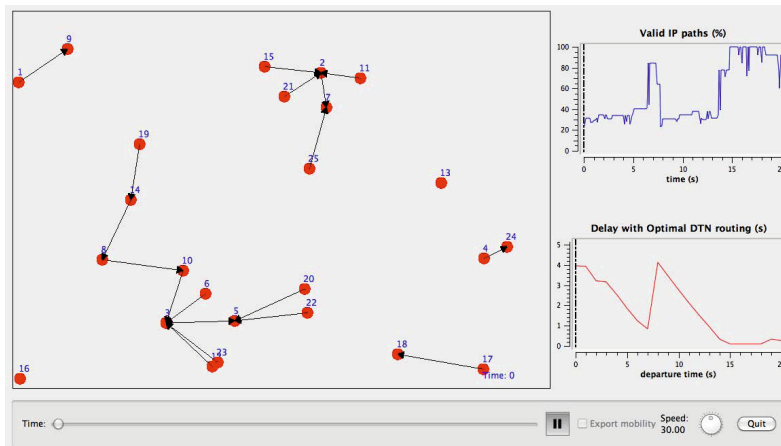

Fig. 4. Simulator GUI

scenarios using the classical random waypoint model and simulates the progressive attachments between nodes. We performed several simulations to analyze the network connectivity obtained using the following attachment strategies:

- *Signal*: a strategy that selects the neighbor with the highest signal quality
- *Optimal*: a strategy that provides the best ad hoc connectivity. This strategy simulates the traditional ad hoc interface where nodes can exchange data as soon as they are in radio range (i.e. without the constraint of the cellular mode, where one node can only attach to one of its neighbors)
- *Greedy*: the greedy strategy that we propose.

To evaluate the performance of these different strategies, we have considered the following measures during and at the end of each simulation:

- Percentage of connected pairs of nodes (i.e. for which an IP path exists)
- Average path length between connected nodes, in number of radio hops
- Average minimum Delay Tolerant Network [10] (DTN) delay, i.e. the average minimum time that a message, or *bundle*, would need to reach its destination using store, carry and forward transfers. This indicator reflects the quality of network connectivity since messages propagate from one node to another thanks to contacts between mobile nodes.
- Number of handovers (reattachments).
- Percentage of flow disruptions, i.e. the portion of flows that are rerouted or interrupted by handovers. We have considered that each node maintains a flow with every nodes in its connected component.

We have considered a scenario with 25 nodes and performed simulations to (i) analyze the performance results and compare our greedy attachment strategy against the others, and to (ii) focus on our greedy strategy to study the impact of thresholds we can set to keep only the most beneficial reattachments (maximum percentage of impacted flows, minimum gain in average path length). The table shown in Fig. 5 gives the simulation parameters that are common to all the simulations. Nodes are moving according to a random waypoint model with no pause time and a new speed selected randomly at each direction change.

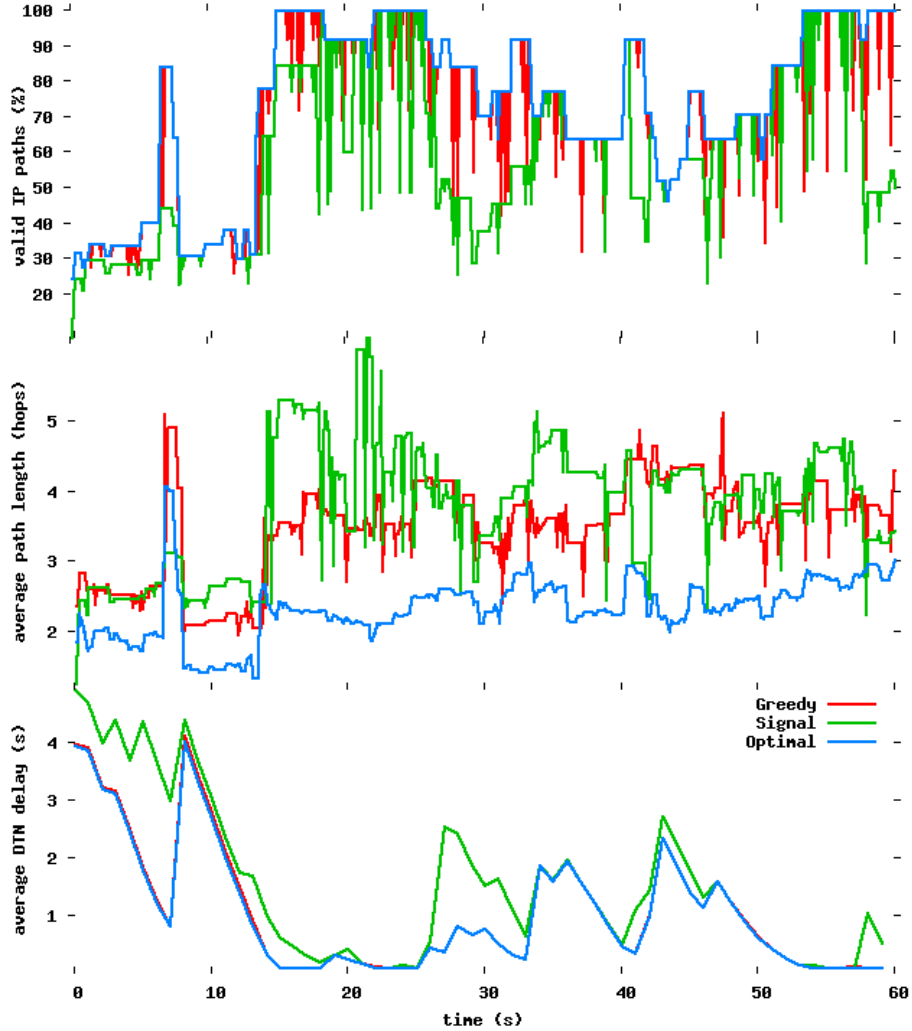| # nodes	25
Width × Heigth	600m x 600m
Radio range	120m
Mobility speed	[0:30] m/s
Simulation duration	1 minute

Fig. 5. Global simulation parameters

Fig. 6. Percentage of connected pairs of nodes (top), average path length (middle), and average DTN delay (bottom) over time

4.1   Global Performance Comparison

We have first compared the different strategies regarding their global performance results, without setting any of the thresholds defined above for the greedy strategy. Fig. 6 presents several simulation results. From top to bottom, these plots show the evolution over time of (top) the percentage of connected pairs of nodes, (middle) the average path length, and (bottom) the average DTN delay. We can observe that:

– the percentages of connected pairs of nodes for Greedy and Optimal are very close. We analyzed that the average percentage of connected pairs of nodes (over the whole simulation) for Greedy, Signal and Optimal is respectively 70.4%, 59.8% and 73.2%. The greedy strategy is thus very close to optimal while Signal leads to less than 60% of pairs of nodes connected in average. This confirms the network connectivity problem that occurs with traditional attachment strategies, and demonstrates the efficiency of our greedy strategy to alleviate this problem: the network connectivity is near-optimal almost all the time.

– the average path length for Greedy is higher than for Optimal, but most of the time lower than for Signal. We analyzed that the average path length is 1.16 hop longer for Greedy than Optimal for which it is 2.33 hops in average. This path elongation is due to the connectivity constraint (one node can attach to only one other node) that does not exist in pure ad hoc mode. Greedy does not reach Optimal due to this constraint, but it still performs better than Signal regarding average path length.

– the plots of the average DTN delay for Greedy and Optimal overlap, which means that Greedy offers the same performance than Optimal for delay-tolerant message delivery. In average, the time that is necessary to deliver all messages sent from one node to any other node for Greedy, Signal and Optimal is respectively 1.09, 1.56 and 1.07 seconds. Greedy and Optimal plots very similar performances, as for the first result on connected pairs of nodes. This confirms that the network connectivity obtained by Greedy and Optimal is very close.

These results show that our greedy attachment strategy outperforms traditional access point selection schemes in terms of connectivity between network nodes. It provides connectivity performances very near to those of a traditional ad hoc network. The single attachment constraint results in slightly longer average path length compared to the optimal, but this does not significantly impact the delivery of data packets.

4.2   Impact of Thresholds

We also studied the impact of the threshold values we can set in order to control reattachments. These thresholds define (i) the relative gain on average path length above which the attachment change is considered valuable, and (ii) the maximum portion of ongoing flows that can be impacted (considering one bi-directional flow between each node pair). We performed several simulations with different values for each of these thresholds to evaluate their impact on the number of handovers, the average percentage of connected pairs of nodes, and the mean of average DTN delay, over a whole simulation. Fig. 7 presents these results.

In top figures we show, for several threshold values set to the minimum gain in the average path length, the impact on (left) the number of connectivity events, (middle) the average percentage of valid IP paths between nodes, and (right) the
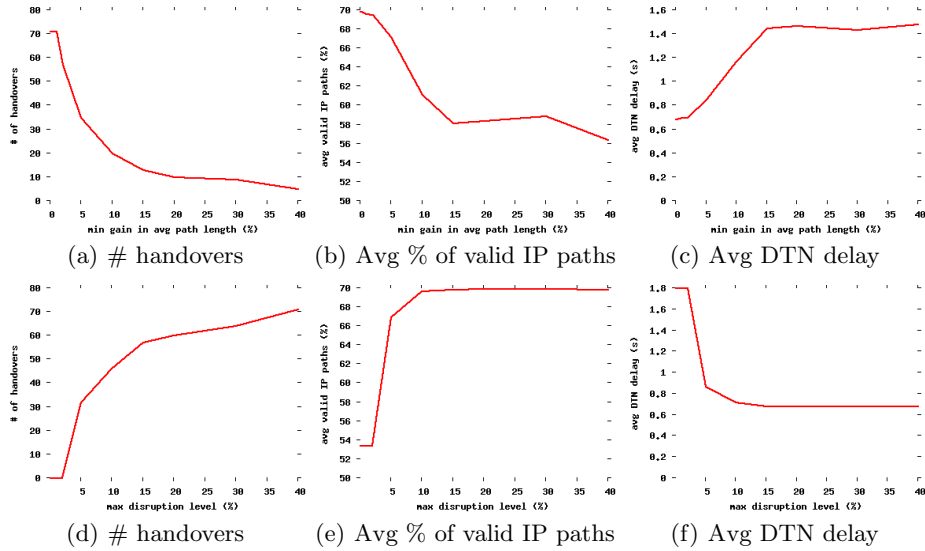
(a) # handovers   (b) Avg % of valid IP paths   (c) Avg DTN delay

(d) # handovers   (e) Avg % of valid IP paths   (f) Avg DTN delay

Fig. 7. Impact of the path length gain threshold (top figures) and the flow disruption threshold (bottom figures) on handovers, network connectivity and DTN delay

average DTN delay. We observe a fast decrease of the number of handovers when the threshold on expected gain in average path length increases (cf. Fig. 7(a)). For instance, it decreases by 50% for a threshold set to only 5%. On the other side (cf. Fig. 7(b) and Fig. 7(c)), such a low threshold value does not severely impact the connectivity performance since the average percentage of connected pairs of nodes only decreases from 70% to 67%, and the average DTN delay sligthly increases of 0.1 second. However, setting higher threshold values has a more significant impact on connectivity and DTN delay.

We can conclude that setting this threshold to a low value (such as 5%) allows to avoid many reattachments that do not generate significant gains in average path length, while it has a limited impact on network connectivity and average DTN delay.

In bottom figures we show, for several threshold values set to the maximum portion of ongoing flows a reattachment can disturb, the impact on (left) the number of connectivity events, (middle) the average percentage of valid path between nodes, and (right) the average DTN delay. As expected, we see from Fig. 7(d) an important increase of the number of handovers when the threshold value increases. Indeed, many reattachments do not occur with low threshold values. For instance, the number of connectivity events is divided by 2 for a threshold around 7%. For such a low threshold value, we observe from Fig. 7(e) that the average portion of valid IP paths between nodes is not so much affected, while important losses of connectivity occur for a threshold under 5%. The average percentage of connected pairs of nodes rapidly converges towards a maximum of 78% for a threshold set to 10% and above. Combining these two

results, we can say that many handovers that would disturb ongoing flows do not significantly improve the network connectivity. This confirms the relevance of limiting such reattachments. We also note from Fig. 7(f) that setting the threshold under 5% can lead to higher DTN delay.

We can conclude that setting this threshold to a low value (7 to 10%) allows to avoid many reattachments that do not improve connectivity so much while they would generate flow disruptions, with limited impact on the DTN delay.

These two thresholds are finally useful to trade off between achieving the best connectivity and reducing the negative impact of frequent changes of attachments. To highlight this, we ran one more simulation to evaluate our greedy strategy with the threshold on flow disruptions set to 10%, that we call Greedy.10%. Fig. 8 compares the results to those obtained with Optimal and Greedy. Using this threshold preserves a near-optimal network connectivity. At the same time, we analyzed that the number of handovers decreased by 54%. Using only a low value for the threshold on flow disruption thus allows to filter many handovers that are not useful while preserving a near-optimal network connectivity.
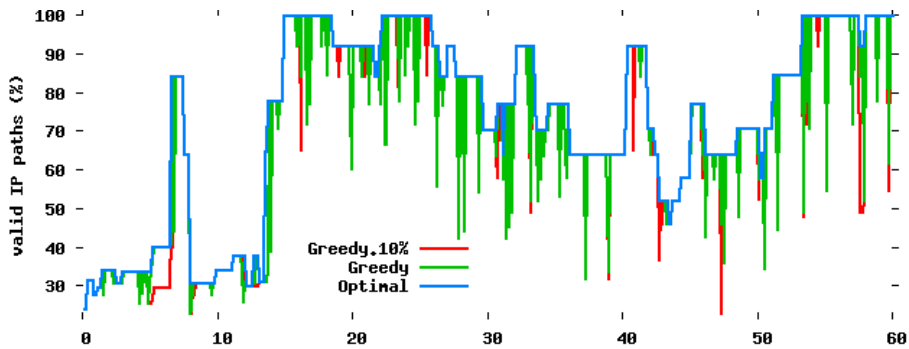

Fig. 8. Influence of the flow disruption threshold on the percentage of connected pairs of nodes

5   Conclusions and Perspectives

This paper has presented a greedy attachment strategy in order to improve the global network connectivity in mobile ad hoc networks built with cellular wireless technologies. Our strategy is driven by the objective of maximizing the number of nodes that can communicate so that the network connectivity is the best possible. We have presented a practical way of implementation that takes benefit from existing control channels of WiFi and LTE technologies to exchange connectivity information between nodes with no overhead. Through simulations, we show that our strategy leads to a network connectivity almost as good as in pure ad hoc networks, while the average path length is a bit longer. We also study the impact of thresholds we fix on the gain in average path length and on flow disruption to maximize the utility of reattachments. The evaluation we made has

shown that this strategy gives results close to the optimal and outperforms the traditional signal strength based strategy. The thresholds that can be configured to control reattachments allow to trade off between best connectivity and the impact of those reattachments.

The attachment decision strategy presented in this work may be included in a large multi-criteria algorithm, taking for instance into account signal strength, the social proximity of nodes as it influences traffic patterns, or the number of terminals already attached to a node. We plan to study such complex decisions in future work. We also plan to work on prototyping activities. We conjecture that the path elongation that we observed between cellular and pure ad hoc networks should be compensated by the more efficient MAC layer (infrastructure mode) used in cellular ad hoc networks.

References

1. FunkFeuer: FunkFeuer: free net, `http://www.funkfeuer.at`
2. Guifi: Guifi network, `http://guifi.net/`
3. Knopp, R., Nikaein, N., Bonnet, C., Aiache, H., Conan, V., Masson, S., Guibe, G., Martret, C.: Overview of the widens architecture, a wireless ad hoc network for public safety. In: Proc. of SECON (2004)
4. Iovine, R., Bouis, J.: The flexnet-waveform in the international sdr arena. In: Military Communications Conference. In: IEEE MILCOM (2009)
5. Wirtz, H., Heer, T., Backhaus, R., Wehrle, K.: Establishing mobile ad-hoc networks in 802.11 infrastructure mode. In: ACM Chants (2011)
6. Cao, L., Zheng, H.: Distributed spectrum allocation via local bargaining. In: Proc. of SECON, pp. 475–486 (2005)
7. Bereski, P., Soulie, A.: Flexible radio network (patent), `http://patent.ipexl.com/EP/EP1936872.html`
8. Sarshar, M.H.: Poo K.H., Abdurrazaq, I.A.: NodesJoints: A Framework for Tree-Based MANET in IEEE 802.11 Infrastructure Mode. In: IEEE Symposium on Computers and Informatics (2013)
9. Nicholson, A.J., Chawathe, Y., Chen, M.Y., Noble, B.D., Wetherall, D.: Improved access point selection. In: Proc. of ACM MobiSys (2006)
10. Fall, K.: A delay-tolerant network architecture for challenged internets. In: Proc. of ACM SIGCOMM (2003)