

# Social Sharing of Connectivity Resources: Control and Encouragement of Unselfishness in Mobile Environments

Paolo Bellavista, Carlo Giannelli

DEIS – University of Bologna

Viale Risorgimento, 2 – 40136 Bologna - Italy

Tel.: +39-051-2093001; Fax: +39-051-2093073

{paolo.bellavista, carlo.giannelli}@unibo.it

## ABSTRACT

In the last years has clearly emerged the opportunity of extending traditional single-hop wireless technologies for Internet connectivity, by introducing practical effective solutions to dynamically use the best multi-hop heterogeneous paths available at runtime. Our primary idea is to enable, via proper, context-aware, and effective middleware, the mass of mobile devices already in the market to collaborate together toward innovative forms of Multi-hop Multi-path Heterogeneous Connectivity. Potential advantages are multiple, from extended wireless coverage to "green" cost reduction via cooperative sharing, from balancing energy consumption of collaborative nodes to maximizing overall bandwidth. In particular, this work-in-progress paper originally concentrates on the challenging issue of promoting connectivity sharing via effective forms of network management and control, capable of i) monitoring the selfish/collaborative behaviors of participants in a very lightweight way, ii) fairly distributing relay duties in order not to penalize too much "generous" nodes, and iii) rewarding cooperativeness by limiting the consumption of shared resources and by privileging unselfish participants. To that purpose, the paper presents original solution guidelines based on the regional fairness principle to achieve effectiveness and limited overhead; they can relevantly contribute to encourage connectivity sharing in open and dynamic deployment environments.

## Categories and Subject Descriptors

C2.8 Mobile Computing, C2.8.c Mobile Communication Systems, C.2.8.e Support Services, C2.3.a Network Management, J.9.d Pervasive Computing.

## Keywords

Wireless Computing, Always Best Served Connectivity, Connectivity Sharing, Middleware, Effective Management, Fairness.

## 1. INTRODUCTION

The wide availability of mobile terminals with heterogeneous (and often multiple) wireless interfaces is stimulating a number of novel research activities, with the primary goal of exploiting at

best all the potential Internet connectivity opportunities available at runtime. The primary idea is to dynamically determine and build opportunistic, multi-hop, and heterogeneous paths toward the Internet, by collaborating with mobile terminals in proximity for connectivity resource sharing. The approach is based on the observation that nodes frequently tend to underutilize their resources (not only connectivity ones) and are ready to offer part of them. In this process, it is crucial that potential benefits (either personal or community-oriented) are correctly perceived and understood, as demonstrated by successful file sharing applications in the recent past. By following this approach, we have recently worked on the design and implementation of a context-aware and effective middleware to enable off-the-shelf portable devices to collaborate together toward innovative forms of Multi-hop Multi-path Heterogeneous Connectivity (MMHC) [1-3]. Our MMHC middleware effectively supports the dynamic collaboration of existing devices/equipment to share connectivity resources by playing the role of either i) *infrastructure connectors* that are usually fixed and offer direct connectivity to the Internet through them, such as existing IEEE 802.11 Access Points (APs) and GPRS/UMTS Base Stations (BSs), or ii) *peer connectors* that are usually mobile and provide ad-hoc connectivity towards other peers, e.g., based on Bluetooth Personal Area Network (PAN) or IEEE 802.11 Independent Basic Service Set (IBSS) connections.

In previous work we have demonstrated the relevance of properly and effectively managing the wide set of dynamically available MMHC opportunities. We have pointed out the crucial role of context for efficient evaluation of MMHC opportunities [1] and presented the architecture of the MMHC middleware [2, 3]. Our MMHC prototype supports the self-organization of cooperating nodes, by dynamically retrieving the set of available infrastructure/peer connectors. In addition, MMHC clients self-hail multi-hop paths to the Internet, by modifying routing rules at runtime in the case of intermediate peer failure. To achieve these goals effectively, MMHC enables the visibility of several innovative, coarse-grained, and lightweight context indicators: node mobility prediction to infer link reliability, throughput estimation to evaluate connectivity quality, and estimated energy of peer connectors to infer path durability.

However, our current MMHC prototype does not estimate users' expected behaviors, which can greatly affect the overall sharing performance. For instance, shared bandwidth availability may relevantly decrease if the owner of the connector node exploits traffic-intensive services, e.g., FTP downloading. We claim that, to help the widespread diffusion of self-organizing forms of connectivity, there is the need to push for a "more social" attitude in resource sharing, i.e., through automated solutions to advise users if they are utilizing shared resources in a "non-social" way, by

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Mobimedia'09, September 7-9, 2009, London, UK.

Copyright 2009 ICST 978-963-9799-62-2/00/0004...\$5.00.

possibly punishing them in case of repetitive misbehavior, e.g., via bandwidth limitation.

For social connectivity sharing, and more generally for any other kind of resource sharing in highly dynamic and open environments, it is appropriate to have lightweight mechanisms and highly decentralized strategies to favor partial and approximated forms of control over the cooperative system. The aim is to encourage "fair and community-proper" behavior of cooperating nodes while discouraging excessive selfishness (i.e., excessive parsimony in offering own resources and excessive greediness in requesting them to other nodes). In fact, it is expectable that MMHC users are willing to have some form of monitoring/control on the exploitation of their own resources by other nodes, for instance to preserve a quota of effective Internet bandwidth to their local applications at any time. At the same time, rewarding the node availability to share resources could effectively stimulate and motivate participation to the community. Let us rapidly point out that our purpose is not to support maximum fairness, e.g., by providing exactly the same bandwidth to any collaborating node despite its position, but only to detect possible misbehaviors in order to push users for resource sharing.

By following these observations, this work-in-progress paper presents original solutions for ensuring effective and fair exploitation of shared resources in order to realize multi-hop multi-path connections. The principle is to enable lightweight forms of network management and control of local/remote nodes in a MMHC-specific and appropriate way. First of all, we propose the active monitoring of local/remote behaviors to estimate if nodes act in an either collaborative or selfish manner. Secondly, our solution controls and modifies the behavior of local/remote nodes in order to fairly share traffic load due to packet relaying that occurs in intermediary nodes along the activated paths. Finally, we encourage traffic sharing by rewarding collaborating nodes that behave as peer connectors: the basic idea is to provide these nodes with wider bandwidth than nodes acting only as clients. The proposed solution is completely decentralized and based on a small set of context indicators spread within a regional scope, i.e., either gathered locally or generated and communicated by nodes at single-hop distance. Our regional scope permits, on the one hand, to impose a very limited computing/communication overhead and, on the other hand, to actually induce emerging behaviors on participants, with no need of expensive multi-hop management connections.

The remainder of the paper is organized as follows. Section 2 presents our objectives and solution guidelines, also stemming from practical considerations about the performance of the off-the-shelf wireless interfaces that are currently available. Section 3 rapidly gives the needed background about the MMHC middleware and its architecture, in order to fully understand the design/implementation details of the following original proposal for the promotion of connectivity sharing (Section 4). Conclusive remarks and directions of current research work end the paper.

## 2. SOLUTION GUIDELINES FOR SOCIAL CONNECTIVITY SHARING

Based on our experience, we have identified two primary and general objectives to fulfill in order to leverage novel social connectivity sharing scenarios. The first objective is to provide Inter-

net connectivity **effectively**. The second goal is to support **fairness** in resource exploitation.

About the first objective, we address the issue of supporting the Internet access of any node belonging to the MMHC network, despite its position in the MMHC topology and the behavior of other cooperating nodes. Let us stress that the performance of the multi-hop wireless paths supported by MMHC greatly differs from what it is possible to achieve in traditional wired networks. In fact, as better detailed in the following, nodes close to infrastructure connectors would tend to gain relevantly larger bandwidth than nodes that exploit a long multi-hop path. Our primary idea is to support effective connectivity of any MMHC by inducing proper network topologies and by avoiding any form of starvation for nodes far from infrastructure connectors.

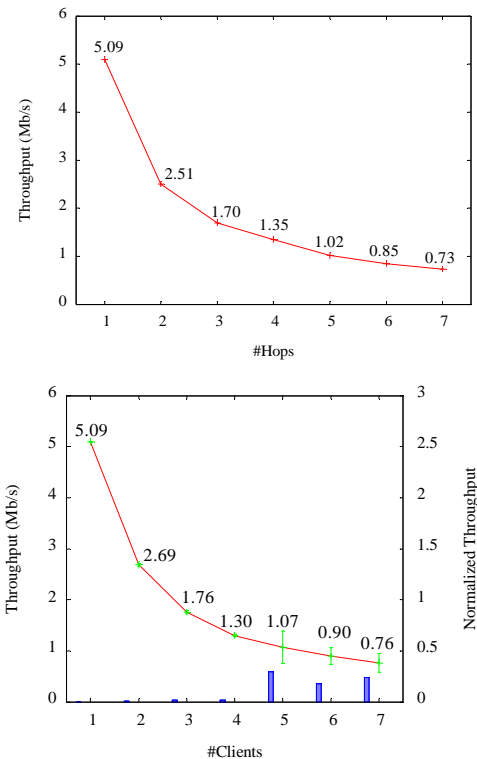
Once connectivity is provided with reasonable effectiveness, i.e., by supporting Internet access of any node with a minimum quality, the MMHC objective is to push for a fair exploitation of connectivity resources, from the points of view of both connectors and clients. On the one hand, we aim at not harming the own personal connectivity capabilities of connectors. In fact, in social connectivity sharing applications, connector nodes are expected to be inclined to lower their own throughput (generated by their local applications) to allow connectivity of their clients. But, of course, they expect their own throughput not to decrease too much. Otherwise, they will tend not to cooperate anymore in the MMHC network. On the other hand, clients should receive shared bandwidth depending on their previous behavior, e.g., wider bandwidth to nodes serving as connectors in the past and tighter bandwidth to nodes only getting connectivity. That would enable an effective rewarding for cooperative nodes, thus encouraging connectivity sharing. Let us note that the objective is not to guarantee maximum fairness, e.g., same throughput for any node with similar requirements in the MMHC network, but only to "equalize" the actual connectivity consumption of nodes that, without any countermeasure, will receive very different connectivity qualities depending on their position in the MMHC topology.

In our opinion, to achieve the above goals and to permit the full exploitation of resources in multi-hop wireless environments, there is the crucial need for a deep and thorough understanding and experimental characterization of the behaviors of wireless connectors, of their peculiarities, and of their limitations. In the following, we show some preliminary performance results, deriving from our experimental campaigns over both simulation and in-the-field environments. We claim that these quantitative indicators, even if coarse-grained and preliminary, are relevant for suitably deciding effective and practical solution guidelines in MMHC scenarios. In particular, we have measured how it is possible to achieve maximum throughput in multi-hop heterogeneous paths and how performance results may greatly vary in case of multiple clients getting cooperative connectivity concurrently.

By delving into finer details, we have experimentally observed that three primary factors influence the maximum throughput of heterogeneous multi-hop paths: i) the wireless technology of each single-hop sub-path, ii) the number of hops in the path, and iii) the number of clients/peers simultaneously served by each node in the path. Other factors, that have non-negligible effects on connectivity quality, e.g., node mobility, are anyway not so influential in a very first approximation. Figure 1 - Up shows the average throughput of a multi-hop path in saturated bandwidth conditions

and depending on path length. Reported results are from ns-2 simulations and have been confirmed by extensive in-the-field observations. The one-hop path demonstrates to reach a throughput slightly greater than 5Mb/s (around the typical maximum throughput of IEEE 802.11b). By increasing the length of the sender-receiver path, the throughput lowers dramatically (less than 1Mb/s in a 7-hop path). This is mainly due to the interference of neighbor nodes, especially relevant when different wireless interfaces on the same node exploit similar frequencies. In fact, each peer connector usually serves as a forwarder, thus potentially producing collisions among incoming and outgoing packets. This effect even increases with longer paths (larger number of intermediate nodes).

Figure 1 - Down shows the average throughput depending on the number of senders, each one at single-hop distance from the receiver. Senders have shown to share the available bandwidth quite uniformly in saturation conditions: only with 5 or more senders the normalized throughput (namely the ratio between the standard deviation and the average throughput) slightly increases. In other words, to a first approximation, even when there are many senders, generally the throughput is fairly shared and almost constant (slightly greater than 5Mb/s). By comparing Figure 1 - Up and Down, it is possible to observe that, while the trends are almost the same, multi-hop throughput tends to decrease slightly quicker than multi-client one.



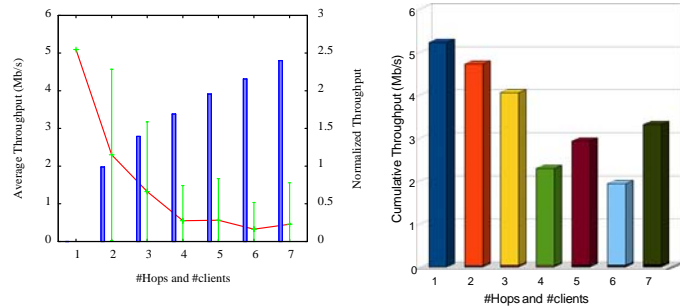
**Figure 1. Up: single-client throughput depending on path length. Down: single-hop throughput depending on number of clients: average (line), standard deviation (error bars), normalized throughput (impulses).**

Another central characteristic of multi-hop paths is that nodes achieve a throughput much lower than the maximum possible when intermediate connectors concurrently request Internet connectivity. To this purpose we have evaluated the performance results of a multi-hop multi-client path, by measuring throughput when increasing the number of hops and clients. For example, in the case of a 3-hop path (Figure 2), 3 clients (C1, C2, C3) reside on the same path and send data to the same receiver (R).



**Figure 2. Multi-hop multi-client paths.**

Figure 3 reports quantitative performance indicators in the specific cases ranging from 1-hop 1-client to 7-hop 7-client paths. The figure points out two important aspects of multi-hop multi-client paths. On the one hand, the total throughput is slightly lower if compared with paths of the same length but with only one client. On the other hand, the bandwidth is not fairly shared among nodes, even in the case of only two senders, as demonstrated by the value of the standard deviation and normalization throughput. Considering the two-hop case, note that a normalized throughput of about 1 means that, while the average throughput is about 2.15 Mb/s, one of the two nodes receives almost 4.30 Mb/s while the other almost nothing.



**Figure 3. Average throughput at increasing number of hops and clients (a client at each hop): average (line), standard deviation (error bars), normalized throughput (impulses).**

**Figure 4. Cumulative throughput of clients: the first node achieves almost the whole bandwidth.**

Moreover, Figure 4 shows the throughput distribution among senders in the previous deployment case: almost the whole bandwidth is provided to the node closest to the receiver; the other nodes achieve only really limited throughput (see little rows at the bottom of 2-7 bars). That highlights how the exploitation of multi-hop multi-client paths tends to distribute connectivity quality very unfairly among the cooperating nodes.

Given the above observations, the primary goal in MMHC is to maximize and harmonize the available throughput of all cooperating nodes by switching among the available paths, also depending on predictions and coarse-grained estimations of the maximum throughput achievable. We have already developed a MMHC prototype able to infer and compare the quality provided by different paths based on coarse-grained and lightweight estimations, used to select the paths with estimated best quality [1].

The estimation of maximum achievable throughput has demonstrated to be useful to effectively compare the available paths and to activate only the most promising ones. However, it neither provides hints about the actual throughput achievable at runtime nor supports the equalization of bandwidth among the cooperating nodes. To that purpose, with effectiveness and fairness as the primary objectives of our work-in-progress, originally presented here, we have determined the following guidelines to significantly extend our MMHC solution to promote the participation to social scenarios of connectivity resource sharing.

The first guideline is to **effectively monitor**, in a reasonably lightweight way, node behavior in order to distinguish between too selfish and collaborative nodes, e.g., respectively, that generate traffic starvation for other nodes or offer most bandwidth to their connected clients. Maximum throughput evaluation should be coupled with mechanisms to enable the on-line monitoring of currently activated MMHC paths. The second guideline is to manage and induce node behaviors in a decentralized way in order to **fairly share the load** of traversing traffic within the community of cooperating nodes. In fact, peer connectors usually desire to maintain control on their own connectivity resources, e.g., by limiting the bandwidth offered to other nodes depending on the requirements of locally running applications. The third guideline is to **reward nodes** that behave cooperatively. Peer connectors should be able to evaluate the behavior of their clients and reserve different connectivity quotas depending on the observed degree of cooperativeness, e.g., whether the nodes currently asking for connectivity have provided or not connectivity resources to other nodes in the past. In this way it is possible to effectively and practically push for collaborative attitudes, by stimulating resource sharing via the offering of better expected quality in successive interactions.

We also claim that these guidelines should be coupled with efficient solutions to limit the impact on cooperative nodes due to communication and computational overhead. The main idea is the dynamic adoption of a proper trade-off between global and local management mechanisms. Global, complete, and updated knowledge of the MMHC topology and of its characteristics may incur in intolerable overhead. Only-local lightweight operations may have the drawback of taking management decisions that are far from being optimal. Our proposal is to follow a **regional scope**, by limiting the scope of monitoring/management actions on the managed node and its one-hop-distant neighbors. The goal is to enhance the overall MMHC connectivity (all the cooperating network opportunities as a whole) by monitoring and influencing the behavior of nodes via only regional scope considerations.

### 3. MMHC ENVIRONMENT

We have designed and implemented our MMHC middleware, already presented in [2, 3], by carefully considering novel and lightweight context indicators about reliability, quality, and durability. The MMHC open-source prototype is freely available for the wireless management research community at its companion Web site <http://lia.deis.unibo.it/research/MMHC/> in different distributions for the most spread operating systems (Linux, MSWindowsXP/Vista, and MacOSX).

Figure 5 gives a high-level overview of the MMHC architecture. Its main components are Network Interface Provider (NIP) that offers homogeneous management access to heterogeneous wireless interfaces, Connector Manager (CM) that enables and man-

ages single-hop wireless links, and Routing Manager (RM) that properly configures multi-hop paths.

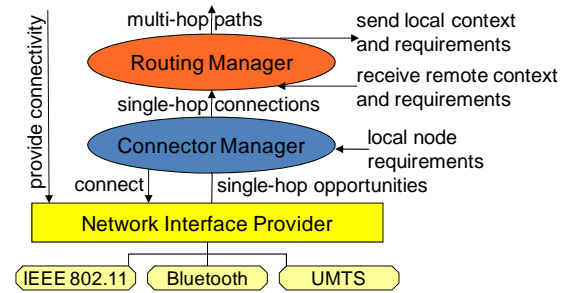


Figure 5. Our MMHC middleware architecture.

In particular, NIP is in charge of providing a common API by hiding low-level details of underlying drivers and operating systems. CM gathers RSSI sequences to evaluate connector reliability for any single-hop MMHC opportunity. On this basis, it takes local decisions on the subset of single-hop paths to activate for performance sake. RM, instead, is in charge of managing routing rules for multi-hop path construction. It works to send/collect information/requirements on path suitability to/from collaborating nodes to the purpose of the coarse-grained estimation of path quality/durability. Further details are in [2, 3].

From the network layer point of view, CM is in charge of creating single-hop links to newly discovered networks, e.g., by performing IEEE 802.11 associations to APs and configuring IP parameters via DHCP. If a node has multiple interfaces, CM can connect the node to different networks simultaneously. Instead, RM is in charge of interconnecting different networks, by modifying local routing rules and activating NAT mechanisms in order to solve potential conflicts between different namespaces. Note that interconnected networks may be based on heterogeneous wireless solutions, such as Bluetooth and UMTS. For instance this is the case of Node A in Figure 6, which accesses the Internet via a UMTS BS while it provides connectivity via a Bluetooth interface. The main purpose of MMHC is to effectively organize and handle Internet connectivity, by exploiting context information for coarse-grained evaluation of which are the expected best collaborative paths to forward packets to, e.g., based on estimated maximum throughput.

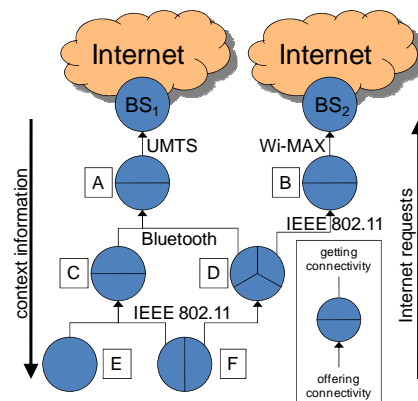


Figure 6. A simple example of multi-hop multi-path scenario enabled by MMHC.

Let us point out that CM/RM management leads to the establishment of a tree-network topology where path determination is triggered from the bottom and directed toward the Internet via one of the activated MMHC paths [1]. For instance, node A can get access to the Internet via  $BS_1$  but cannot exploit MMHC to get a shared file from node E. Instead, MMHC context indicators are distributed along a top-down direction, e.g., node A can inform its one-hop neighbor clients that it is able to provide connectivity at a given estimated throughput, but nodes C and E cannot directly inform node A about their current situation, e.g., suffering from traffic starvation. This simplifying approach was adopted in the current MMHC prototype to ensure fast configuration of self-organizing multi-hop paths with limited management costs.

Our original proposal here is of enhancing the MMHC middleware to exploit more deeply all the potential of MMHC spontaneous networking by encouraging cooperative behaviors. To this purpose there is the need for a more sophisticated mechanism for the distribution of context indicators among participating nodes. For instance, it could be useful to inform, in a bottom-up way, node A about the number of nodes its neighbor clients are currently serving, e.g., to offer larger bandwidth if this number grows and A has not strict requirements for its local applications. While nodes should be aware of a wider set of context information, at the same time we claim that it is unfeasible to pursue global and consistent awareness of context indicators in order to limit management overhead. Maintaining context awareness within a regional scope promises to be a suitable tradeoff for the MMHC middleware to take proper control decisions to improve connectivity effectiveness and fairness, e.g., by disadvantaging/favoring nodes in relation to the amount of locally generated traffic and the forwarding traffic they have served.

To better clarify the guidelines of our proposal with a practical example, consider the topology in Figure 6. Based on throughput estimation, Node F will correctly select the path on the right via Node D and B because this path performance is expected to be better than the one on the left (including one Bluetooth link with narrow bandwidth). Later on, when Node D starts intensively exploiting the path on the right, Node F will probably suffer from connectivity starvation, potentially forcing Node F to switch to the path on the left. However, triggered by regional-scope context indicators (gathered locally and provided by Node F), Node D can become aware of the "unfairness" of its behavior and take connectivity sharing countermeasures, such as reserving to Node F part of its bandwidth toward Node B.

Let us rapidly point out that the MMHC middleware is based on the assumption that users are willing to behave fairly, at least as far as connectivity sharing does not negatively affect their own performance too much. Trust management mechanisms to ensure the reliability of exchanged context indicators are out-of-the-scope of this paper and part of our future work.

#### 4. MMHC FAIRNESS MANAGEMENT WITH REGIONAL SCOPE

Given the above motivations, in MMHC we have decided to adopt an approximated fairness management solution with regional scope. Our original solution is based on i) local monitoring to estimate the sharing behavior of single nodes and ii) the delivery of local monitoring data to neighbor nodes in the MMHC topology. Each node works as a local monitor by comparing the

amount of traffic it generates with the one traversing it (either successfully sent or dropped packets). The primary idea is to exploit these data to understand to a first approximation whether a node (local or neighbor) is inducing/suffering from starvation (**guideline #1**).

By delving into finer details, we define Traversing Load (TL) the ratio between traversing and locally generated traffic. If TL is lower than 0.05, either neighbors are producing very limited traffic to forward or the local node is starving their traffic. Starvation is likely to occur particularly if the locally generated traffic is greater than 50% ET, where ET is the maximum estimated throughput that the local node can achieve (for the approximated dynamic determination of ET in MMHC, see [1]). In this case the local node activates the **Possible Local Starvation (PLS)** state. PLS is deactivated when TL grows over 0.25, by enabling neighbors to correctly transmit again.

In addition, we define Output Effectiveness (OE) the percentage of successfully sent packets of locally generated traffic. If OE is lower than 0.05, it is highly probable that there is a situation of local starvation. In this case the local node sends a **Remote Starvation (RS)** event to its peer connector(s). Note that in MMHC, even if a node cannot connect to the Internet through a multi-hop path, it still can exchange few traffic management data directly with its connector. Both indicators are collected periodically (every 60 seconds). In that way, MMHC limits its monitoring overhead, with an obvious tradeoff on misbehavior detection delay. However, we took that design/implementation decision because we are mainly interested in pushing for social connectivity sharing (achieved even with sporadic monitoring) and not in supporting fairness with strict time requirements.

Each node autonomously exploits the above context indicators to estimate whether and which actions are required (**guideline #2**):

- 1) as soon as the PLS state is activated, the node informs its local user of the detection of her probable misbehavior (at least to the purpose of effective connectivity sharing);
- 2) if the node receives an RS event from a neighbor but PLS has not been activated yet, it forwards the RS event to its connectors;
- 3) if the node receives an RS event and the PLS state has been already reached, it rises a **Local Starvation (LS)** event to indicate that the local node is generating starvation in its neighbors.

Point 1 allows the well-intentioned user to take proper countermeasures autonomously, e.g., by manually limiting the maximum bandwidth of an FTP client she is exploiting. It is worth noting that this mechanism can enable the positive effect of limiting local bandwidth only when strictly required: in the case that local starvation is detected but other nodes do not need bandwidth, the local node does not suffer from any bandwidth limitation because the user can simply decide to ignore limitation suggestions from the MMHC monitor.

Point 2 is motivated by the cases where a node suffers from starvation, but not caused by its connectors at one-hop distance. In this case it is advisable to propagate the RS event to upper nodes in the path, until the event reaches the connector that is the reason of current bandwidth saturation. Let us point out that this does not mean the adoption of a global scope. Instead, our management approach is regional in the sense that nodes evaluate received data

and forward them to single-hop neighbors/connectors only if required. In that way, management overhead is confined only within the subset of nodes actually interested by starvation.

Point 3 certainly addresses the most interesting cases. Based on our effectiveness/fairness guidelines, we claim that in this case the local node has to modify its local bandwidth reservation policy in order to i) favor traversing traffic of cooperating nodes in place of locally generated traffic and ii) favor the nodes currently evaluated as more cooperative (**guideline #3**).

Figure 7 shows the architecture of our decentralized mechanism for regional fairness management in the case of a node getting connectivity with interface B and offering connectivity resources through interfaces A and C. Let us anticipate that Traffic Marker and Starvation Monitor enable guideline #1, determining whether nodes behave in a selfish/collaborative way. Traffic Controller and its management policy, instead, enable guidelines #2 and #3, thus "pushing" nodes to fairly distribute traffic load and to reward the ones with more collaborative behavior.

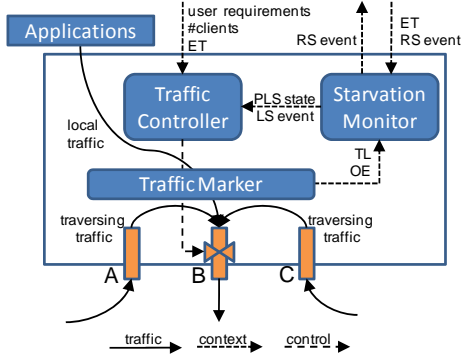


Figure 7. The architecture of MMHC fairness management.

**Traffic Marker** is the component in charge of monitoring data packets to estimate TL and OE. It is based on the capability of differently marking incoming/traversing data packets, of monitoring how data packets leaving the node are marked, and of understanding whether they are correctly delivered.

**Starvation Monitor** exploits TL and ET parameters to infer if the node is currently generating starvation to other collaborating nodes (PLS state). In addition it monitors the OE parameter to infer if the local node is suffering from starvation (RS event). On the one hand, it informs connectors in the case of the approximated evaluation that the local node is under starvation (outgoing RS event). On the other hand, it gathers information related to the possible starvation of remote nodes (ingoing RS event). These data are required to decide, to a first approximation and in a coarse-grained but efficient manner, if the local node is inducing starvation in other collaborating entities (LS event).

**Traffic Controller** has the main goal of enforcing management policies to avoid starvation. As already anticipated, the currently adopted policy simply informs the user whenever Starvation Monitor notifies the possibility that the local node is causing starvation (PLS state). Instead, whenever Starvation Monitor notifies an LS event, Traffic Controller actively manages local bandwidth allocation. The amount of bandwidth reserved to collaborating nodes reflects their degree of cooperativeness to a first approximation. For instance, wider bandwidth is offered to nodes cur-

rently serving a higher number of clients (as better detailed in the following). Note that Traffic Controller also has to modify local bandwidth allocation to avoid starvation.

To practically exemplify how our regional-scope fairness management solution works, consider the realistic case of Figure 6. Suppose that Node C is intensively exploiting the path to BS<sub>1</sub>, while both Node E and F do not generate traffic. In this case the Starvation Monitor on Node C can observe possible local starvation. However, Traffic Controller does not perform any bandwidth limitation operation because neither Node E nor Node F has notified any starvation-related event. Then, when node E starts exploiting the network, its own Starvation Monitor detects that there is no bandwidth available and informs Node C that it is suffering from starvation. Only at this point Node C starts lowering the bandwidth reserved to locally generated traffic to enable node E's Internet connectivity.

To demonstrate the feasibility and effectiveness of the above fairness management solution, we have accordingly extended our MMHC middleware prototype. In the following, without any ambition of completeness, we simply present some insights, hopefully of general interest, about the prototype implementation. The main purpose is to point out i) how our solution can recognize a possible starvation and ii) which actions it performs to autonomously take appropriate countermeasures to fairly share traffic load and to reward collaborative clients.

As already stated, the mechanism underlying both starvation identification and differentiated bandwidth reservation is the differentiated marking of traffic packets. To this purpose we use the `mark` option of the Linux `iptables` command in the `mangle` table. In that way, MMHC distinguishes packets generated locally (`output` chain), packets sent to the local node (`input` chain), and packets simply routed to other nodes (`forward` chain). To correctly mark packets coming from different clients, we activate a new marking rule whenever the local DHCP server (instantiated by the MMHC middleware) provides a new lease. For instance, in case the DHCP server assigns the IP address 192.168.1.8, our solution creates the following marking rules:

```
i) iptables -t mangle -A forward -s 192.168.1.8 -j mark --set-mark 8
```

activated on connectors to mark packets coming from a newly discovered client;

```
ii) iptables -t mangle -A output -s 192.168.1.8 -j mark --set-mark 8
```

activated on the client requesting connectivity resources to mark packets generated locally and sent to the new connector.

The activation of management rules is performed by exploiting the `tc` command:

```
tc class change dev $dev parent 1:1 classid 1:1 htb rate $bw prio 1
```

The command is executed whenever a client joins/leaves the network. It creates a new traffic class based on the Hierarchy Token Bucket discipline, particularly suitable for reserving different amounts of bandwidth to different classes of traffic flows. It specifies the maximum bandwidth allowed via the `$dev` interface, i.e., the interface exploited by the peer connectors for offering connectivity resources.

The selection of a proper bandwidth reservation policy permits to fairly share traffic load and to reward collaborative unselfish nodes. In particular, a client  $C_i$  requesting connectivity resources receives a bandwidth  $\$bw_{C_i} = w_{C_i} * (ET * RB)$ , while the local node

achieves  $\$bw_i = ET * (1 - RB)$ , where Remote Bandwidth (RB) is the ratio between total bandwidth and bandwidth reserved for collaborating nodes and  $w_i$  is the weight related to node  $C_i$  ( $\sum_{C_i} w_{C_i} = 1$ ). The initial policy is  $RB = 0.20$  and  $w_{C_i} = 1 / \#clients$  for any node, thus reserving 80% of the estimated throughput to the local node and equally sharing the remaining 20% among the others. However, local application conditions and LS events can push to reconsider this default configuration. In particular, the user of a node can specify if she requires low, medium, or high performance for her local applications. MMHC consequently sets RB to 0.80, 0.50, and 0.20, respectively (note that the default configuration tries to maximize the satisfaction of the local user). However, RB is fixed at least to 0.50 in the case that the Starvation Monitor has notified an LS event. This is justified by our primary objective of providing a minimum quality of connectivity to any collaborating node, despite its position in the MMHC network topology. In addition, nodes notify if they behave as peer connectors and how many clients they are currently serving. MMHC sets weight values accordingly, thus fairly reserving and distributing the quota of available bandwidth. For instance, consider a deployment environment with two clients, C1 not providing connectivity and C2 providing connectivity to other two nodes. MMHC autonomously sets  $w_{C1} = 0.25$  and  $w_{C2} = 0.75$  (differently from the initial  $w_{C1} = w_{C2} = 0.50$  configuration).

## 5. RELATED WORK

Several proposals have recently investigated some specific and partial aspects of what we define here as the social connectivity sharing scenario. Some work is starting to propose the synergic and simultaneous exploitation of heterogeneous interfaces hosted at mobile terminals. Most of them have focused on one specific wireless technology, such as IEEE 802.11 or GPRS/UMTS. Their main purpose is to provide seamless connectivity in deployment environments where these technologies are integrated, possibly with different levels of dynamicity. For instance, [4] aims at extending cellular network capabilities via relay stations, with the main goal of increasing cellular coverage. [5] and [6], instead, specifically address the issue of managing client mobility among heterogeneous multi-hop networks.

By focusing on multi-hop scenarios, recent contributions about spontaneous networking have started to recognize the relevant effect of rapid throughput degradation at the increasing of the hop number. [7] aims at exploiting short and non-overloaded paths; however, it does not propose how to monitor traffic load of cooperating nodes in a lightweight and effective way. To the best of our knowledge, [8] is the only notable proposal that recognizes the importance of lightweight mechanisms to maximize reliability and practically addresses this issue by distributing context data about path robustness.

Regarding the management of fairness in connectivity resource sharing, a few contributions have recently investigated some issues related to multi-hop wireless scenarios. Some work has specifically addressed the analysis of throughput behavior in multi-hop multi-client paths. [9] provides a deep study of CSMA-based MAC protocols in terms of maximum throughput and fairness achievable. [10] primarily aims at fair bandwidth sharing by imposing a desired topology; however, the solution demonstrates to be hard to apply to dynamic environments where nodes may join/leave the network abruptly. [11] aims at providing a fair exploitation of network resources by performing load balancing of

the channels exploited by IEEE 802.11. The main goal is to maximize throughput of (distant) nodes by minimizing interferences between close single-hop links. Other proposals focus on the effective allocation of wireless medium frequencies and on the scheduling of time slots to minimize packet collisions [12, 13]. These proposals can properly work to maximize the overall throughput in the collaborative network, but do not consider traffic starvation of single collaborative nodes and, most important, their applicability is limited to homogeneous multi-frequency environments, such as all-IEEE802.11 multi-hop networks.

Finally, some solutions in the literature actively control traffic in order to maximize fairness and avoid starvation. [14] limits flows at the first hop to minimize negative impact on the remainder of the path. Its main drawback is that it is based on idealized perfect knowledge of offered load and link capabilities. On the contrary, our solution is based on practical, easy-to-gather, coarse-grained, and lightweight context indicators with regional scope. Other contributions propose a modification of layer-2 protocols, which does not fit well the highly dynamic and heterogeneous environments targeted by MMHC. For instance, both [15] and [16] adopt the Max-Min fairness definition applied to multi-hop wireless networks. The former dynamically reserves larger bandwidth for a flow only if it does not affect the channel time of other concurrent flows. Maximum fairness/performance trade-off is achieved when it is impossible to improve the quality of any flow without lowering the throughput of other flows. Also this solution requires global knowledge of network topology/state, with the associated non-negligible management overhead. [16] is based on channel time allocation too, but its approach is distributed and incremental. It defines contention regions consisting of nodes up to a distance of 2 hops. Its primary drawbacks are its approach at the MAC level and the consideration of only homogeneous multi-hop paths. [17] tries to mitigate throughput differences of long and short paths by decreasing the drop probability of packets traversing a high number of hops when traffic queues are full. To some extent it proposes a solution guideline similar to what presented in this paper, thus confirming the suitability of our approach, but it only exploits path length as the exchanged context indicator.

In short, most recent contributions aim at supporting fairness in homogeneous networks, by introducing non-standard modifications to layer-2 protocols. In addition, they do not consider high-level and expressive context indicators, such as number of served nodes and expected maximum throughput.

## 6. CONCLUSIONS AND ONGOING WORK

Our research efforts for the design/implementation of the MMHC prototype and our practical experience on its in-the-field deployment over multi-hop heterogeneous networks with off-the-shelf equipment demonstrate the feasibility of middleware solutions for self-organizing MMHC. In particular, this paper, presenting our work-in-progress in the field, highlights the relevance of stimulating spontaneous sharing of connectivity resources, by discouraging selfish behaviors. In addition, we claim that it provides some useful practical insights on how to achieve these goals with a limited overhead. In fact, to actually promote the utilization of self-organizing connectivity solutions, there is the need of effectively pushing users to behave "socially", by sharing connectivity resources and by providing their neighbors with sufficient bandwidth to get Internet connectivity satisfactorily.

Our regional fairness proposal enables the managements of rewarding for connectivity sharing in a decentralized and efficient way, by exploiting context indicators that are gathered in a lightweight manner. At the same time, our solution works to ensure a minimum connectivity quality to all collaborating nodes, in order not to harm peer connectors with excessive forwarding load.

The encouraging preliminary results achieved up to now are motivating further research work and MMHC prototype extensions. In particular, we are working on the adoption of distributed trust management models for the decentralized and reliable ranking of nodes depending on their trust level. In addition, we are extending the MMHC prototype with a continuity manager to counteract temporary connectivity interruptions during path reconfigurations for applications with soft realtime continuity requirements, such as audio-on-demand services.

## 7. REFERENCES

- [1] P. Bellavista, A. Corradi, C. Giannelli, "Context-aware Middleware for Reliable Multi-hop Multi-path Connectivity", 6th IFIP Work. on Software Technologies for Future Embedded & Ubiquitous Systems (SEUS 2008), Anacapri, Italy, Oct. 2008.
- [2] P. Bellavista, A. Corradi, C. Giannelli, "Mobility-aware Middleware for Self-Organizing Heterogeneous Networks with Multi-hop Multi-path Connectivity", IEEE Wireless Comm., Vol. 15, No. 6, pp. 22-30, Dec. 2008.
- [3] P. Bellavista, A. Corradi, C. Giannelli, "Mobility-aware Management of Internet Connectivity in Always Best Served Wireless Scenarios", Mobile Networks and Applications (ACM/MONET), Vol. 14, No. 1, pp. 18-34, Feb. 2009.
- [4] L. Le, E. Hossain, "Multihop Cellular Networks: Potential Gains, Research Challenges, and a Resource Allocation Framework", IEEE Comm. Mag., vol.45, no.9, pp.66-73, Sep. 2007.
- [5] S. Pack, X. Shen, J.W. Mark, J. Pan, "Mobility Management in Mobile Hotspots with Heterogeneous Multihop Wireless Links", IEEE Comm. Mag., vol.45, no.9, pp.106-112, Sep. 2007.
- [6] P.P. Lam, S.C. Liew, "Nested Network Mobility on the Multihop Cellular Network", IEEE Comm. Mag., vol.45, no.9, pp.100-104, Sep. 2007.
- [7] B.N. Park, W. Lee, S. Ahn, "QoS-driven Wireless Broadband Home Networking based on Multihop Wireless Mesh Networks", IEEE Trans. on Consumer Electronics, vol.52, no.4, pp.1220-1228, Nov. 2006.
- [8] R. Baumann, S. Heimlicher, B. Plattner, "Routing in Large-Scale Wireless Mesh Networks Using Temperature Fields", IEEE Network, vol.22, no.1, pp.25-31, Jan.-Feb. 2008.
- [9] M. Garetto, T. Salonidis, E.W. Knightly, "Modeling Per-Flow Throughput and Capturing Starvation in CSMA Multi-Hop Wireless Networks", IEEE/ACM Trans. on Networking, vol.16, no.4, pp.864-877, Aug. 2008.
- [10] C.P. Chan, S.C. Liew, A. Chan, "Many-to-One Throughput Capacity of IEEE 802.11 Multihop Wireless Networks", IEEE Trans. on Mobile Computing, vol.8, no.4, pp.514-527, April 2009.
- [11] A. Raniwala, Tzi-cker Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network", 24th Annual Joint Conf. of the IEEE Computer and Communications Societies (INFOCOM 2005), vol.3, pp. 2223-2234. 3, March 2005.
- [12] L. Badia, A. Erta, L. Lenzini, M. Zorzi, "A General Interference-Aware Framework for Joint Routing and Link Scheduling in Wireless Mesh Networks", IEEE Network, vol.22, no.1, pp.32-38, Jan.-Feb. 2008.
- [13] H.T. Cheng, W. Zhuang, "Joint Power-Frequency-Time Resource Allocation in Clustered Wireless Mesh Networks", IEEE Network, vol.22, no.1, pp.45-51, Jan.-Feb. 2008.
- [14] V. Gamberoza, B. Sadeghi, E.W. Knightly, "End-to-end performance and fairness in multihop wireless backhaul networks", 10th Annual Int. Conf. on Mobile Computing and Networking (MobiCom '04), pp. 287-301, Oct, 2004.
- [15] Ping Wang, Hai Jiang, Weihua Zhuang, H.V. Poor, "Redefinition of max-min fairness in multi-hop wireless networks", IEEE Trans. on Wireless Communications, vol.7, no.12, pp.4786-4791, Dec. 2008.
- [16] A. Rao, I. Stoica, "Adaptive Distributed Time-Slot Based Scheduling for Fairness in Multi-Hop Wireless Networks", 28th Int. Conf. on Distributed Computing Systems (ICDCS '08), pp.874-882, June 2008.
- [17] Chaegwon Lim, Chong-Ho Choi, Hyuk Lim, "A weighted RED for alleviating starvation problem in wireless mesh networks", 33rd IEEE Confe. on Local Computer Networks (LCN 2008), pp.841-842, Oct. 2008.