

# Point to Multipoint Transport in Multichannel Wireless Environments

Hicham Khalifé, Vania Conan, Jérémie Leguay  
Thales Communications & Security - France  
Email: {name.surname}@thalesgroup.com

Thrasylvoulos Spyropoulos  
EURECOM - France  
Thrasylvoulos.Spyropoulos@eurecom.fr

**Abstract**—We propose a transport protocol capable of dynamically adapting to network and receiver properties in multi-destination, multi-channel wireless networks. The key feature of our solution resides in its ability to convey common traffic to a group of users, while at the same time distributing information to each user as quickly as possible. This is achieved by clustering receivers in groups, each group being served at a suitable throughput. We emphasize in this study on the two groups of receivers case. We show analytically and through OMNet++ simulations that groups formation is decided by the wireless link performance and the proportion of receivers constituting each group. Our solution captures dynamically these effects. Indeed, our transport is capable to cope transparently with wireless links changes (i.e spectrum handoff) by adapting dynamically its transmission rate and groups composition. It is therefore adapted for point-to-multipoint cognitive radio networks.

## I. INTRODUCTION

Point-to-multipoint communication schemes were proposed to convey information of common interest to a group of users. Initially, this communication model addressed the applications' need to distribute content, such as video streams, to multiple receivers. It motivated the birth of group addressing paradigms like multicast, thus creating sets of receivers sharing the same interest. Since then, a wide range of novel applications tend to distribute information with multiple users. These users may share socially a common interest or be located in geographically close positions. Disseminating alerts, videos and pictures to a set of workers, in public safety networks, are known examples.

More recently, off-the-shelf wireless communication devices have become capable of exploiting multiple frequency bands or channels [1]. This technology improves spatial reuse and increases the observed throughput. However, in a point-to-multipoint context, users can be spread over different frequencies, channels, or locations. Thus, the links “connecting” each destination to the source, might have inherently different characteristics (e.g. bandwidth, center frequency and thus propagation properties, etc) or be interfered by different (primary or secondary) users (in a cognitive radio context). As a result, users in the same group may experience very heterogeneous performance in terms of latency, physical transmission rate, MAC layer retransmissions, etc. Beyond wireless access networks, this situation occurs in wireless ad hoc

networks where receivers are possibly connected over multiple hops.

Recently proposed transport protocols for point to point cognitive radio networks [2], [3], [4] do not address challenges of the point-to-multipoint communication scheme. In fact, these solutions make the comprehensive assumption that at time  $t$  a single destination needs to be reached. Hence rate adaptation is based on optimizing the transmission parameters based on this destination reception capabilities. Alternatively, present point-to-multipoint transport solutions do not cope well with the new conditions created by multi-channel environments. Standard multicast solutions target essentially multicast sessions with large groups [5], [6], [7]. For receivers with different flow rates, one can compute multicast groups based on throughput [8] or create layered multicast protocols [9], [10]. They apply well to layered content/stream distribution, where each quality layer is mapped to the corresponding receiver rate.

To support point-to-multipoint distribution of the same data across heterogeneous receivers, the source can adapt its flow to the slowest receivers, like in the NORM [11] protocol or in RTMP [7]. This however translates in pulling down the reception rate of all nodes; while this might be appropriate in the Internet where receivers have close behaviors, it is not the case in a multi-channel wireless context. Alternatively the source could follow the fastest receivers, but it is taking the risk of “losing” the slowest, resulting in too many packets being dropped on the saturated slow channels.

To solve this dilemma, we propose PMT (Point-to-Multipoint Transport), an acknowledgement based transport protocol which dynamically differentiates among receivers and separates them according to their reception capabilities. PMT creates dynamic groups of receivers managed by the source to improve delivery time for the nodes that can receive data early, and thus the overall throughput. In general PMT would cluster the receivers dynamically in an arbitrary number of groups. In this paper we thoroughly study the case where receivers fall naturally into two groups.

To compute the two groups of fast and slow receivers we introduce a dynamic algorithm based on the observed Round Trip Time (RTT) of receivers which maximizes the average throughput of the system. We also provide a simplified analytical model of the mechanism to study its dependency on RTT distributions, and on the respective proportion of fast

This work is supported by the French National Research agency (ANR) under grant reference LICoRNe ANR-10-VERS-005

and slow nodes. We verify that the behavior of the greedy algorithm is aligned with the simplified model on truncated Gaussian RTT distributions.

Finally we evaluate PMT in a wireless multichannel cognitive radio context through OMNet++ simulations. Our evaluations highlight the fact that PMT can cope dynamically with link property variations thus adapting dynamically the group members as well as the transmission rate of every group.

The remainder of the paper is structured as follows. Section II describes the protocol and its implemented mechanisms. Section III provides an analysis of the behavior of the grouping principle. We simulate and analyze the performance of our proposal in Section IV. Section V concludes this paper and discusses future work.

## II. TRANSPORT PROTOCOL

### A. Preliminaries

In a multideestination configuration, the throughput of a group composed of  $N$  members can be expressed as the sum of the throughput of all the members of the group.

$$\Phi_{group} = \sum_{i=0}^N \Phi_i$$

where  $\Phi_i$  is the throughput observed by the member  $i$  of the group.

In order to prevent slow receivers from penalizing those benefiting from favorable network conditions, we seek to create dynamically separate groups each served at a particular throughput. We base the group formation algorithm on the round trip time (RTT) observed by each node. With these observations the source node is able to differentiate between slow and fast nodes. A single time threshold  $T$  is sufficient to discriminate between both groups: all nodes below the threshold  $T$  go in the fast group, all nodes above in the slow group (their RTT is larger).

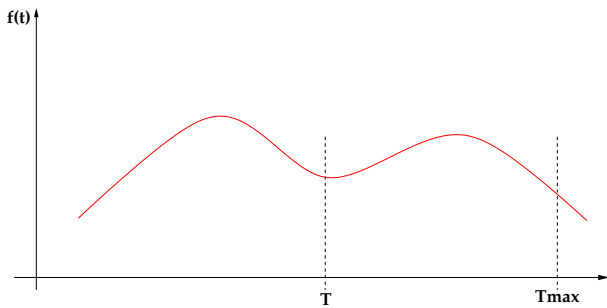


Fig. 1. Distribution of RTTs observed by the source node exemplifying a mix of fast and slow receivers separated by threshold  $T$

Let us consider the distribution of RTTs from all  $N$  nodes observed by the source node (refer to Figure 1 for an example). The source sends a message at  $t = 0$  and waits for an acknowledgement (ACK) from the receivers. Let  $X$  denote the arrival time at the source of the ACK messages.  $X$  is a random variable of message RTTs. Let us denote as  $T$  the threshold that separates the fast group from the slow one.

In PMT, nodes who respond to the first packet before  $T$ , are placed in the fast group and the next packet for them is sent at  $T$ . Nodes who respond after  $T$ , will be placed in the slow group, and the next packet for this group will be sent only after an expiration timer  $T_{max}$ .

Therefore, if  $f(t)$  denotes the probability density function (p.d.f) of  $X$  and  $F(t)$  its cumulative distribution function (c.d.f), the average throughput per node,  $\Phi$ , can be written as follows:

$$\begin{aligned} \Phi &= \frac{F(T)}{T} + \frac{\int_T^{T_{max}} f(t) dt}{T_{max}} \\ &= \frac{F(T)}{T} + \frac{F(T_{max}) - F(T)}{T_{max}} \end{aligned} \quad (1)$$

$T_{max}$  is often a fixed protocol parameter, dictated by the application requirements, or the need to ensure quite slow nodes can still get enough packets. However, in some settings, one could also optimize  $T$  and  $T_{max}$  together. Furthermore, one could possibly consider more than two groups (with a complexity-performance tradeoff at hand). In order to better illustrate our approach, we choose to handle here the simple case of two groups and fixed (large)  $T_{max}$ , and defer the more general cases to future work.

In the considered setting, the tradeoff is the following: one could try to maximize the number of receivers in the first group. However, this would imply increasing  $T$  and thus reducing throughput for all nodes in the first group. It is intuitive that an optimal value of  $T$  should exist, depending on the RTT probability distribution. We formally analyze this optimal value in Section III-A.

### B. Protocol description

Our mechanism is source driven, in other words the source node maintains, in a special database, the group affiliation for every receiver. The average RTT for every receiver (other participants within the zone) is also stored inside this database. Moreover two transmission buffers are added, each handling transmissions for a precise group. The protocol building blocks are shown in Figure 2.

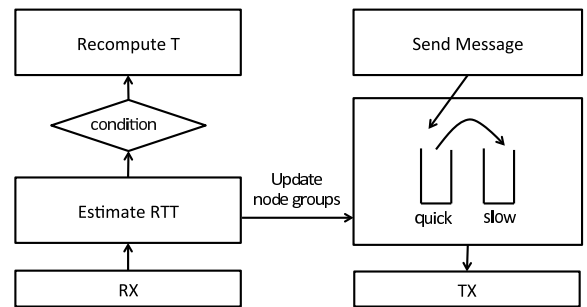


Fig. 2. Protocol building blocks

The source sends a message for the fast group every  $T$  and serves the slow receivers every  $T_{max}$ . In fact, the source

transmits the message available in the fast nodes queue to the fast receivers and waits for the acknowledgements. After  $T$  seconds (i.e at the expiry of the fast nodes interval), receivers that have answered are labeled as fast; all others are labeled as slow in the specific database. Using timestamps, the smoothed RTT of fast receivers is also updated. The message is then transferred to the slow group buffer and transmitted to the slow receivers at  $T_{max}$ . This flow control process is repeated whenever new messages are available for transmission. More generally, the throughput of our protocol is dictated by  $T$  and  $T_{max}$  as follows:

- at  $T$  source pushes the packet to the slow group queue, pops a new packet and sends it to the fast group receivers.
- at  $T_{max}$  sources removes from the slow queue the packet sent  $T_{max}$  seconds earlier, then transmits the packet in head of queue to the slow group members.

Note that the purpose of the protocol is to improve delivery time for the nodes that can receive data early. The long-term throughput of the system is unchanged, as it is dictated by the second queue (the slow nodes that are served every  $T_{max}$ ), since all nodes receive the same data.

### C. Algorithm for dynamic group calculation

In order to select the appropriate value of  $T$  that separates the fast from slow nodes, we propose a greedy algorithm which maximizes the average throughput per node (which is the same as maximizing total network throughput for a fixed number of receivers).

---

#### Algorithm 1 Estimate optimal value of $T$

---

**Input:**  $N$  //total number of receivers

$\tau[N]$  //table containing smoothed RTT of every receiver  
 $T_{max}$

$max = 0$ ,  $index$ ,  $result$ ,  $j$  //intermediate variables

**Output:**  $T$

```

1: sort( $\tau[N]$ )
2: while  $j < N$  do
3:    $result \leftarrow (j \cdot \frac{1}{\tau[j]}) + (N - j) \cdot \frac{1}{T_{max}}$ 
4:   if  $result > max$  then
5:      $max \leftarrow result$ 
6:      $index \leftarrow j$ 
7:   end if
8:    $j = j + 1$ 
9: end while
10:  $T \leftarrow \tau[index] + \epsilon$ 
11: return  $T$ 

```

---

The basic idea of the algorithm is to determine the value of  $T$  by computing the average throughput based on the receivers' RTTs. First, we start by sorting received RTTs in increasing order (line 1). Then, by sequentially selecting the RTT of receiver  $j$  and computing throughput of each group accordingly (i.e by also including all receivers having smaller RTT) we estimate the throughput as if the RTT of receiver  $j$  equals the value of  $T$  (line 3 of the algorithm). At the end

of this loop the algorithm returns the RTT value that offers the highest total throughput. In practice, a slightly bigger ( $+\epsilon$ ) value from this RTT is selected for  $T$  in order to maximize the total network throughput. In fact, this small margin allows to account for potential RTT fluctuations.

## III. PROTOCOL ANALYSIS

In this section we analyze the behavior of the mechanism that we propose. For this purpose we describe a simplified and tractable model for the computation of the groups and we investigate its properties. We show that the mechanism applies to a large number of RTT distributions. We conclude that it is stable in the way it separates the two groups of RTTs when they occur naturally. We also show that there is a transition between two states depending on the relative number of fast and slow receivers. We also verify that this general behavior translates to the algorithm of Section II.

### A. Maximizing average throughput

We first consider a group of nodes whose RTTs follow a unimodal distribution. Formally, let  $X_a$  denote the RTT distribution of a subset of receivers, whose probability distribution is defined by function  $f_a$ . We further suppose that  $f_a$  is symmetric around its mean value  $a$ , with bounded support of length  $l$  ( $l < a$ ), null outside  $[a - l, a + l]$ , increasing on  $[a - l, a]$  and continuous derivable on  $]a - l, a + l[$ .

**LEMMA.** The above hypotheses on the distribution  $f_a$  of RTTs ensures that the throughput  $\phi(T)$  has a unique maximum, which we note  $t_a$ . In other words:

$$t_a = \max_{T>0} \phi(T) = \max_{T>0} \frac{\int_0^T f_a(t) dt}{T} \quad (2)$$

**PROOF.** for  $t > 0$ ,  $\phi$  is  $C^1$ , and

$$\phi'(T) = \frac{t f_a(t) - \int_0^T f_a(t) dt}{t^2} \quad (3)$$

The sign of  $\phi'$  is given by its numerator  $\xi(t) = t f_a(t) - \int_0^t f_a(t) dt$ , which one can differentiate again yielding  $\xi'(t) = f_a'(t)$ . The shape of  $f_a'$  gives the sign of  $\xi'$ . Noting that  $\xi(a) \geq 0$  and  $\xi'(a + l) = -1$  yield the existence and uniqueness of  $t_a$ . Furthermore one has  $t_a \geq a$  and  $t_a$  verifies  $t_a f_a(t_a) = \int_0^{t_a} f_a(t) dt$  since  $\phi'(t_a) = 0$ .  $\square$

Now let us turn to the case where the receivers' RTTs fall naturally into two groups. This occurs for example when the two groups of receivers are on two different channels with significantly different propagation properties (delay, bandwidth, hop count, ...). The fast group responds with average RTT  $a$ , support length of  $2l$ , distribution  $f_a$ , contains a proportion  $0 < \alpha < 1$  of receivers, and the slow one with average RTT  $b$ , support length of  $2m$  and distribution  $f_b$ , contains a proportion  $1 - \alpha$  of receivers.

When the two groups are clearly separated (i.e., if  $b > a + l + m$ ) one can apply the above lemma twice yielding the following result.

**COROLLARY.** For a mixed distribution of RTTs that falls into two groups (as shown in Figure 3), one given by  $X_a$  with

average RTT  $a$ , support length of  $2l$ ,  $l < a$ , the other given by  $X_b$  with average RTT  $b$ , support length of  $2m$ ,  $m < b$ , and for  $b > a + l + m$ , throughput  $\phi(T)$  is achieved by only two possible values of  $T = t_a$  or  $T = t_b$ ,  $t_a < t_b$  which verify  $t_a f_a(t_a) = \int_0^{t_a} f_a(t) dt$ , and  $t_b f_b(t_b) = \int_0^{t_b} f_b(t) dt$ . Whenever the proportion  $\alpha$  is below a threshold  $\alpha_{th}$ ,  $T = t_b$ , otherwise  $T = t_a$ .  $\alpha_{th}$  is given by:

$$\alpha_{th} = \frac{\int_0^{t_b} f_b(t) dt}{\int_0^{t_a} f_a(t) dt} \quad (4)$$

In other words, when there is a small proportion of fast receivers, all receivers are gathered in a single group (i.e.,  $T = t_b$ ); when there is a sufficient proportion of fast receivers, only the fast receivers are served (i.e.,  $T = t_a$ ) to maximize average throughput.

One can apply this result to uniform RTT distributions; the two values that separate the groups are  $t_a = a + l$  and  $t_b = b + m$ , and the transition occurs for a proportion of fast receivers  $\alpha_{th} = \frac{a+l}{b+m}$ .

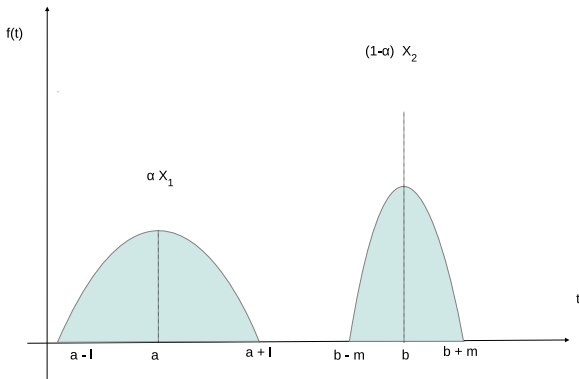


Fig. 3. Example of a mixed distribution  $X = \alpha X_1 + (1 - \alpha) X_2$  of fast ( $X_1$ ) and slow ( $X_2$ ) nodes

### B. Numerical validation

In this section, we first confront the theoretical model above to the results provided by the on line algorithm. Throughout our validation, we choose the normal distribution  $N(\tau, \sigma^2)$  for the acknowledgement arrivals at the source. More precisely, we generate arrivals at the source following two separate normal distributions having each a different mean ( $\tau$ ). We modify the group size ( $\alpha$ ) and characterize the behavior of the value of  $T$ . In fact, a normal law maps well distributions of receivers using the same channel. This can be seen as a high number of receivers acknowledging around a mean value.

In Figure 4, we vary in our on line algorithm the proportion of receivers in each group and investigate its impact on  $T$ , the threshold separating the two groups for optimal throughput. In these numerical validations we consider a network of 80 receivers (that we modify  $\alpha$ , their proportion in each group), acknowledgements arriving at a  $\tau_1=10$  and  $\tau_2=50$  seconds

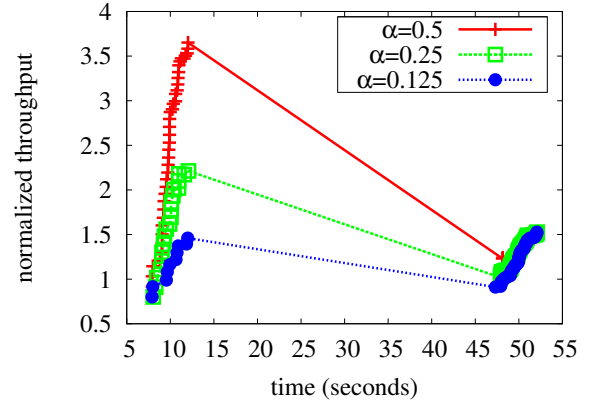


Fig. 4. Maximum overall network throughput for various values of  $\alpha$ , proportion of fast receivers

for group 1 and group 2 respectively, both with a standard deviation of 1, and we set  $T_{max}$  to 80 seconds.

In fact, Figure 4 shows that when the number of receivers in the first group is low in proportion to the second group members ( $\alpha = 0.125$ ), the best performance is obtained when a single group is defined having a throughput determined by the largest RTT. Intuitively, this can be explained by the fact that the high number of receivers in group 2 makes the total throughput less dependent on the contributions of the receivers having small RTT.

More formally, in such configurations, as the theoretical study in Section III-A shows, the number of receivers in each group plays an important role in defining the value that maximizes the network throughput. In this specific case Eq. (2) gives  $\int_0^{t_a} f_a(t) dt = 3.92$ , and  $\int_0^{t_b} f_b(t) dt = 19.89$ . Applying Eq. (4) provides a threshold value  $\alpha_{th} = 0.197$ . This matches the results in Figure 4 where one can observe that the shift indeed takes place between  $\alpha = 0.125$  and  $\alpha = 0.25$ .

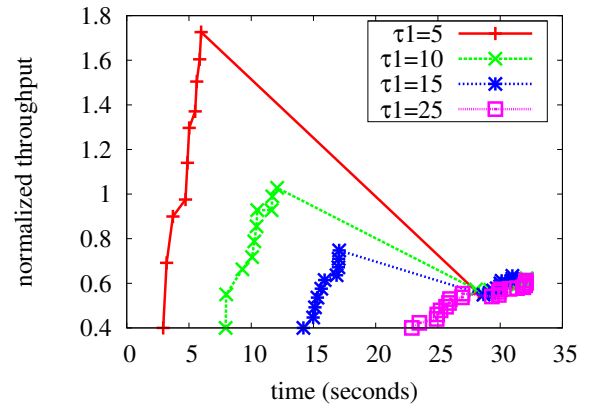


Fig. 5. Maximum total network throughput for several means of the arrival time of the fast group. This maximum is used to position  $T$

We further study what happens when, instead of two clearly separated groups the gap between the mean RTT reduces. To do so, we vary the mean arrival time of the normal distribution



of the first group  $\tau_1$  (with values 5, 10, 15, and 25) while keeping the mean of the second group at  $\tau_2 = 30$ . In these experiments, we consider a network consisting of a source and 20 receivers evenly distributed between the two groups. The value of  $T_{max}$  is fixed at 50 seconds.

Our results displayed in Figure 5 show that when the two groups are clearly separated (i.e the mean RTT are not close), maximizing the network throughput is achieved by selecting a  $T$  value equal to the RTT of the slowest receiver in the fast group. Practically, if the two sets are clearly disjoint the best performance is obtained when two groups are created: a) the fast served at the throughput of a packet every larger RTT in the group and b) the slow group served a new packet every  $T_{max}$ . It is also interesting to notice that when the gap between groups becomes smaller, the value of  $T$  that maximizes the network throughput is the longest observed RTT. In other words, in these cases the interval separating the arrival of new acknowledgements is reduced in a way that waiting for new ACKs and serving all receivers at the same rate becomes more beneficial than creating new groups with different throughputs.

These two results are of a high interest for our transport protocol. Indeed, in extreme cases where the groups become less disjoint or asymmetric in terms of receivers the best performance is obtained with a single group as in traditional point to multipoint solutions. Practically, the capacity of our solution to recompute  $T$  dynamically allows moving from two groups to a single one (and vice versa) thus reducing the system complexity and overhead.

#### IV. SIMULATION IN A MULTICHANNEL WIRELESS ENVIRONMENT

In order to validate the behavior of PMT in realistic multichannel conditions, we present a set of simulations realized with the OMNet++ framework. We have extended the MiXiM module [12] to simulate two separate channels each operating at a different frequency band with a different theoretical capacity. More practically, we have configured our simulator with two bands each having a tunable capacity and center frequency.

##### A. Simulation scenario

Throughout our simulations, a dual band sender transmits the same information to destinations spread over two different frequencies. Receivers can exclusively exploit a single band to receive packets and transmit their acknowledgements. They can communicate either over frequency 1 or frequency 2 thus experiencing different throughputs and delays. We evaluate our PMT protocol which automatically differentiates receivers into two groups and derives the transmission rate for each one. More practically, we have implemented our transport protocol in the OMNet++ framework. In our simulations, a source node using PMT embeds the protocol building blocks defined in Figure 2. It is thus capable to calculate the RTT of its receivers and adapt the transmission rate accordingly.

We consider the representative scenario (Figure 6) to characterize PMT performance in a real deployment. In this topology

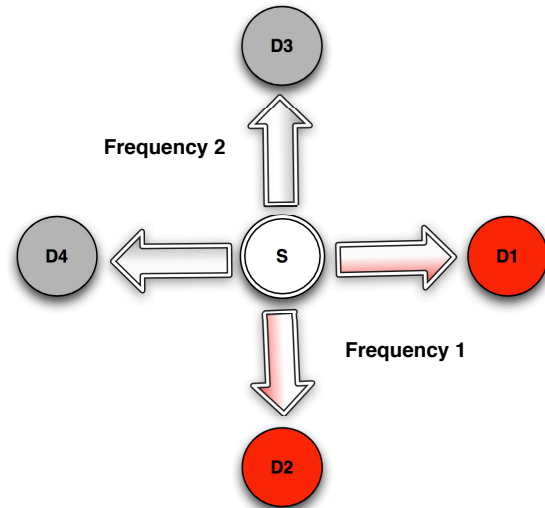


Fig. 6. Simulated topology

a source dual radio radio S transmits the same information to 4 intended destinations. Nodes D1 and D2 exploit frequency 1 (f1) to communicate whereas D3 and D4 can communicate exclusively over frequency 2 (f2). Such scenarios are typical in point to multipoint multichannel environments where each receiver selects the appropriate channel for communication based on the interference observed locally. Clearly, these different bands possess different properties and performance.

In our simulations, we modified the capacity offered by f2 (while changing also its operating center frequency) and observed the variations of RTT values as well as the computed value of  $T$ . We transmit packets of 1000 bits to the four destinations that acknowledge every 5 packets. The ACK message is then used to compute the RTT and update the group membership of sending node. The RTTs are kept in the source database then used to compute a new value of  $T$  when the 4 destinations acknowledge the message. Simulations of 100 seconds are run however we show here the values of  $T$  and RTT until these values converge. Note that we bootstrap our simulations with a relatively big value of  $T$  that equals 0.05 seconds.

##### B. Simulation results

Figure 7 shows the observed RTT for f1 and f2 as well as the RTT when an important gap exists between the capacities on each link. In this simulations on f1 (called fast RTT in the figure) a capacity of 2 Mbits/s was available whereas channel 2 operates at 100 Kbits/s. Our simulations show that in such configurations, 2 groups are created: (a) the fast group served by a packet every  $T$  seconds and (b) the slow group that is served only at  $T_{max}$ . Therefore PMT protocol is able to serve fast receivers at nearly the maximum possible throughput. In known existing solutions a.k.a the NORM protocol, all receivers would have been served at the throughput allowed by the slowest receiver (i.e the slow group). Note here that the

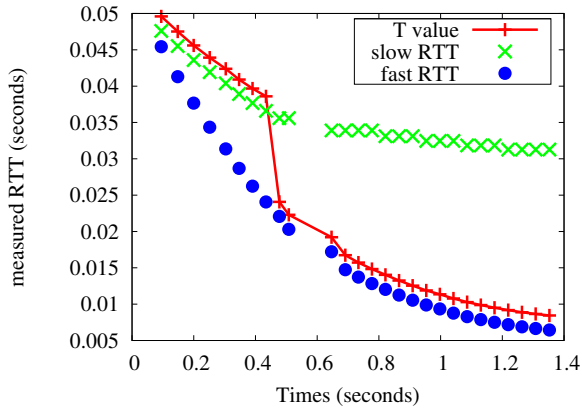


Fig. 7. High wireless link capacities differences

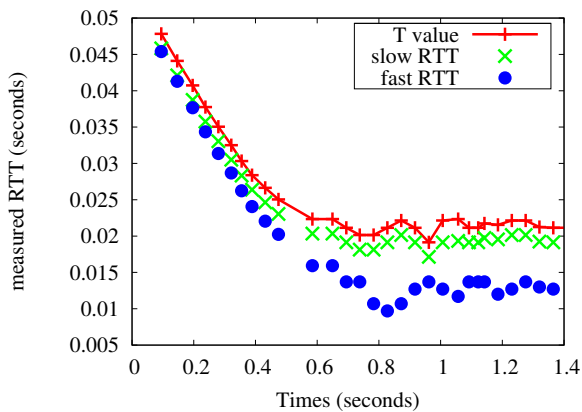


Fig. 8. Observed RTT when link capacities become closer

slow convergence at the system bootstrap is due to the fact that measured RTT by the source are smoothed with an EWMA technique in order avoid oscillations. Most importantly, this convergence shows that PMT is able to adapt dynamically the transmission rate of every group without any manual configuration.

In Figure 8, we increase the capacity of frequency 2 to 500kbts/s. In these simulations, the gap between the RTT of the fast and slow group is reduced. Consequently, the value of  $T$  that optimizes the network throughput falls at the end of the second group. This behavior was also observed theoretically in Figure 5. Indeed, waiting for new ACKs in this scenario and serving all receivers at the same rate is more beneficial than creating a separate group with a different throughput. Finally, it is worth noting here that the throughput of the fast group decreases (the RTT increases) when frequency 2 capacity is increased in Figure 8. This is practically due to the fact that the dual band sender is equipped with a single transceiver in our simulations. Therefore a transmission on frequency 2 impacts the actions of the other band. This results in less capacity on f1 with the rate increase of f2. Nevertheless, PMT being robust to the physical layer variations, copes transparently with such situations.

## V. CONCLUSION AND FUTURE WORK

In this paper we have presented an acknowledgement based transport protocol for point-to-multipoint multi-channel networks which splits receivers into groups, each served at a suitable throughput. The major advantage of our solution consists in preventing slow receivers from affecting the service offered to receivers possessing better conditions (fast group). Using a mathematical formulation we have computed the optimal group forming policy and exploited it in our solution. Our validations prove that our algorithm is able to dynamically estimate the optimal sending rate for each created group. We have further shown that when the fast group has a small proportion of receivers the best performance in terms of overall throughput is obtained by creating a single group. It is also the case when the mean RTTs of the two groups are close.

Most importantly, our solution manages groups dynamically by recomputing transparently to receivers the optimal transmission rate of every group while maximizing overall network throughput. Our transport protocol can be particularly useful in the context of cognitive radio networks since group forming and transmission rate computation can smoothly adapt to opportunistic channel selection when serving each receiver.

In the future we plan to extend the solution to  $N$  groups. Intuitively, this can be seen as running the same algorithm recursively on the created groups. However, optimality of this solution should be verified in terms of obtained throughput for every group. Moreover we plan to include additional parameters when computing group sizes. For instance, it would be interesting to include the generated overhead in terms of duplicated transmission in our computation.

## REFERENCES

- [1] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey," *Comput. Netw.*, vol. 50, no. 13, pp. 2127–2159, Sep. 2006.
- [2] K. Chowdhury, M. D. Felice, and I. Akyildiz, "Tp-crahn: A transport protocol for cognitive radio ad-hoc networks," in *IEEE infocom conf*, 2009.
- [3] D. Sarkar and H. Narayan, "Transport layer protocols for cognitive networks," in *IEEE infocom workshops*, 2010.
- [4] C. Luo, F. R. Yu, H. Ji, and V. C. M. Leung, "Optimal channel access for tcp performance improvement in cognitive radio networks," *Springer Wireless Networks*, vol. 17, no. 2, 2011.
- [5] G.Kwon and J.Byers, "roma: a reliable overlay multicast with loosely coupled tcp connections," in *IEEE infocom conf*, 2004.
- [6] J.Baek and J.Paris, "A tree-based reliable multicast scheme exploiting the temporal locality of transmission errors," in *IEEE IPCCC conf*, 2005.
- [7] S. Paul, K. K. Sabnani, J. C.-H. Lin, and S. Bhattacharyya, "Reliable multicast transport protocol (rmtp)," *IEEE Journal of Selected Areas in Communication (JSAC)*, vol. 15, no. 3, 1997.
- [8] L.Rizzo, "pgmcc: a tcp-friendly single-rate multicast congestion control scheme," in *ACM SIGCOMM conf*, 2000.
- [9] G.Kwon and J.Byers, "Smooth multirate multicast congestion control," in *IEEE infocom conf*, 2003.
- [10] J.Byers, G.Horn, M.Luby, M.Mitzenmacher, and W.Shaver, "Fliddl: Congestion control for layered multicast," *IEEE Journal of Selected Areas in Communication (JSAC)*, vol. 20, no. 8, 2002.
- [11] B.Adamson, C.Bormann, M. Handley, and M. Handley, "Nack-oriented reliable multicast (norm) transport protocol," *RFC 5740.*, 2009.
- [12] "The mixim (mixed simulator) project." [Online]. Available: <http://mixim.sourceforge.net>