Subset-sum Based Relay Selection for Multipath TCP in Cooperative LTE Networks

Dizhi Zhou, Wei Song, *Member, IEEE*, and Peijian Ju

Faculty of Computer Science
University of New Brunswick, Fredericton, Canada
Emails: {q5frc, wsong, i1689}@unb.ca

***Abstract*—Pooling mobile devices in vicinity as a cooperative community offers an opportunity to enable multipath transmission for multi-homed mobile devices, even when there is no multiple access coverage. Nonetheless, the available bandwidth provided by relays can be highly varying due to a range of factors such as wireless channel fading and dynamic local traffic load at relays. As a result, it is challenging to maintain a stable multipath aggregate throughput over relays. In this paper, we propose an enhancement module within the application layer for a cooperative community in the Long Term Evolution (LTE) network. Our extension is based on the standardized multipath transport control protocol (MPTCP) [1]. Based on relay bandwidth monitoring, a dynamic relay selection algorithm is developed for adding and deleting paths so as to ensure a stable aggregate throughput in a highly varying environment. The proposed relay selection algorithm is based on a fully polynomial-time subset-sum approximation [2]. Extensive simulations are conducted to evaluate the proposed solution in different background traffic patterns. The simulation results well demonstrate the strengths in minimizing throughput outage, the number of active subflows, and performance variation.**

***Index Terms*—MPTCP, LTE, relay selection, subset sum, cooperative wireless networks.**

I. Introduction

Most mainstream mobile devices are now equipped with multiple radio interfaces. The multi-homed capability of multi-radio devices offers a good opportunity to explore multiple interfaces for multipath transmission. Multipath transport control protocol (MPTCP) [1] is standardized by Internet Engineering Task Force (IETF) in 2011. MPTCP runs in multi-homed mobile devices to simultaneously deliver transport control protocol (TCP) packets over multiple interfaces. However, the Wi-Fi link as a potential transmission path is not ubiquitously available. When there is no Wi-Fi coverage (e.g., within subways), only the wireless wide area network (WWAN) interface, such as Long Term Evolution (LTE), is active for wireless access. Hence, MPTCP may not be always feasible even if the mobile device is equipped with multiple interfaces.

Pooling mobile devices in vicinity together as a cooperative community [3] is an alternative way to enable multipath transmission for multi-homed mobile devices even when there is no multiple access coverage. Fig. 1 shows a cooperative community associated with an Evolved Node B (eNB) in the LTE network, in which one user equipment (UE), referred to as *root UE*, can aggregate the available bandwidths of two nearby user equipments, referred to as *relay UEs*, by utilizing short-distance communications (e.g., Wi-Fi). The relay UEs can receive packets on behalf of the root UE via their own LTE interfaces and then forward packets toward the root UE via Wi-Fi. By this means, the root UE can run multiple paths with MPTCP with each path corresponding to a relay UE.

This research was supported by research grants from Natural Sciences and Engineering Research Council (NSERC) of Canada.

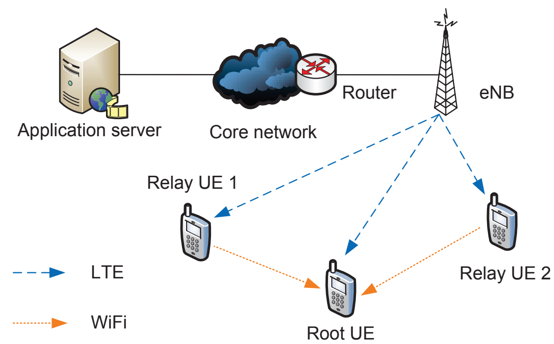

Fig. 1. Cooperative community in the LTE network.

However, the available bandwidth provided by a relay UE is highly dynamic due to the fading effect of the wireless channel and varying local traffic load of the relay UE, which may in turn introduce serious side effects to MPTCP [4]. Here, we use the term, *available bandwidth*, to refer to the bandwidth that a relay UE provides to the root UE. Then, there comes up an essential problem that, given distinct and varying available bandwidths of relay UEs, how MPTCP guarantees a stable aggregate throughput to the application layer of the root UE. A possible approach is to dynamically select relay UEs so that the desired aggregate throughput defined in terms of *target bit rate (TBR)* is achieved at the root UE.

Relay selection is one of key issues for cooperative wireless networks and many solutions have been proposed at different layers [5]. In [6], Fu *et.al.* propose a physical-layer network coding approach to improve the overall performance by having multiple transmitters send data to the same receiver. However, this solution focuses on the physical layer and does not address the end-to-end cooperative performance. In [7], Zhu *et.al.* propose an energy-efficient relay selection for TCP, which still focuses on single-path TCP and cannot be used in the multiple relay scenario. The current MPTCP standard [8] defines a priority option to indicate whether a path is set as a backup or not. However, it does not provide any specific algorithm to select backup paths and to map backup paths to active paths.

In this paper, we propose a subset-sum based relay selection algorithm for MPTCP running within a cooperative community in the LTE network, as shown in Fig. 1. The objective is to guarantee a stable aggregate throughput that satisfies the application-layer TBR requirement of the root UE. The key idea is to have the root UE maintain multiple relay sets whose total available bandwidths are within an acceptable TBR range (e.g., between 90% and 110% of TBR). Once the total available bandwidth is found out of this range (referred to as an *outage*), the root UE updates the current in-use relay set, referred to as *active set*, to a new set, referred to as *backup set*. The selection of backup sets is based on several criteria, including the total available bandwidth, the number of relay UEs (i.e., the number of subflows), and the required path migration operations (i.e., adding and deleting MPTCP paths). The corresponding general problem is the well-known subset-sum problem, which is NP-complete. Extending an existing fully polynomial-time approximation algorithm, we propose a dynamic subset-sum based relay selection scheme. The scheme is integrated into the application-layer structure of the root UE, so that the relay paths are adapted dynamically and migrated smoothly. Simulations are conducted with `ns-3` [9] for a complete LTE and MPTCP environment. The simulation results well demonstrate the strengths of the proposed solution with respects to minimizing TBR outage, the number of subflows, and performance variation.

The remainder of this paper is organized as follows. In Section II, we first briefly introduce the background of MPTCP, and then the system framework and proposed relay selection algorithm. Simulation results are presented in Section III to demonstrate the achieved MPTCP performance with varying background traffic load at relay UEs. Related work is reviewed in Section IV, followed by conclusions in Section V.

II. Dynamic Relay Selection Solution

A. MPTCP background

MPTCP loosely splits the transport layer into two sublayers, namely, MPTCP and subflow TCP. Based on this architecture, MPTCP can be easily implemented within current network stack. Specifically, subflow TCP runs on each path independently and reuses most functions of regular TCP. On the other hand, MPTCP sublayer is responsible for packet scheduling which defines the policy of forwarding packets to each subflow at the source, congestion control which aims to balance the traffic load among subflows and ensure TCP-friendliness, as well as path management that discovers, adds and deletes subflows for the multipath connection between two hosts.

B. System framework

Fig. 2 shows the system framework at the root UE and relay UEs. The rectangular shapes show components of existing software stack, while the rounded rectangular shapes

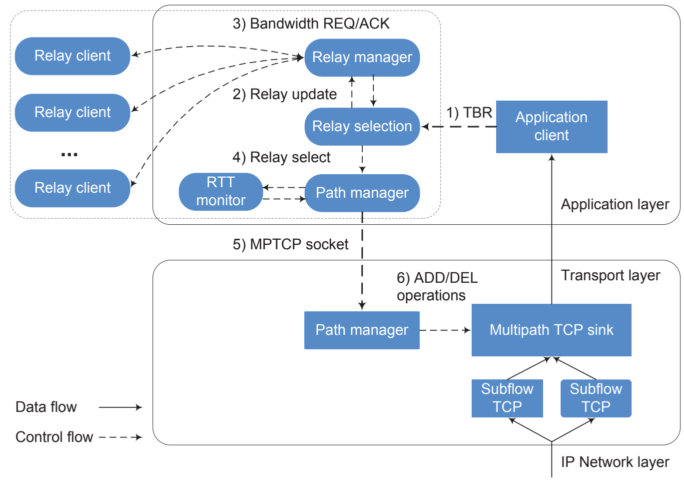

Fig. 2. System framework.

in the dashed frame represent the proposed enhancement module. The proposed module has two interfaces with the standard stack. Firstly, it allows the application to provide a desired TBR requirement. Secondly, the proposed module uses standard MPTCP socket APIs to enable multipath transmission [10]. In particular, `TCP_MULTIPATH_ADD` and `TCP_MULTIPATH_REMOVE` are used to add and delete subflow paths. As such, an effective relay selection scheme can be easily deployed in MPTCP-enabled mobile devices without modifying the MPTCP and application protocol stacks.

Based on the framework in Fig. 2, we have the following path management procedure. At the beginning of an application session, the relay selection module acquires the desired TBR from the application client, and the relay manager retrieves the available bandwidths of relay UEs from their periodical broadcasts. The available bandwidth is provided by each relay client by estimating its average local traffic throughput. Once the relay manager obtains the available bandwidths of relay UEs, it can determine all feasible relay sets whose total available bandwidths are within the acceptable TBR range (e.g., between 90% and 110% of TBR). Then, a best relay set can be selected based on certain criteria and configured as the active set. The algorithm of determining feasible relay sets and selecting the best relay set will be discussed in the next subsection. Finally, the active set is forwarded to the path manager at the application layer, which further calls MPTCP socket APIs to add all subflows to MPTCP.

During the application session, relay UEs periodically update the relay manager on the available bandwidths. Accordingly, the relay selection module regenerates feasible relay sets whose total available bandwidths satisfy the acceptable TBR range. A new best relay set is then selected as the backup set for the current active set. Once the total available bandwidth of the active set is found out of the TBR range, the path manager is triggered to migrate the current active set to this backup set. The detailed algorithm of selecting the backup set is given in the next subsection. The path manager compares the active set and the backup set so as to derive the required operations of adding and/or deleting subflows. New subflows are added first, whereas the deletion of old subflows starts after a maximum round-trip time (RTT) of the active paths so that the MPTCP sink waits to receive the packets on the fly over the paths to be deleted.

C. Subset-sum based relay selection

In order to select a best relay set as the active or backup set, we need to apply a real-time algorithm to efficiently update feasible relay sets whose total available bandwidths satisfy the TBR range. This is the well-known subset-sum problem, which is proved to be NP-complete. Given a set of N elements, there are totally 2^N possible subsets so that the searching scale is exponential. Fortunately, a fully polynomial-time approximation algorithm is available to "trim" subsets that have sums sufficiently close to neighbouring subsets [2]. Hence, we first adapt this approximation algorithm to obtain the feasible relay sets. The original approximation algorithm can determine the relay subsets whose total available bandwidths add up to an exact given value. Here, we need to find relay subsets whose total available bandwidths fall into a range $\big[(1-\theta)\mathrm{TBR}, (1+\theta)\mathrm{TBR}\big]$, $(0 < \theta < 1)$. This is because the available bandwidth of each relay path may vary dynamically and a small buffer space is enough to tolerate a certain level of throughput variation.

Given N relay UEs, we use r_i to denote the available bandwidth of relay UE i $(1 \le i \le N)$ and define $\mathbb{R} = \{r_1, r_2, ..., r_N\}$. All possible total available bandwidths of relay UEs $\{1, 2, ..., i\}$ are denoted by

$$\mathbb{L}_i = \{S_1^i, S_2^i, ..., S_{|L_i|}^i\} \qquad (1)$$
$$S_j^i \le (1+\theta)\mathrm{TBR}, \quad 1 \le j \le |L_i|.$$

All subsets of relay UEs $\{1, 2, ..., i\}$ that have a total available bandwidth S_j^i are denoted by

$$\mathbb{U}_j^i(S_j^i) = \big\{X = \{n_1, n_2, ...\} | \sum_{k \in X} r_k = S_j^i\big\} \qquad (2)$$

where the subset X contains the relay UEs $\{n_1, n_2, ...\}$.

Algorithm 1 shows the iterative procedure to obtain feasible relay subsets. In each round, a new bandwidth set $L_i^{'}$ is defined by adding the available bandwidth r_i of a new relay UE i to each element of the previous available bandwidth set L_{i-1}. As given in Line 4 of Algorithm 1, $L_i^{'} = \{S_1^{i'}, S_2^{i'}, ..., S_{|L_i^{'}|}^{i'}\}$, where $S_j^{i'} = S_j^{i-1} + r_i$ for any $1 \le j \le |L_i^{'}|$. That means, $L_i^{'}$ lists the total available bandwidths of subsets of relay UEs $\{1, 2, ..., i\}$, and those subsets must include relay UE i. The corresponding subsets of relay UEs $\mathbb{U}_j^{i'}(S_j^{i'})$ are also updated by adding the new relay UE i to each subset. Next, we merge the previous set L_{i-1} and the new set $L_i^{'}$, and then sort the combined set in a descending order. All elements greater than the TBR upper bound are removed because they are definitely greater than the TBR upper bound in the next round. All elements that fall into the TBR range are further trimmed by introducing an approximation parameter ε $(0 < \varepsilon < 1)$. Given

Algorithm 1 Polynomial-time subset-sum approximation.

1: $L_f = \{null\}$
2: $L_0 = \{0\}$
3: **for** $i = 1$ to N **do**
  // Consider a new relay i of available bandwidth r_i
4: $L_i^{'} = \{S_1^{i'}, S_2^{i'}, ..., S_{|L_i^{'}|}^{i'}\}$, where $S_j^{i'} = S_j^{i-1} + r_i$
5: **for** $j = 0$ to $|L_i|$ **do**
6: $\mathbb{U}_j^{i'}(S_j^{i'}) \leftarrow \big\{X = \{n_1, ..., i\} | \sum_{k \in X \setminus i} r_k = S_j^{i-1}\big\}$
7: **end for**
  // Merge sets L_{i-1} and $L_i^{'}$
  // Sort combined set in descending order
8: $L_i = \mathrm{MergeSort}(L_{i-1}, L_i^{'}) = \{S_0^i, S_1^i, ..., S_{|L_i|}^i\}$
  // Remove all elements greater than TBR upper bound
9: **for** $j = 0$ to $|L_i|$ **do**
10: **if** $S_j^i > (1+\theta)\mathrm{TBR}$ **then**
11: Remove S_j^i from L_i
12: **else if** $(1-\theta)\mathrm{TBR} \le S_j^i \le (1+\theta)\mathrm{TBR}$ **then**
13: **if** $S_j^i \notin L_f$ **then**
14: $L_f \leftarrow L_f \cup S_j^i$
15: **end if**
16: **else**
17: break
18: **end if**
19: **end for**
20: $\mathrm{Trim}\big(L_i, \varepsilon/(2N)\big)$
21: **end for**
22: Return $L_f = \{S_1^N, S_2^N, ..., S_{|L_f|}^N\}$ and $\mathbb{U}_j^N(S_j^N)$ for all $1 \le j \le |L_f|$

two neighbouring elements S_j^i and S_{j+1}^i, if they are sufficiently close, i.e., they satisfy the following equation,

$$\frac{S_{j+1}^i}{1 + \varepsilon/(2N)} \leqslant S_j^i \leqslant S_{k+1}^i \qquad (3)$$

then S_{j+1}^i is removed from L_i. Actually, ε is an indicator of the variance of the approximation result from the optimal solution. In this study, since the user requires to ensure an approximate total bandwidth in the range $[(1-\theta)\mathrm{TBR}, (1+\theta)\mathrm{TBR}]$, it is natural to set $\varepsilon = \theta$. Compared with the original subset-sum algorithm in [2], Algorithm 1 does not increase the search space, so that the running time remains polynomial in both $1/\varepsilon$ and N.

Based on the feasible relay UE subsets obtained from Algorithm 1, the best relay set can be selected according to two main factors, i.e., the difference between TBR and the total available bandwidth of a relay subset (denoted by ΔB, and the number of relay UEs (denoted by N_r). The two factors are normalized according to the following equations

$$\alpha_B = \frac{\Delta B - \Delta B_{\min}}{\Delta B_{\max} - \Delta B_{\min}}, \quad \alpha_N = \frac{N_r - N_{r,\min}}{N_{r,\max} - N_{r,\min}}. \qquad (4)$$

Then, we define the following priority index γ and select the relay subset of the highest priority index as the active set

$$\gamma = \frac{1}{\alpha_B + \alpha_N}. \tag{5}$$

III. Experimental Results and Discussion

To evaluate the performance of the proposed solution, we implement subset-sum based relay selection in the network simulator `ns-3` [9] for the scenario of Fig. 1. Specifically, we include the core functions of MPTCP such as socket APIs, coupled congestion control, path management, and packet scheduler. Consider an eNB connected to 11 UEs, among which there are 1 root UE and 10 relay UEs. These UEs are uniformly distributed within a rectangle area of a distance $800 - 1000$ meters to eNB. Each UE is equipped with both LTE and Wi-Fi interfaces. The relay UEs and root UE can use their Wi-Fi interfaces to directly communicate in an ad hoc mode. The root UE receives packets from the application server via relay UEs. The relay manager at the root UE monitors the available bandwidths of relay UEs for every 2 seconds. The detailed parameters are given in Table I.

It is known that the LTE channel fading and Wi-Fi link contention will introduce variation to the bandwidth provided by the relay UEs. We also adjust background traffic load based on user datagram protocol (UDP) at relay UEs to simulate varying available bandwidths. To evaluate the proposed dynamic relay selection solution, three UDP traffic patterns are considered to test the adaptiveness and achieved performance:

- *Static traffic load pattern*: the UDP traffic rates do not change during the simulation time, so that only the wireless channel effects such as fading can affect the available bandwidths.
- *Increasing traffic load pattern*: the UDP traffic rates at relay UEs are linearly increased over the simulation time.
- *Decreasing traffic load pattern*: the UDP traffic rates at relay UEs are linearly decreased over the simulation time.

To better understand the performance of the subset-sum based relay selection algorithm, referred to as *MPTCP-Subset sum* in the results, we consider a greedy relay selection scheme as a benchmark, referred to as *MPTCP-Greedy*. In this greedy scheme, the MPTCP sink simply adds relay UEs one by one until their total available bandwidth falls into the acceptable TBR range. When the total available bandwidth becomes higher than the TBR upper bound, the greedy scheme deletes relay UEs one by one until their total available bandwidth returns to the TBR range. If the sink fails to push the aggregate throughput back to the range, it ranks the relay UEs according to available bandwidths in an ascending order, and deletes the relay UEs from the lowest bandwidth to the highest, until the total available bandwidth is no greater than the TBR lower bound. In the following experiments, the available bandwidths and aggregate throughputs are updated for every 2 seconds in both MPTCP-Subset sum and MPTCP-Greedy.

A. Static traffic load pattern

Fig. 3 compares the aggregate throughput of MPTCP-Subset sum and MPTCP-Greedy. Two straight lines show the upper bound and lower bound of TBR requirement. In the static scenario, only the channel fading affects the available bandwidth of each relay UE. Once the total available bandwidth is out of the TBR range, the root UE can update the active relay set according to either the subset-sum based algorithm or the greedy algorithm. As seen in Fig. 3, MPTCP-Subset sum achieves a more stable aggregate throughput in comparison with MPTCP-Greedy. First of all, MPTCP-Subset sum results in less throughput variation to the root UE. This is because MPTCP-Subset sum takes into account the number of active relay UEs when selecting the active set at the beginning and throughout the MPTCP connection. Therefore, the impact of channel fading over each path can be under control. Conversely, MPTCP-Greedy only aims to satisfy TBR and uses more active subflows, which further exacerbates the impact of channel fading.

Another interesting observation of Fig. 3 is that MPTCP-Subset sum is less likely to have an aggregate throughput outside the TBR range. This is attributed to two main factors. First, MPTCP-Subset sum has a larger search space for relay UE subsets than MPTCP-Greedy. It efficiently scans most feasible subsets of relay UEs so as to select a relay set that provides a total bandwidth closest to TBR. In other words, MPTCP-Subset sum can tolerate larger variation because the total bandwidths of selected relay subsets are more concentrated in the middle of the TBR range. Therefore, there is a lower outage probability in the next update period. In contrast, MPTCP-Greedy only adds and deletes paths based on the

TABLE I
System parameters for experiments.

Parameter	Value
Transmit power of eNB	30 dBm
Transmit power of UE	23 dBm
Noise figure at eNB	5 dB
Noise figure at UE	5 dB
Transmission time interval (TTI)	1 ms
eNB scheduler	Blind equal throughput
Radio link control (RLC) mode	Acknowledge mode (AM)
Adaptive modulation & coding (AMC)	PiroEW2010 [11]
Number of resource blocks (RBs)	50
Fading channel trace	Pedestrian at 3 km/h
Wi-Fi link	IEEE 802.11a
Wi-Fi transmission rate	54 Mbit/s
Wi-Fi Fragmentation threshold	2200 bytes
Wi-Fi RTS/CTS threshold	2200 bytes
Number of relay UEs	10
Subset-sum algorithm parameters	ε = 0.1, θ = 0.1
Background traffic type	UDP
TBR	3 Mbit/s
Simulation time	50 seconds

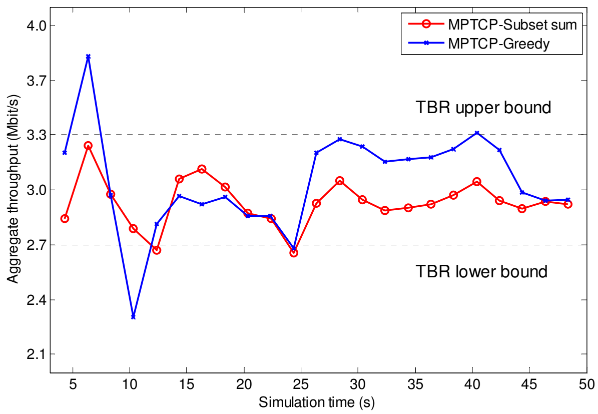

Fig. 3. Aggregate throughput in static traffic load pattern.

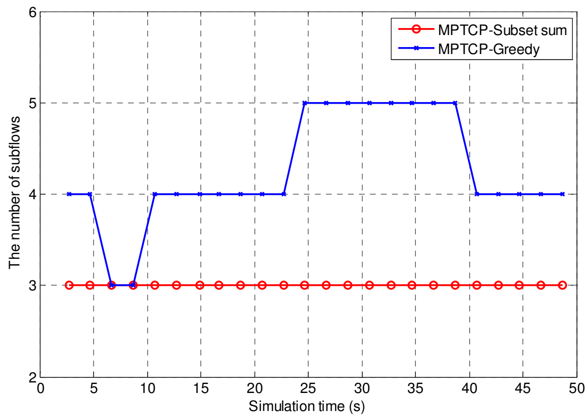

Fig. 4. The number of subflows in static traffic load pattern.

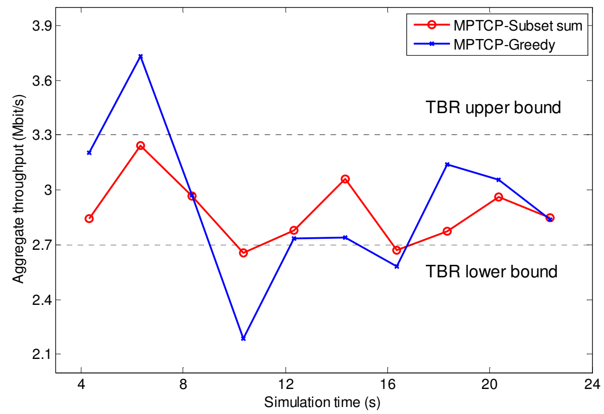

Fig. 5. Aggregate throughput in increasing traffic load pattern.

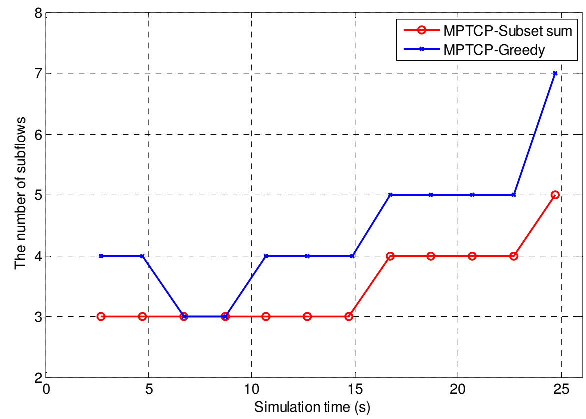

Fig. 6. The number of subflows in increasing traffic load pattern.

current active set. As a result, there is a larger chance to select a relay set having a total available bandwidth close to the TBR edges. For example, at 25 second, MPTCP-Subset sum selects a backup set of a total available bandwidth 2.9 Mbit/s, whereas the backup set selected by MPTCP-Greedy has a total bandwidth 3.25 Mbit/s, which almost approaches the TBR upper bound 3.3 Mbit/s. At 40 second, MPTCP-Greedy eventually violates the TBR boundary.

On the other hand, MPTCP-Subset sum aims to engage fewer subflows and thereby reduces throughput variation and the chance of throughput outage. This is more clearly demonstrated in Fig. 4. As seen, the number of subflows is very stable and fixed at 3 throughout the simulation time, although the specific relay UEs in use may change for every update period of 2 seconds. The main reason behind this behavior is the criteria we use for active set selection at the beginning of the MPTCP connection and the backup set selection during the run time. In both cases, MPTCP-Subset sum prefers to select the set with a smaller number of relay UEs, while ensuring the desired aggregate throughput. In contrast, MPTCP-Greedy only takes into account the aggregate throughput target. It simply deletes relay subflows from the active set when the TBR upper bound is violated (e.g., at 5 and 39 second), and adds new paths from inactive relay UEs when the root UE experiences a throughput less than the TBR lower bound (e.g., at 9 and 23 second).

B. Increasing and decreasing traffic load patterns

Given an initial local UDP throughput at each relay UE, in the increasing traffic load pattern, we linearly increase the UDP traffic load at each relay UE by 20% at 7, 13 and 19 second to simulate decreasing available bandwidths. Then, the local UDP throughput at each relay UE is linearly decreased by 20% at 31, 37 and 47 second to simulate decreasing traffic load pattern and increasing available bandwidths. We will use these two example cases to illustrate the responsiveness and effectiveness of the MPTCP-Subset sum algorithm.

Fig. 5 shows the aggregate throughput of MPTCP-Subset sum and MPTCP-Greedy in the increasing load pattern. As seen, the throughput achieved at the root UE goes down because of less available bandwidth at each relay UE. MPTCP-Subset sum adapts more smoothly with less variation and throughput outage than MPTCP-Greedy. As discussed above, this is because MPTCP-Subset sum effectively selects the backup set whose total available bandwidth is much closer to TBR. Thus, there is a larger guard space for throughput variation so as to minimize the possibility of throughput outage. Nonetheless, both algorithms need to employ more

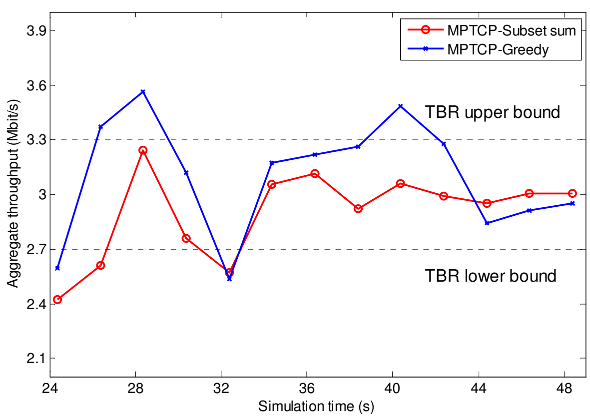

Fig. 7. Aggregate throughput in decreasing traffic load pattern.

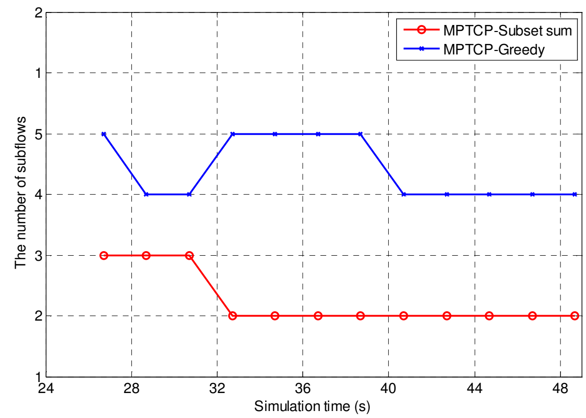

Fig. 8. The number of subflows in decreasing traffic load pattern.

relay UEs to accommodate the lower available bandwidth of each relay UE.

As shown in Fig. 6, MPTCP-Greedy activates much more subflows than MPTCP-Subset sum. With the decreased available bandwidths of relay UEs, MPTCP-Subset sum is able to switch to another relay set searched over a much larger space of relay UE subsets. In contrast, MPTCP-Greedy only relies on adding new relay UEs to the current active set so as to meet the desired TBR. For example, at 9 second, the aggregate throughputs of both algorithms are below the TBR lower bound. In this case, MPTCP-Greedy adds new relay UEs to increase the total available bandwidth, which increases the number of subflows from 3 to 4. However, MPTCP-Subset sum selects a backup set with the same number of subflows while achieving the desired TBR.

Fig. 7 and Fig. 8 show the aggregate throughput and the number of subflows in the decreasing traffic load pattern, respectively. It is seen in Fig. 7 that both MPTCP-Subset sum and MPTCP-Greedy achieve a higher aggregate throughput because of the increasing available bandwidths of relay UEs. Similarly to the behaviors exhibited in the increasing traffic pattern, MPTCP-Subset sum presents much less throughput variation compared to MPTCP-Greedy. A main reason is that MPTCP-Subset sum engages much fewer subflows, as shown in Fig. 8, which effectively mitigates the impact of bandwidth variation. Specifically, during 28 to 32 second, the aggregate throughputs are observed decreasing for both algorithms, which is mainly due to channel fading and violates the TBR lower bound at 32 second. At this point, MPTCP-Subset sum switches to another set of two relay UEs having larger available bandwidths. On the contrary, MPTCP-Greedy just adds new paths to increase the aggregate throughput, which leads to another throughput outage at 40 second. In addition, it is worth noting that, after 32 second, MPTCP-Subset sum only uses two relay UEs that have large available bandwidths, which enables a stable aggregate throughput around TBR for a long duration.

IV. Conclusions and Future Work

In this paper, we design an enhancement module within the application layer for a cooperative community running MPTCP in the LTE network. In particular, we develop a dynamic relay selection algorithm based on a fully polynomial-time subset-sum approximation [2]. By tracking available bandwidth variations, relay paths are effectively adapted in a highly varying environment so as to satisfy the application throughput requirement defined in terms of TBR. Specifically, the aggregate throughput at the root UE is maintained within an acceptable TBR range via adding and deleting relay paths. While the best relay set is updated periodically, an active set is migrated to the backup set whenever the aggregate throughput is observed out of the TBR range. Based on extensive simulations via `ns-3` [9] for varying background traffic patterns, we show that the proposed solution achieves a stable aggregate throughput with less performance outage and variation by engaging a much smaller number of subflows. It would be interesting to further investigate the performance of our solution with advanced background traffic patterns.

References

[1] A. Ford, C. Raiciu, M. Handley, S. Barre, and J. Iyengar, "Architectural guidelines for multipath TCP development," IETF RFC 6182, Mar. 2011.
[2] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algortihms*. The MIT Press, 2009.
[3] P. Sharma, S. Lee, J. Brassil, and K. Shin, "Aggregating bandwidth for multihomed mobile collaborative communities," *IEEE Trans. Mobile Comput.*, vol. 6, no. 3, pp. 1536–1233, Jan. 2007.
[4] D. Zhou, P. Ju, and W. Song, "Performance enhancement of multipath TCP with cooperative relays in a collaborative community," in *Proc. IEEE PIMRC'12*, Sep. 2012.
[5] P. Ju, W. Song, and D. Zhou, "A survey on cooperative medium access control protocols," *IET Communications*, to appear.
[6] S. Fu, K. Lu, T. Zhang, Y. Qian, and H. Chen, "Cooperative wireless networks based on physical layer network coding," *IEEE Trans. Veh. Technol.*, vol. 17, no. 6, pp. 86–95, Dec. 2010.
[7] Z. Hu and G. Li, "On energy-efficient TCP traffic over wireless cooperative relaying networks," *EURASIP Journal on Wireless Communications and Networking*, Mar. 2012.
[8] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, "TCP extensions for multipath operation with multiple addresses," IETF RFC 6824, Jan. 2013.
[9] "The network simulator - ns-3," http://www.nsnam.org/.
[10] M. Scharf and A. Ford, "MPTCP application interface considerations," IETF draft-ietf-mptcp-api-07, Jan. 2013.
[11] G. Piro, L. Grieco, G. Boggia, and P. Camarda, "A two-level scheduling algorithm for QoS support in the downlink of LTE cellular networks," in *Proc. European Wireless Conference (EW)*, Apr. 2010.