



HAL
open science

On the Energy Inefficiency of MPTCP for Mobile Computing

Mohammad Javad Shamani, Weiping Zhu, Saeid Rezaie

► **To cite this version:**

Mohammad Javad Shamani, Weiping Zhu, Saeid Rezaie. On the Energy Inefficiency of MPTCP for Mobile Computing. 14th International Conference on Wired/Wireless Internet Communication (WWIC), May 2016, Thessaloniki, Greece. pp.93-105, 10.1007/978-3-319-33936-8_8. hal-01434870

HAL Id: hal-01434870

<https://inria.hal.science/hal-01434870>

Submitted on 13 Jan 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

On the Energy Inefficiency of MPTCP for Mobile Computing

Mohammad Javad Shamani¹, Weiping Zhu¹, and Saeid Rezaie²

¹ School of Engineering and Information Technology, UNSW, Canberra
m.shamani@student.unsw.edu.au, w.zhu@adfa.edu.au

² School of Computer, West Tehran Islamic Azad University, Tehran
s.rezaie@wtiau.ac.ir

Abstract Mobile devices have embraced Multi-Path TCP (MPTCP) for leveraging the path diversity. MPTCP is a double-edged sword since mobile phones are suffering excessively from short battery life span. In order to find energy efficiency of MPTCP, the signal quality and the transferred file size have been taken into account. We formulate the above problem as a Markovian Decision Process (MDP) for symmetric and asymmetric network traffic. Numerical and simulation results surprisingly show that MPTCP is not efficient in any selected scenarios and the proposed scheme can save 56% of energy compared to the conventional MPTCP.

Keywords: MPTCP · Mobile device · Energy · MDP

1 Introduction

Wide coverage oriented cellular base stations and WiFi hot-spots coverage go hand in hand with arming mobile phones by different transmission interfaces such as cellular and WiFi, enhance the users' ubiquitous connectivity. Due to the error-prone nature of wireless channels, poor wireless signal quality is frequently experienced spatially and temporally by end users. It has been shown that the signal quality in WiFi or cellular-based networks dips in most cases. For instance, 43% and 21% of mobile phone data traffic is transmitted during low cellular and WiFi signal strength, respectively [2]. In addition, low signal strength could drain the mobile device battery. Due to the limited nature of mobile phone batteries, meeting high quality of service (QoS), while maintaining reasonable level of energy consumption, is of great importance for end-users. It is noteworthy that reaching high QoS and energy efficiency is not possible, in many cases, due to their conflicting requirements. Thus, designing a proper scheme to obtain reasonable levels of QoS and energy efficiency is critical.

Recently, MPTCP is introduced to enhance TCP performance by exploiting the path diversity [3,12]. However, base MPTCP suffers from energy inefficiency in mobile phones. For instance, when a mobile phone experiences poor signal quality, not only is an adverse effect on energy consumption inevitable, but also QoS

degradation is unavoidable. In [11] we show that the available bandwidth could be estimated by getting the signal quality as an observation and applying hidden Markov model; however, throughput is a more trustworthy metric to understand the different platforms energy consumption. In order to estimate the throughput accurately it is also important to consider the transferred file size. We explore how file size affects the throughput and how to estimate the file size by queue dynamics. The objective of this paper is to investigate whether MPTCP is more energy efficient than WiFi and LTE in real scenarios by leveraging simultaneous uploads and downloads in symmetric and asymmetric traffic pattern. Therefore, we proposed Signal Aware MPTCP (SA-MPTCP) to enhance MPTCP energy efficiency. The contribution of this paper is threefold as follows:

- We make a tight relation between signal quality, file size and energy consumption; besides the optimization problem is formulated as a Markovian decision making problem by employing buffer size dynamics to better understand transferring file sizes.
- We endeavor to minimize energy consumption by exploiting diverse characteristics of interfaces, and for the first time we have found that MPTCP is not energy efficient in real scenarios, particularly in simultaneous uploading and downloading.
- We investigate the effect of energy consumption on asymmetric and symmetric data traffic for a mobile user facing variety of signal strength on different interfaces.

The paper is organized as follows. Section 2 introduces the energy model, and shed light on the effect of file size on MPTCP throughput. Section 3 describes the problem formulation. Section 4 outlines the simulation results, and Section 5 reviews the related works. Finally, we conclude in Section 6.

2 proposed model

The proposed model aims to optimize the transmission energy consumption as a utility function based on signal strength and transferring file size. In the following, we describe the underlying network model and assumptions for the problem at hand.

2.1 Energy model

The energy model used here addresses energy consumption for uploading and downloading by taking throughput into account. Where throughput would be estimated in our optimization problem by the interface's signal-to-noise ratio (SNR) and the transferring file size. To the best of our knowledge, the energy model introduced in [4] is the only proposed model that analyses energy consumption for WiFi, cellular networks and MPTCP. Considering this, we use the same concept in describing network model. We assume that the mobile device is equipped with two interfaces to transmit data. Let i denote the interface, where $i = w$ and $i = l$ represent the WiFi and LTE interface, respectively. Available uplink and downlink throughput for interface i is denoted by $Th_{u,i}$ and $Th_{d,i}$, respectively.

The energy consumption in uploading data for interface i which is based on Joule (J) per byte (B), and is defined by E_i^u

$$E_i^u = (\alpha_i^u \times Th_{u,i}^{\beta_i^u}) \quad (\mu J/B), \quad (1)$$

where α_i^u and β_i^u are upload packet transfer coefficients of interface i , which are obtained through regression model in [4]. Similarly, for downloading, we have

$$E_i^d = (\alpha_i^d \times Th_{d,i}^{\beta_i^d}) \quad (\mu J/B), \quad (2)$$

where α_i^d and β_i^d are download packet transfer coefficient in interface i . Hence, total energy consumption of interface i in downlink and uplink can be formulated as \mathcal{E}_i which is equal to

$$\mathcal{E}_i = E_i^u \times L_i^u + E_i^d \times L_i^d \quad (J), \quad (3)$$

where L_i^u and L_i^d are the size of uploading and downloading files in bytes for interface i , respectively. In [4], it has been shown that due to the concurrent use of interfaces (WiFi and LTE) in MPTCP, the energy consumption is less than total energy consumption of WiFi and LTE, if they are not used jointly. MPTCP energy consumption, denoted as \mathcal{E}_m , can be written as

$$\mathcal{E}_m = (\mathcal{E}_w + \mathcal{E}_l) - \theta, \quad (4)$$

where θ represents the shared component energy. In MPTCP about 13% ~16% energy would be consumed simultaneously [4]. Thus, in this paper we consider θ as a fixed number 14.5%. Packet transfer coefficients for different platforms based on uplink and downlink are shown on table 1.

Table 1: Packet transfer coefficients [4]

		LTE	WiFi
Download	α^d	10.04	4.64
	β^d	-0.89	-0.81
Upload	α^u	13.34	3.61
	β^u	-0.83	-0.66

2.2 Effect of file size on throughput

Estimating the available bandwidth by knowing the signal quality is possible. For instance, Fig. 1 shows the uplink and downlink throughput as a function of received signal strength indicator (RSSI), where data is extracted from [9], [2], and [5]. However, the amount of transfer size has a significant effect on MPTCP throughput. In Fig. 2 and Fig. 3 throughputs for MPTCP, WiFi and LTE have been plotted based on the transferred file size if the WiFi and LTE RSSI are (-76,-90) and (-106,-135), respectively. As it can be seen as the file size increases, the

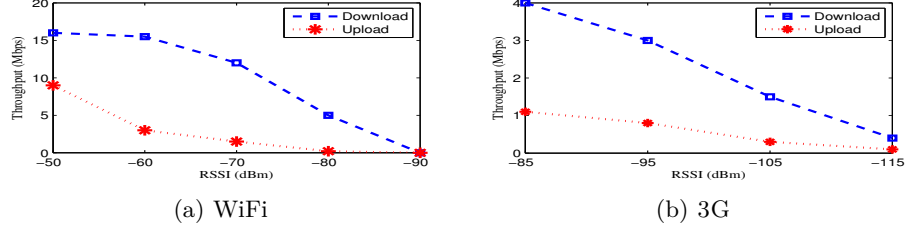


Figure 1: Average throughput based on signal strength

throughput reaches to the maximum capacity, which is the available bandwidth. Based on the available signal quality state and file size, the uplink and downlink throughputs could be estimated by a regression model. We have calculated the uplink and downlink coefficient for the noted RSSI states in table 2 and table 3. The uplink throughput for interface i is defined as

$$Th_{u,i} = \kappa_i^u * (L^u)^{\eta_i^u} + l_i^u, \quad (5)$$

where κ_i^u and η_i^u are uplink coefficients for interface i . Downlink throughput is equal to

$$Th_{d,i} = \kappa_i^d * (L^d)^{\eta_i^d} + l_i^d, \quad (6)$$

where κ_i^d and η_i^d are defined as downlink coefficients for interface i . It is noteworthy that the number of selected states uplink and downlink tables should be calculated prior to the simulation.

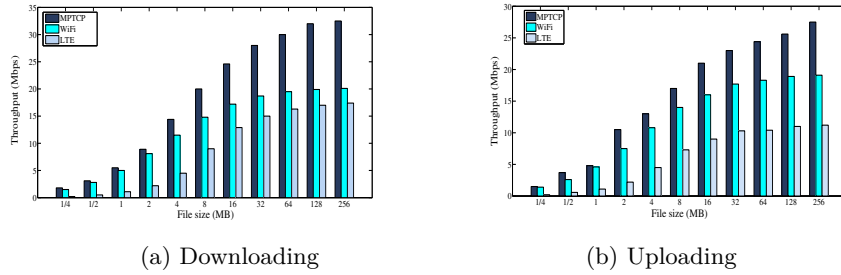


Figure 2: Platforms throughput for different file sizes

3 Problem formulation

3.1 Definitions

A mobile device decision making is based on the assumption that the system dynamics are determined by an MDP. MDP consists of the following elements:

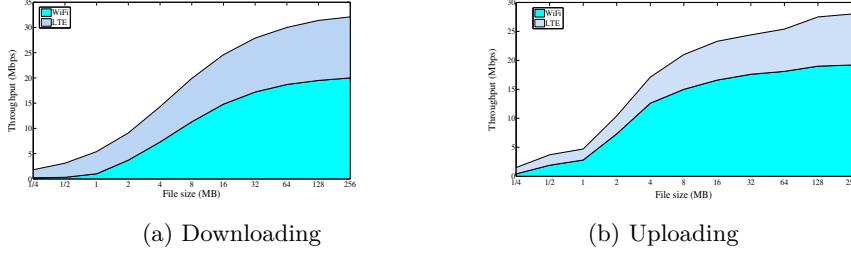


Figure 3: MPTCP throughput for different file sizes

Table 2: Upload coefficients

	κ^u	η^u	ι^u
TCP over WiFi	-23.35	-0.16	29.48
TCP over LTE	-27.38	-0.08	29.62
MPTCP over WiFi	-27.58	-0.14	32.57
MPTCP over LTE	205.7	0.0052	-203.3

decision epochs, states, states transition probabilities, actions and costs. The mobile device can make a decision about tuning one or both interfaces from idle to active mode in each time epoch, where $\mathcal{T} = \{1, 2, \dots, y\}$ represent a set of decision epoch. The signal quality states are introduced by the following: firstly, a set of signal quality states for interface i , $\mathcal{R}_i = \{r_i^1, r_i^2, \dots, r_i^n\}$, where r_i^1 is minimum and r_i^n is maximum signal quality state. It is worth adding here that each signal quality corresponds to specific uplink and downlink bandwidth. Secondly, signal quality states transition probability matrix for interface i , $P^{r_i} = [p^{r_i}(r_i^k | r_i^j), r_i^1 \leq r_i^j, r_i^k \leq r_i^n]$. In [11] we have shown that how to find the most probable signal quality states by getting RSSI as an observation and in this paper we assume the signal quality state is known. Let $\mathcal{S} = \mathcal{R}^1 \times \mathcal{R}^2 = \{(r_1^1, r_2^1), (r_1^1, r_2^2), \dots, (r_1^n, r_2^n)\} = \{s_1, s_2, \dots, s_N\}$ represent the total state space, where $\mathcal{N} = |\mathcal{R}_1| \times |\mathcal{R}_2|$. The action space $\mathcal{A} = \{(on, on), (on, off), (off, on)\}$ denotes a set of all possible actions for both interfaces. Without loss of generality, let LTE be the first interface and WiFi be the second; consequently, (on, off) means LTE interface is active and the WiFi interface is idle.

The transition probability for the next state is $p(s_{t+1} | s_t, a_t)$. Since the state transition probability is the product of all state transition probabilities, we can rewrite it as

$$p(s_{t+1} | s_t, a_t) = p^{r_i}(\zeta(s_{t+1}) | \zeta(s_t)) p^{r_j}(\zeta(s_{t+1}) | \zeta(s_t)), \quad (7)$$

Table 3: Download coefficients

	κ^d	η^d	ι^d
TCP over WiFi	-24.1	-0.17	30
TCP over LTE	541	0.0055	-539
MPTCP over WiFi	-95	-0.039	98.22
MPTCP over LTE	-14.05	-0.15	18.59

where $\zeta(s_t)$ gives the signals quality state of the compound state s_t . At time-slot t , when the action a_t is taken, the system shifts to the state s_{t+1} ; therefore, the system incurs costs expressed as $c(s_t, a_t, s_{t+1})$.

Before discussing decision rules and policies, we have to shed light on cost calculation. In order to get the benefit of high accuracy of throughput estimation we need to have information about file size. Therefore, we use the backlog queue to have a better understanding of transferring file size.

Consider the proposed scheme operates n slotted times with time slots $t \in \{0, 1, 2, \dots\}$. The uplink and downlink queues as defined as Q^u and Q^d , respectively. At the beginning of each time slot, new packets arrive randomly. Let $\lambda_t = (\lambda_t^1, \lambda_t^2, \lambda_t^N)$ be the arrival rate vector. The service rate vector at each time slot t is defined as $\mu_t = (\mu_t^1, \mu_t^2, \dots, \mu_t^N)$. The service rate is calculated based on the selected action at time t . Let's define the queue backlog as Q_t , then the next backlog state are driven by stochastic arrival and service rates. Thus, the backlogs dynamics are given by

$$Q_{t+1} = \max[Q_t - \mu_t, 0] + \lambda_t. \quad (8)$$

In the proposed scheme at each time slot we only observe the arrival rate and the quality of signal strength. Then we employ Q_{t+1} to estimate the transfer file size in uploading and downloading. Let's define the expected immediate cost as

$$c_t(s_t, a_t) = \begin{cases} \mathcal{E}_w + \mathcal{E}_l & a = (\text{on}, \text{on}) \\ \mathcal{E}_l & a = (\text{on}, \text{off}) \\ \mathcal{E}_w & a = (\text{off}, \text{on}) \end{cases}. \quad (9)$$

In the explained MDP problem, the objective is minimizing the expected long-term cost which is defined as

$$v^\pi(s) = \min_{\pi} \mathbb{E}^\pi \left\{ \sum_{t=1}^T \gamma^t c_t^\pi(s_t, a_t^\pi(s_t)) \right\}. \quad (10)$$

Solving (10) would be computationally expensive; therefore, Let's define a policy by π , which maps states to actions. MDP endeavors to find a policy π that minimizes some cumulative function of the stochastic costs. Given a cost criterion, a policy has an expected value for every state, specifying the cost of mobile phone receiving from following a policy in that state. Policy π^* is an best policy if there is no policy π' , and no state s such that value of π' be lower than the value of π . Letting value of being at state s_{t+1} as $v_{t+1}(s_{t+1})$, then in each time step we aim for an action which has the minimum cost. let's express it as

$$a_t^*(s_t) = \arg \min_{a_t \in A_t} (c_t(s_t, a_t) + \gamma v_{t+1}(s_{t+1})), \quad (11)$$

where γ is a discount factor ($0 \leq \gamma < 1$). Because this value shows the cost a mobile device incurs during one time period in the future, we may discount it by

a factor γ . By using optimal action $a_t^*(s_t)$, the optimal value is equal to

$$\begin{aligned} v_t(s_t) &= \arg \min_{a_t \in A_t} (c_t(s_t, a_t) + \gamma v_{t+1}(s_{t+1}(s_t, a_t))) \\ &= c_t(s_t, a^*(s_t)) + \gamma v_{t+1}(s_{t+1}(s_t, a^*(s_t))). \end{aligned} \quad (12)$$

Because the problem is a decision problem with stochastic components and the new information would be available after The decision is made, we need to add uncertainty to (12)

$$\begin{aligned} v_t(s_t) &= \min \left\{ c_t(s_t, a_t) \right. \\ &\quad \left. + \gamma \sum_{s_{t+1} \in S} p(s_{t+1}|s_t, a_t) v_{t+1}(s_{t+1}) \right\}. \end{aligned} \quad (13)$$

(13) is widely known as the standard bellman equation [8]. We seek the best action in each state minimizing over policies. Thus, the expected cost provided by a policy π from time t onward is defined as

$$F_t^\pi(s_t) = \mathbb{E} \left[\sum_{t+1}^{T-1} c_{t+1}(s_{t+1}, a_{t+1}^\pi(s_{t+1})) + c_t(s_t) \mid s_t \right], \quad (14)$$

where $F_t^\pi(s_t)$ represents the expected total cost in state s followed by a policy π from time t onward. To find (14) solution, we need to recursively calculate v_t^π . It has been shown that (14) is equal to $v_t^\pi(s_t)$ [7]. By letting $v_t(s_t)$ as a solution for (13), we will have

$$F_t^* = \min_{\pi \in \Pi} F_t^\pi(s_t) = v_t(s_t). \quad (15)$$

There are several ways to solve the infinite horizon optimization problems. For example, value iteration, or policy iteration algorithms [7]. The value iteration algorithm estimates the value function iteratively. At each iteration the best decision will be selected by a policy. In fact, value iteration begins at the end, and then goes backward to find out the best policy. The iteration counter starts at 0, and goes up until the algorithm satisfies the left-hand-side of (16). Thus, v is the largest absolute value, and the value iteration algorithm stops if the largest modify in the value of being in any state is less than the right-hand-side of (16).

$$\|v_n - v_{n-1}\| < \epsilon(1 - \gamma)/2\gamma, \quad (16)$$

where ϵ is an error tolerance parameter [7].

4 Simulation and results

In order to appraise the proposed algorithm, we have performed the simulations by *ns-3* (<https://www.nsnam.org>). The simulation parameters are included in table 4. We have considered three LTE and three WiFi signal quality states. Based on the selected states, we found nine spots on the simulation field as it noted in Fig. 4. We calculated nine upload and download tables as in table 2

ALGORITHM 1: SA-MPTCP Algorithm

Input : λ_t^u, λ_t^d
Input : ϵ , a tolerance parameter
Output : Best Policy for each state
Begin
 set $\rightarrow v_0=0$
 set $\rightarrow n=1$
 set $\rightarrow \lambda_u=\lambda_t^u, \lambda_d=\lambda_t^d$
 $L_{t+1}^u = \max[L_t^u - \mu_t^u, 0] + \lambda_t^u$
 $L_{t+1}^d = \max[L_t^d - \mu_t^d, 0] + \lambda_t^d$
for $s \in S$ **do**
 $v_n(s_t) = \max(c(s_t, a_t)) + \gamma \sum_{s_{t+1} \in S_t} P(s_{t+1}|s_t, a_t)v_{n-1}(s_{t+1})$
 if $\|v_n - v_{n-1}\| < \epsilon(1 - \gamma)/2\gamma$ **then**
 $\pi^\epsilon = \pi^*, v_\epsilon = v_n$
 else
 $n = n + 1$
 end
end
if action $a = (on, off)$ **then**
 Enable MPTCP in backup mode
 Establish LTE sub-flow, postpone WiFi sub-flow establishment
else if action $a = (off, on)$ **then**
 Enable MPTCP in backup mode
 Establish WiFi sub-flow, postpone LTE sub-flow establishment
else
 Enable MPTCP in full mode
end
End

and 3. Before the simulation the best policy in each state for different arrival rates have been calculated . After that the traffic has been generated between two nodes, and the mobile node (UE-1) moves from one spot to another by using the most efficient policy in each state.

In order to compare the proposed scheme to other platforms, we have simulated the same scenario using MPTCP, WiFi, and LTE in all spots. Also to have a practical perspective and general view of how the system works, two different file sizes have been selected for uploading and downloading in symmetric and asymmetric traffic scenarios.

From Fig. 5 to Fig.8, the optimal policies have been marked by a red arrow. Taking the advantage of the packet capture feature of NS3, we have plotted the energy consumption behavior of other policies in each state. SA-MPTCP optimal decision was completely compatible with real simulation and surprisingly MPTCP has not been chosen in any scenario. In fact, this is the first time that uploading and downloading have been considered together, and the results contrast prior research such as [4] and [6]. Most of the time either WiFi or LTE are the best policy and even according to Fig. 6 WiFi is the best option for

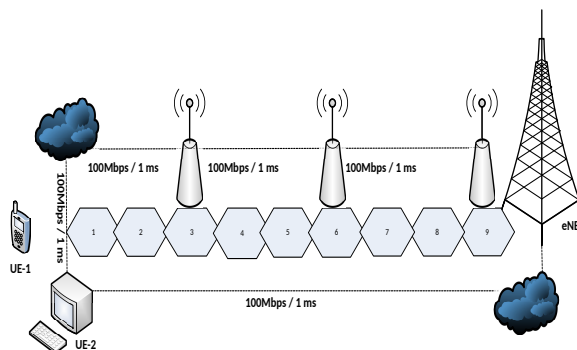


Figure 4: Simulation topology

Table 4: Simulation parameters

LTE RSSI states	$(-45, -75), (-76, -105), (-106, -135)$
WiFi RSSI states	$(-50, -61), (-62, -75), (-76, -90)$
Point to Point bandwidth	100 Mbps
Point to Point delay	1 ms
Direct Code Execution (DCE)	version 1.7

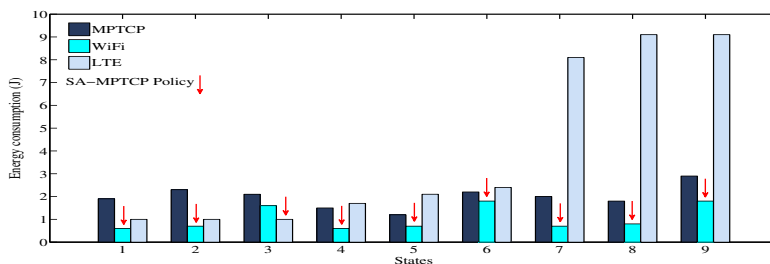


Figure 5: 256 KB upload and 256 KB download energy consumption

uploading small file size and downloading large file size. SA-MPTCP energy consumption outperforms other platforms.

Comparing SA-MPTCP to MPTCP, WiFi, and LTE, SA-MPTCP saves 56%, 16% and 65% energy, respectively. On the other hand, MPTCP is 23% more throughput efficient than SA-MPTCP. However, SA-MPTCP achieves a higher throughput compared to WiFi and LTE.

5 Related works

MPTCP has been investigated widely from different perspective; nonetheless, energy efficiency attracts the growing interest of researchers. Comparison of

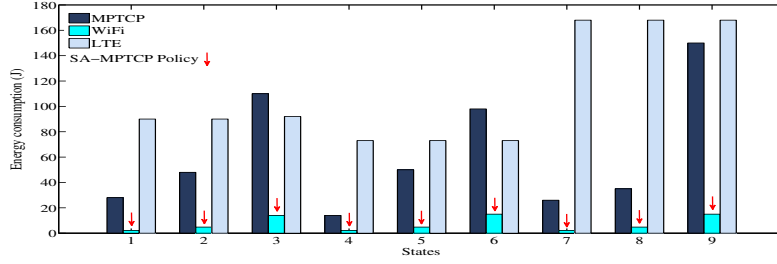


Figure 6: 256 KB upload and 32 MB download energy consumption

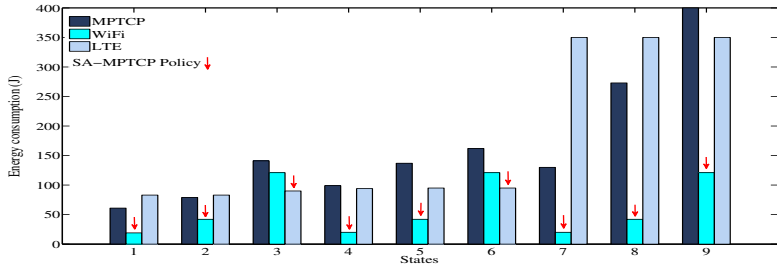


Figure 7: 32 MB upload and 256 KB download energy consumption

TCP over LTE and WiFi with MPTCP in terms of energy efficiency have been discussed in [6] by modeling, measuring and simulating. The authors proposed eMPTCP, which manages subflows expected energy efficiency based on available throughput. In fact eMPTCP is the most similar work to us, but the difference is eMPTCP only considered the downloading case and doesn't take the uploading energy into account. In addition, to estimate the available throughput eMPTCP sampled downloaded bytes, which introduces the overhead to a network.

GreenBag [1] was proposed as a bandwidth aggregate middleware on wireless links. It estimates the available bandwidth on wireless links and specifies the amount of traffic allocation while taking QoS into account. GreenBag reduces

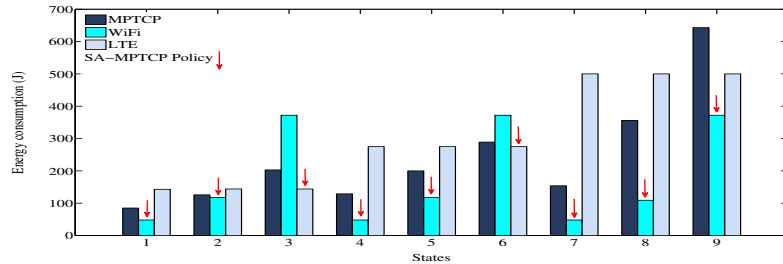


Figure 8: 32 MB upload and 32 MB download energy consumption

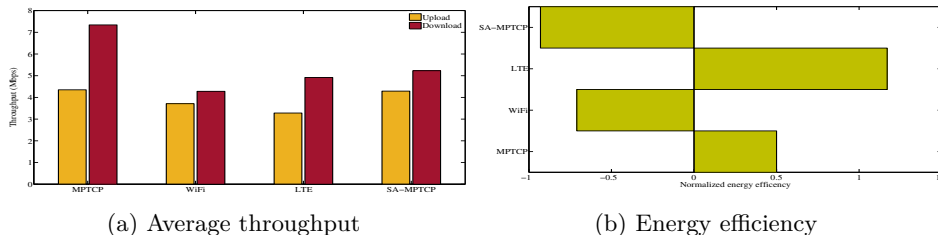


Figure 9: Different platforms throughput and energy efficiency

energy consumption between 14% to 25%; however, it needs major applications modification which is not feasible. Peng et al. [6] designed a MPTCP algorithm, decreasing energy consumption up to 22% by intelligently compromising between throughput and energy. However, they only considered bulk data applications such as video streaming and file transfer, and they did not consider the effect of signal strength on throughput.

Markov decision process has been applied by [7] to utilize the optimal interface for different applications. Nevertheless, authors only considered throughput and stationary applications which is infeasible in dynamic environment. For instance, each application has different traffic patterns, and modeling all applications traffic is not practical. Furthermore, the optimal throughput performance is only achieved by one path, and more importantly the system performance would dwindle immensely when the optimal path is the most congested one. Energy aware MPTCP is proposed in [8] to balance the increasing throughput with the energy consumption. Unfortunately, data has been offloaded simply to WiFi whenever possible since they assumed WiFi consumes less energy than LTE. That assumption may not be true in many scenarios. In fact, when a mobile device experiences a weak WiFi signal, not only a huge amount of packet loss and delay would be irrefutable [5], but also as we showed, energy cost would increase sharply.

Lim et al. [5] proposed an inspiring algorithm, MPTCP-MA, which manages the path establishment by taking cross layer information, such as Mac layer behavior. They endeavored to increase the WiFi throughput efficiency up to 70%; nonetheless, energy efficiency hasn't been taken into account, and they haven't addressed cellular networks behavior. In contrast to most studies, Tailless MPTCP (TMPTCP) [10] showed that MPTCP energy efficiency could be enhanced by the tail energy minimization. Since the tail energy has a enormous impression on the energy contribution, TMPTCP optimizes jointly energy and delay. However, TMPTCP deals with static bandwidth and the offline experiment demands more investigation.

6 Conclusion

In this paper, we studied the problem of energy efficiency in the context of multi-path TCP. We proposed a control mechanism based on Markov Decision Process

to distribute traffic over diverse paths while optimizing energy. Performance of the proposed control mechanism was evaluated and compared with the different platforms. The results indicate that the proposed control mechanism significantly outperforms other schemes in all considered scenarios and MPTCP is not energy efficient in any selected cases. Even though we have applied our approach to LTE, the model could simply extend to 3G networks. The effect of delay would be investigated in future work.

References

1. Bui, D.H., Lee, K., Oh, S., Shin, I., Shin, H., Woo, H., Ban, D.: Greenbag: Energy-efficient bandwidth aggregation for real-time streaming in heterogeneous mobile wireless networks. In: Real-Time Systems Symposium (RTSS), 2013 IEEE 34th. pp. 57–67. IEEE (2013)
2. Ding, N., Wagner, D., Chen, X., Pathak, A., Hu, Y.C., Rice, A.: Characterizing and modeling the impact of wireless signal strength on smartphone battery drain. In: ACM SIGMETRICS Performance Evaluation Review. vol. 41, pp. 29–40. ACM (2013)
3. Ford, A., Raiciu, C., Handley, M., Barre, S., Iyengar, J., et al.: Architectural guidelines for multipath tcp development. IETF, Informational RFC 6182, 2070–1721 (2011)
4. Lim, Y.s., Chen, Y.C., Nahum, E.M., Towsley, D., Gibbens, R.J.: Improving energy efficiency of mptcp for mobile devices. arXiv preprint arXiv:1406.4463 (2014)
5. Lim, Y.s., Chen, Y.C., Nahum, E.M., Towsley, D., Lee, K.W.: Cross-layer path management in multi-path transport protocol for mobile devices. In: INFOCOM, 2014 Proceedings IEEE. pp. 1815–1823. IEEE (2014)
6. Peng, Q., Chen, M., Walid, A., Low, S.: Energy efficient multipath tcp for mobile devices. In: Proceedings of the 15th ACM international symposium on Mobile ad hoc networking and computing. pp. 257–266. ACM (2014)
7. Powell, W.B.: Approximate Dynamic Programming: Solving the curses of dimensionality, vol. 703. John Wiley & Sons (2007)
8. Puterman, M.L.: Markov decision processes: discrete stochastic dynamic programming. John Wiley & Sons (2014)
9. Schulman, A., Navda, V., Ramjee, R., Spring, N., Deshpande, P., Grunewald, C., Jain, K., Padmanabhan, V.N.: Bartendr: a practical approach to energy-aware cellular data scheduling. In: Proceedings of the sixteenth annual international conference on Mobile computing and networking. pp. 85–96. ACM (2010)
10. Shamani, M.J., Zhu, W., Naghshin, V.: Tmptcp: Tailless multi-path tcp. In: to appear in BWCCA, 2015 Proceedings IEEE. IEEE (2015)
11. Shamani, M.J., Zhu, W., Rezaie, S., Naghshin, V.: Signal aware multi-path tcp. In: WONS, 2016 Proceedings IEEE/IFIP. pp. 104–107. IEEE/IFIP (2016)
12. Wischik, D., Raiciu, C., Greenhalgh, A., Handley, M.: Design, implementation and evaluation of congestion control for multipath tcp. In: NSDI. vol. 11, pp. 8–8 (2011)