A Contextual Bi-armed Bandit Approach for MPTCP Path Management in Heterogeneous LTE and WiFi Edge Networks

Aziza Alzadjali
Department of Computer Science
University of Nebraska-Lincoln
Nebraska, USA
aalzadjali@cse.unl.edu

Flavio Esposito
Department of Computer Science
Saint Louis University
Saint Louis, USA
flavio.esposito@slu.edu

Jitender Deogun
Department of Computer Science
University of Nebraska-Lincoln)
Nebraska, USA
deogun@cse.unl.edu

***Abstract*—Multi-homed mobile devices are capable of aggregating traffic transmissions over heterogeneous networks. Multi-Path TCP (MPTCP) is an evolution of TCP that allows the simultaneous use of multiple interfaces for a single connection. Despite the success of MPTCP, its deployment can be enhanced by controlling which network interface to be used as an initial path during the connectivity setup. In this paper, we proposed an online MPTCP path manager based on the contextual bandit algorithm to help choose the optimal primary path connection that maximizes throughput and minimizes delay and packet loss. The contextual bandit path manager deals with the rapid changes of multiple transmission paths in heterogeneous networks. The output of this algorithm introduces an adaptive policy to the path manager whenever the MPTCP connection is attempted based on the last hop wireless signals characteristics. Our experiments run over a real dataset of WiFi/LTE networks using NS3 implementation of MPTCP, enhanced to better support MPTCP path management control. We analyzed MPTCP's throughput and latency metrics in various network conditions and found that the performance of the contextual bandit MPTCP path manager improved compared to the baselines used in our evaluation experiments. Utilizing edge computing technology, this model can be implemented in a mobile edge computing server to dodge MPTCP path management issues by communicating to the mobile equipment the best path for the given radio conditions. Our evaluation demonstrates that leveraging adaptive context-awareness improves the utilization of multiple network interfaces.**

I. Introduction

The edge computing paradigm is expected to use different technologies to deliver services and applications with high-throughput and low-latency demands. To this aim, optimizing transmissions at the last mile within a wireless edge network is important, especially for critical applications [1–4]. Nowadays, smartphones, tablets, and laptops are multi-homed, i.e., they are equipped with multiple radio interfaces, such as WiFi and LTE. Past research has shown that standard single-path TCP cannot efficiently serve the coexistence of multi-access technologies. To allow efficient exploitation of multiple Internet paths for the same user connection, several variations of the Multipath TCP (MPTCP) protocol [5, 6] have been recently proposed, with a vibrant activity of the IETF [7]. For example, researchers have used MPTCP to exploit edge-cloud ecosystems [8], to optimize the offloading mechanism in 5G networks [9], to improve video streaming sessions [10], as well as within online gaming, energy-aware telecommunications [11], or even to improve the throughput in Unmanned Aerial Systems [12] and other IoT transmissions. One of the fundamental mechanisms of the MPTCP protocol is the algorithm that governs the path management of MPTCP. Some notable examples of path management schema that have recently been proposed include Context-Aware Multipath-TCP [13], Dynamic MPTCP Path Configuration with SDN [14], Multipath TCP with Path-aware Information [15], Ndiffport Subflow Manager for Data Centres [16], and Fullmesh Path Manager [17]. Nowadays, the Fullmesh has been chosen as the default path manager algorithm in all MPTCP implementations. As the name suggests, the algorithm connects the user with all available transmission paths by creating a full mesh of subflows between the communicating hosts. While the Fullmesh path manager has merit, it is known to be suboptimal in dynamic network environments. One known problem of Fullmesh is that it always starts with establishing the WiFi path first, potentially affecting the MPTCP throughput significantly when the WiFi signal is weak [18]. Moreover, the Fullmesh path manager of MPTCP assigns subflow statically; this means that when one subflow fails, e.g., for excessive retransmissions, it is hard to re-establish it, causing significant performance degradation. Finally, the Fullmesh strategy leads to a large number of established subflows and ignores the benefits offered by path diversity.

The widespread machine learning techniques inspired us to seek a "dynamic" path manager. Various efforts have been carried out to apply machine learning to MPTCP. For example, Rosello and Molla [19] considered the scheduling problem in MPTCP with a non-standard multipath implementation of the QUIC protocol. They used Deep Q-Network reinforcement learning to determine the best action chosen by the agent. Xu et al. [20] presented a deep reinforcement learning-based architecture for MTCP to maximize the throughput of the LTE and WiFi flows. Qi et al. [14] used a support vector machine approach as a regression model to predict the

MPTCP throughput ratio to improve its performance. While the solutions behind these machine learning approaches are sound, their models were designed based on just a limited feature set and did not capture the features of multiple available paths in heterogeneous networks, as we do. Considering the features of multiple available paths in MPTCP is essential for a few reasons. First, the decision of which technology should be the primary path should consider the radio conditions. Second, existing path manager learning algorithms are mostly based on full reinforcement learning models, that are often expensive to train and converge slowly. In the context of MPTCP path management, the state space increases exponentially with the number of features in each subflow, but multiple consecutive actions are not required. Therefore a more efficient learning framework is desirable. Third, as learned from the simulations, deploying a learning-based path manager mechanism in a real wireless network environment has several practical issues, such as the exploration strategy in each location, the cost of model training, and the guarantee of training on real data for real-time decision making.

To cope with the above three challenges, in this paper, we focus on establishing a learning framework for self-evolving path management withing MPTCP, capable of adapting to the diverse last mile wireless radio environment. In particular, we show that the MPTCP path manager module can be viewed as a learning task with an agent seeking optimal actions that maximize a definition of reward in a dynamic wireless network environment. To do so, we proposed a contextual bandit path manager whose workflow is composed of three stages: *(i) collection of contexts, (ii) training and updating the agent oracle, and (iii) online recommendation*. During the collection of contexts, without prior knowledge of the optimal action, the learning agent follows some heuristic path manager rules to explore and takes actions using a probabilistic approach exploring its historical experience. In the training stage, the agents' oracle trains a Stochastic Gradient Descent (SGD) binary classifier based on the collected context experiences and generates actions representing the optimal primary path for the MPTCP connection setup. In the recommendation stage, the MPTCP system uses the up-to-date actions to make online path management decisions and feedback the agent as experience for future learning and model refitting. To improve the learning efficiency, we measure the cumulative reward of the agent policies to derive the optimal path manager.

Our Contributions. The main contributions of this work are summarized as follows. We first present an architecture for an online learning-based network transmission path selection with a high cumulative mean reward. The focus of the learning is on resolving and optimizing path diversity decisions in heterogeneous networks. e.g., composed of both WiFi and LTE technologies. We then formulate the MPTCP path management problem as a contextual bandit learning task and design a Bernoulli reward function to address the path diversity component of MPTCP. Our reward function aims at achieving optimal throughput for every connection by utilizing the maximum available network resources. We prototype our solution within NS3 to demonstrate the proposed system by modifying the MPTCP path manager module. The simulated contextual bandit path manager framework collects radio interface data, train the model to generate classification predictions, apply the predicted actions for a real-time online decision of MPTCP primary path manager connection setup, and update the agent asynchronously to adapt to radio coverage rapid changes. Our experiments compare our contextual bandit path manager's performance with other approaches using available WiFi and LTE measurement traces from 20 separate locations across the USA. We show that our contextual bandit path manager improves the aggregated throughput by up to 30% compared to baseline algorithms.

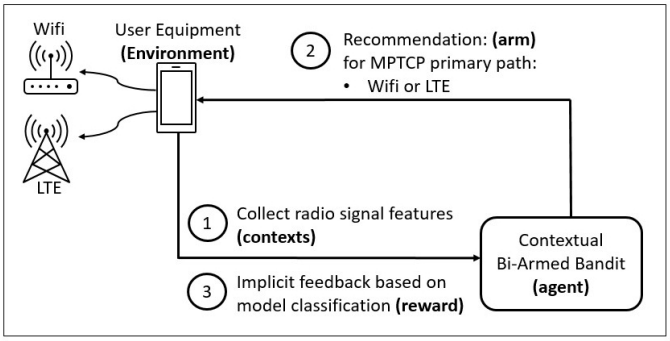

Fig. 1. System Architecture: Online contextual bandits for MPTCP primary path selection: LTE or Wi-Fi.

II. Path Manager Importance in Multi-Path TCP

The Multi-path TCP (MPTCP) provides concurrent transmissions to increase resilience of the connectivity and maximizes the usage of all the available network resources. MPTCP implementations are backwards compatible with all versions of TCP, and allow the transmission of a single data stream to be split across multiple paths. Mobile devices designed with two wireless interfaces are a common use case for MPTCP. The MPTCP consists of three main components, the scheduler, the path manager and the congestion control. Path management task is to only establish new subflows. How these subflows are used during the connection is determined by the scheduler. A good path manager algorithm is needed to manage the multiple paths creation efficiently, each path is equivalent to a subflow and is identified by a pair of source and destination IP addresses. It is used to explore and initialize multiple paths, manage signaling alternative addresses to hosts, and setup new subflows to join the existing MPTCP transmission connectivity. The path management function of MPTCP is independent from other scheduling interface and congestion control functions, this makes it feasible to redefine a smart path management algorithm with no significant changes to other functional components [21, 22].

MPTCP can be set to backup mode or single mode; in backup mode, the system setup a TCP subflow over each interface

and sets the cellular interface as a backup path. Traffic flows only over the WiFi interface and if it fails, then the data transfer switches to the cellular interface. In single-path mode instead, a subflow is created over the WiFi interface and no packet is sent over the cellular interface until the WiFi interface goes down. The MPTCP path manager options are currently *ndiffports* [6] and *fullmesh* [17]. Ndiffports opens multiple subflows between the same IP pairs on both end-hosts, and fullmesh opens a full mesh of subflows among the available IPs. This mode creates multiple subflows for each pair of source-destination IP-address pair, up to the maximum limit of allowed subflows. This policy leads to a large number of established subflows and ignores the benefits offered by the path diversity.

III. Multi-Path TCP and Multi-Armed Bandits

Since our solutions uses Contextual Multi-Armed Bandits approach to MPTCP, in this section we give some background of this approach. In the rest of the paper, we will use notations and definitions highlighted in this section. Multi-armed bandits (MAB) are a set machine learning algorithms that make decisions under uncertainty by balancing between exploration and exploitation of several agents (bandits). A MAB algorithm defines a set A of possible actions, known as "arms". At each round T, the algorithm selects an action and collects a reward for that chosen arm. Unlike reinforcement learning, bandit problems only observe the outcome of a selected action for a given state. Each MAB algorithm solves the exploration / exploitation tradeoff with the help of a policy π. For each round $t \in [T]$, the algorithm observes a context x_t, picks an arm a_t, and experience a reward $r_t \in [0, 1]$, whose value depends on the context x_t and the chosen action a_t [23, 24]. The contextual bandit policies are greedy, that is, they only take into account short term effects.

The context affects how a reward is associated with each bandit, so as contexts change, the model should learn to adapt its bandit choice. Contextual bandits have valuable statistical properties, such as regret guarantees. A good policy allows the choice of different good actions in different contexts. The policy explores by trying with various actions to understand what reward is given by each of them, then it exploits the collected information so far and take actions which maximizes an immediate reward.

IV. Automating MPTCP Path Manager Decisions

Analyzing existing MPTCP path manager mechanisms, we found two suboptimalities in regard with the protocol design and its deployment compatibility for heterogeneous networks: (1) adaptability and (2) autonomy. The classical methods of path management rely on static and predefined rules, lacking the ability of adapting to the rapid network conditions, especially at network edge. Moreover, the latency and link capacity of mobile network is notoriously variable, depends on the radio signal characteristics and on the number of concurrent connections. With an active online path manager algorithm, we improve the MPTCP path selection criteria by actively learning the evolution of the paths.

In our design, we employ a contextual bandit algorithm, a machine learning technique that generates path management decisions for MPTCP to mitigate the above issues. To make its decision, our algorithm uses real dataset collected from different locations. MPTCP leads to better user experience and higher throughput if the primary path was selected correctly. In our design, we aim at answering two network transmissions related questions: (i) what is the benefit of following an online MPTCP path management decision scheme, with respect to a predefined rule? (ii) What is the added value of feeding back the observed physical layer contexts i.e., states, for every link that a packet has traversed?

V. Our Solution: MPTCP Path Manager via Bi-Armed Bandit

In this section, we describe the details of our proposed dynamic path management algorithm for MPTCP. Our MPTCP path manager leverages the wireless signal contexts awareness to decide the best primary interface as recommended by the contextual bandit algorithm, either LTE mobile network or WiFi. Each arm is an MPTCP session with both LTE and WiFi connections to be set up. In our solution the model assigns the primary path based on the given contexts of the wireless parameters, unlike the default fullmesh path manager that always establishes WiFi first despite the signal quality. The online contextual bandit module is integrated with the mobile device to obtain the features for the active learner module. Such module makes the primary path decision based on the emulated wireless environment. As the currently default approach in MPTCP, the proposed contextual bandit path manager creates a fullmesh of subflows among all available flows, but it starts with the recommended first primary that helps maximizing the throughput and utilizing the available network resources efficiently.

The design of a path management mechanism for MPTCP can be represented as a machine learning task to generate optimal path decision rules under uncertain network conditions. The contextual bandit provides the methodology for an agent to learn by interacting with the dynamic network environment and produce actions that maximize the reward. We adopt the contextual bandit to find the optimal primary path for MPTCP under heterogeneous networks as an online active learner. A typical contextual bandit involves the concepts of context, agent, arm (action), and reward, which are defined as follows.

1) ***Contexts:*** The contexts module is responsible for parsing the signal data from the user environment, obtaining the contextual features that are continuously observed from the users devices by the agent. In our MPTCP system, the contexts $x_k^t : t = rounds,\ k = arms$ represent the dynamic LTE and WiFi radio signal characteristics as described in Table I.
2) ***Agent:*** agent is the core entity of the system that performs the learning and recommends the actions. In our MPTCP path management problem, the agent is the component responsible for making decisions of primary path according

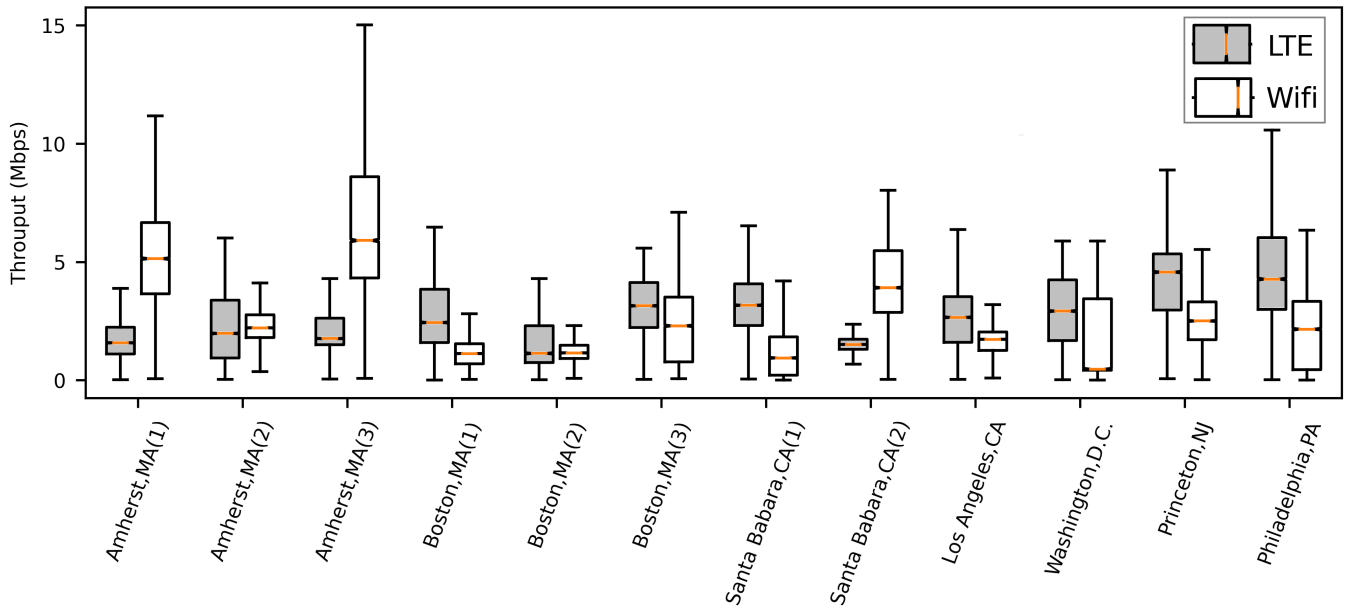

Fig. 2. MTCP throughput for coexisting WiFi and LTE in all the locations given by the dataset, having different proportion of WiFi and LTE throughputs with uplink and downlink combined.

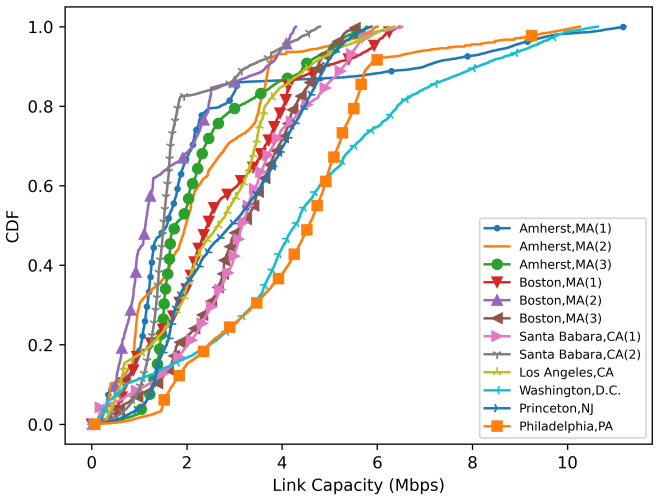

Fig. 3. CDF of link capacity for all locations where MPTCP measurements were conducted having a diverse total link capacity for both LTE and WiFi throughput.

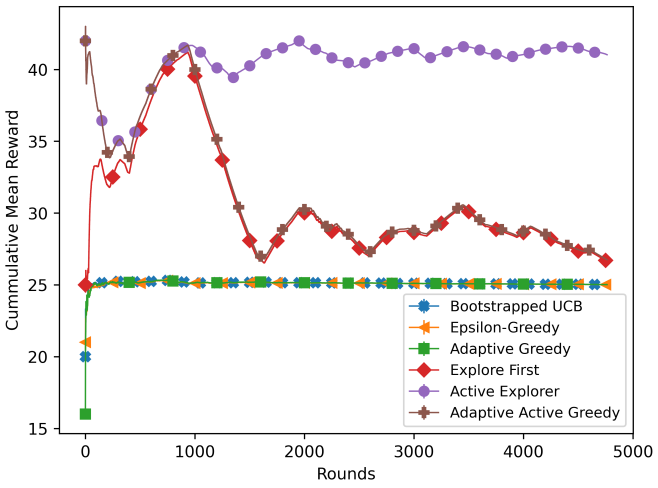

Fig. 4. The mean cumulative reward (and its error upto 95% confidence level) is calculated for each policy over its 50 batch online simulations.

to different wireless network conditions. Those decisions in contextual bandit algorithm are made by the policy and will be discussed in section V-A.

3) ***Arms or Actions:*** The arms in the bandit setting indicate how an agent responds to the observed context. In MPTCP path management, the arms determine the number of possible paths or subflows in the network topology. This work is concerned with a scenario of fixed number of arms $k = 2$, from which an agent must choose one as his action a_t in each round t. Each arm in our problem represents the Radio Access Technology (RAT), LTE or WiFi.

4) ***Reward:*** The reward is the long-term overall benefit that agents wish to maximize. We assign stochastic binary rewards for each arm $r_k^t \in \{0, 1\}$ using the Bernoulli distribution function, and is concerned with cumulative rewards throughout the rounds. The reward r_k^t equals one when the agent observes a throughput and latency above the threshold values, and 0 otherwise.

TABLE I
CONTEXTUAL FEATURES OF THE DYNAMIC LTE AND WIFI RADIO SIGNAL CHARACTERISTICS

WiFi and cell RTT	Allow devices to measure the distance to nearby WiFi routers and determine their location with a precision of 1 to 2 meters.
WiFi DNS lookup time	Reflects how long it takes the DNS servers to respond to a request for the domain. It identifies the DNS hosting performance delays while resolving the domain
WiFi RSSI	Received Signal Strength Indicator, measures how well the device can hear a signal from an access point or a router. It determines if there is enough signal for getting a good wireless connection
WiFi linkspeed	Link speed is the maximum speed in bits per second. For wireless connections many factors affect the link speed, such as wireless standard, WiFi signal strength, and level of interference.
WiFi and cell lossrate	The percentage of frames that should have been forwarded by a network but were not. It is the ratio of the number of lost packets to the total number of sent packets. It expresses the reliability of the cell and WiFi paths in the network
WiFi UDP delta	the variation of network delay computed by time delta between two frame timestamps.
Cell Signal Strength	Signal Strength measured in decibels and indicates the signal received from a cell tower. Represents the entire received power including the wanted power from the serving cell as well as all co-channel power and other sources of noise.
Cell dBm	Represent the strength of a signal at any given location, as well as the amount of power an antenna is capable of amplifying. Measures the average power received from a single Reference signal.
Cell RSRQ	Reference Signal Received Quality indicates quality of the received signal and ranges from -19.5dB (bad) to -3dB (good)). It is a key measure of signal level and quality for mobile network.
Cell CQI	Channel Quality Indicator, carrying the information on how good or bad the communication channel quality is. It is the information that the user equipment sends to the network about current communication channel quality.
Cell RSSNR	Reference Signal Signal to Noise Ratio, level of a desired signal to the level of background noise.

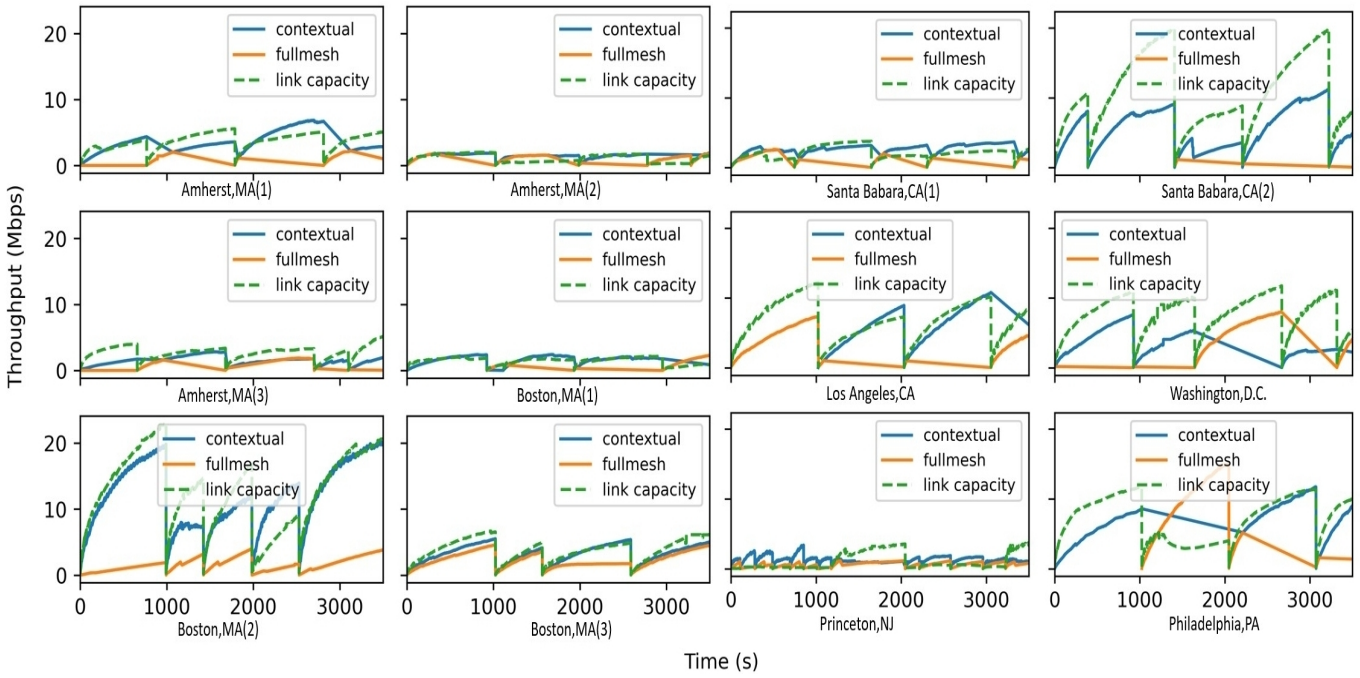

Fig. 5. MPTCP throughput over time, measured during the evaluation of a rapidly changing wireless network and links capacity, for our contextual bandit and the fullmesh MPTCP path management. Contextual bandit path manager maximize utilization of available resource within given capacity limit

A. Contextual Bandit Policies Evaluation

The policy is implemented in the agent and it defines a mapping from the observed contexts of the network environment to the action to be taken in those contexts setup, policy $\pi : (\ context\ x) \ \mapsto (\ action\ a)$. The policy helps the agent choose actions based on contexts by finding a good decision rule for selecting the action. We use a policy that involves machine learning algorithm as an oracle to learn from past information and choose optimal actions based on learnt response patterns. In our work, this oracle fit a linear logistic regression classifier model with Stochastic Gradient Descent (SGD).

Before determining the contextual bandit strategy to deploy, we compared six different policies that differ in their exploring and exploiting strategies. The policies are detailed here below:

1) *Epsilon-Greedy.* This policy chooses the empirical best arm a, based on the classification oracle $\hat{f}(x^t)$, with some given high probability p or a random arm otherwise. Several variations of this explore-exploit dilemma have been proposed for this policy; for example, decaying the probability of choosing a random arm with each successive round, or even dropping it to zero after some turning point.
2) *Bootstrapped Upper Confidence Bound (UCB).* This policy obtains an upper confidence bound by considering the percentile of the predictions among a set of k classifiers $\hat{f}_{1:k,1:m}$ that fit with different bootstrapped samples m per arm.
3) *Adaptive Greedy.* This policy sets a threshold value z on the classification oracle $\hat{f}_{1:k}$ predictions then pick the action a with maximum value of that at each successive round, or else choose either randomly or according to an active learning heuristic.
4) *Explore First.* This policy is also known as "explore then exploit". The policy uses a breakpoint t_b, for each successive round t with context x^t, and it selects action a uniformly at random if $t < t_b$ to explore, or else exploits by selecting the action with maximum reward value $a = argmax\ \hat{f}(x^t)$. This reward a is hence estimated to be the best arm.
5) *Adaptive Active Greedy.* This policy sets an initial threshold z_0 on which it takes action with highest estimated reward, otherwise it takes an action either randomly or based to an active learning heuristic.
6) *Active Explorer:* Selects a proportion of actions according to an active learning heuristic based on gradient. The predictions are made according to an active learning heuristic (the gradient that the observation would produce on each model predicting a class. Since the active explorer policy can control the probability p of selecting an action a according to active learning criteria w_q, it is a good candidate for the rapidly changing network condition contexts. The learning heuristic of this policy is explained as part of Algorithm 1

Algorithm 1 Primary Path selection for User Equipment in any edge based server using contextual multi-armed bandit based on active explore policy.

Input: probability p, classification oracles $\hat{f}(x^t)$ with gradient functions $g(x, r)$
Output: The primary path selection for each user equipment MPTCP new connection attempt

repeat
Collecting the contexts data from the last mile radio environment of LTE and WiFi x^t
for *each successive MPTCP connection attempts t with contexts x^t* **do**
  With probability p:
  Select action $a = argmax\ \hat{f}(x^t)$
  Otherwise:
  for *arm q* **do**
    Set $w_q = (1 - \hat{f}_q(x^t)||g_q(x^t, 0)|| + \hat{f}_q(x^t)||g_q(x^t, 1)||$
    Select action $argmax\ w$
  end
end
Until obtained reward r_a^t, update observation $\{x^t, r_a^t\}$ to the history for arm a, update classification oracle $\hat{f}_a$
return The selected primary path for every connection based on online last 50 trained and updated batch

VI. Performance Evaluation

A. Experimental Setup

To evaluate our proposed contextual bandit MPTCP solution, we implemented the system described in Fig. 1. Our implementation is based on on Kheirkhah's MPTCP implementation in Network Simulator 3 (NS3) [25]. We have implemented the MPTCP transport layer conforming to the RFC-6824, and following the MPTCP Linux kernel design [26]. We integrated the online contextual algorithm with NS3 running the NS library to control the MPTCP primary path selection by modifying the MPTCP path manager module. To quantify the benefits of our approach, we evaluate the performance using real data collected by the NMS lab at MIT CSAIL [27] using a crowd-sourced network measurement tool. The dataset refer to end-to-end measurements of WiFi and cellular network performance on the user equipment, simultaneously. The dataset has different properties. In particular, analyzing the dataset (Figure 3) we found: (i) always a dominant arm to which most observations belong in few locations, (ii) a more balanced scenario with no dominant label, (iii) a large number of labels useful for training if the right hyper-parameters were set.

1) Network Topology: We simulate a heterogeneous network with WiFi and LTE coexisting links, illustrated in Figure 1. The network consists of an MPTCP compatible user equipment and an internet MPTCP compatible server. We emulated the WiFi and LTE link capacity and all other radio channel properties according to the real dataset to evaluate the performance of the proposed contextual MPTCP path manager in 20 different locations provided in the dataset. Each location has divert and dynamic WiFi and LTE coverage throughout time as shown in Figs. 2 and 3.

2) Baseline Algorithms: We compared the performance of MPTCP with our contextual bandit path manager approach and the default MPTCP fullmesh path manager, using two congestion control algorithms: the decoupled (TCP reno) and the coupled congestion control [28]. We also compared our approach with the most similar and recent MPTCP work [14] where the authors used Support Vector Machine (SVM) to decide (based on a throughput threshold) whether to select both paths for LTE and WiFi or the single path with the highest bandwidth.

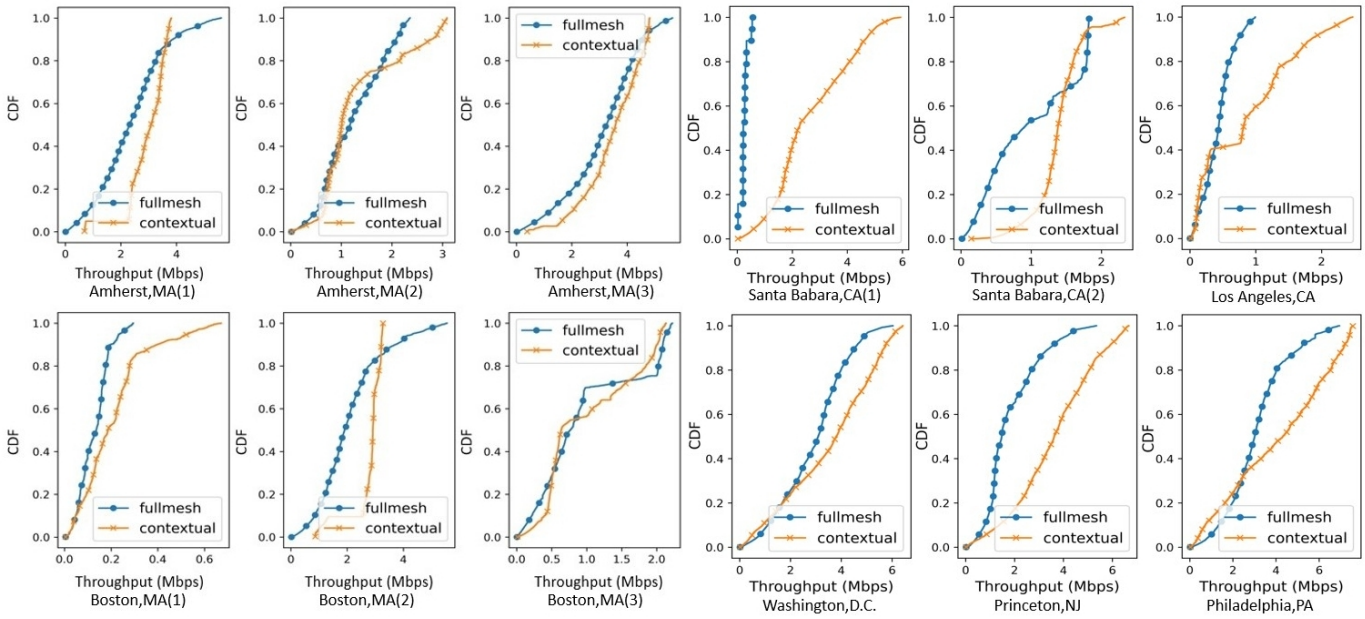

Fig. 6. CDF of the throughput of contextual bandit vs. fullmesh path manager of MPTCP for the 12 simulated locations. The throughput of contextual bandit approach is higher at a rate of around 50% of the times in average for all locations.

B. Performance Metrics and Analysis

We quantified the performance gains of our approach across two metrics: throughput and latency, arguably the most important network performance metrics. Throughput measures the total number of bytes within a unit of time, and latency indicates the delay of the network.

1) Performance of the trained contextual bandit policies: We apply a few multi-armed bandits policies to the online contextual bandits with Bernoulli rewards and binary classification using Stochastic Gradient Descent (SGD) [29]. We implemented the active learning policies to capture the fluctuating behavior of the network environment at the edge of the network. We trained the decision maker model on the full dataset including all locations in a centralized way. This approach can, however be easily ported at each edge where each location trains only with a subset of the data.
We are particularly interested in selecting the policy that maximizes our cumulative reward for each 50 online batch training representing the new MPTCP connections. Figure 4 illustrates the average cumulative reward behavior for all policies with respect to the training time. In our settings, we found that the overall best policy is the contextual active explorer, in terms of both performance and learning speed. This is due to its ability to adapt to the rapid network channel changes and its property of actively exploring probabilities against the variant contexts. Moreover, the policy considers also the characteristics of the network in its last-mile. Figure 4 also shows that the active explorer policy converged faster after 1000 training rounds. For these reasons, the rest of the evaluation experiments are carried using the active explorer policy.

2) Comparison with baselines: We compared our contextual bandit path manager, using active explorer policy, with the default fullmesh path manager. We plotted the results across different locations, each having different radio network behavior. Figure 5 shows the throughput over time as depicted by our NS3 simulation experiments for the default MPTCP with fullmesh path manager and our contextual path manager MPTCP using the same decoupled congestion control on both. The physical properties of radio propagation such as loss rate and signal power eventually cause changes in link performance. As a result, wireless radio channels fluctuate rapidly over short timescales (milliseconds) and change more dramatically over slightly longer time scales. Thus, such instabilities motivate adaptive exploration mechanisms. We observed that contextual path manager helps the MPTCP to outperforms and manages to adapt faster to the rapid network capacity changes. It managed to achieve the highest possible throughput while preventing packets to exceed the link capacity threshold value and thus invoke congestion avoidance procedure.

Fig. 6 show the CDF throughput comparison between contextual and fullmesh path manager of MPTCP for all locations given in the dataset. The overall throughput in those locations is higher using the contextual MPTCP since it will assign to each user the optimal primary path of a given location, based on the wireless signal characteristic at that time. Due to the fluctuating behavior of wireless signals, the fullmesh path manager MPTCP algorithm does not perform well, since it always initiates any new connection with WiFi despite the network environment.

In Figure 7 we present the results of our trace driven experiments for our contextual bandit MPTCP path manager against the baseline algorithms, fullmesh MPTCP path manager and

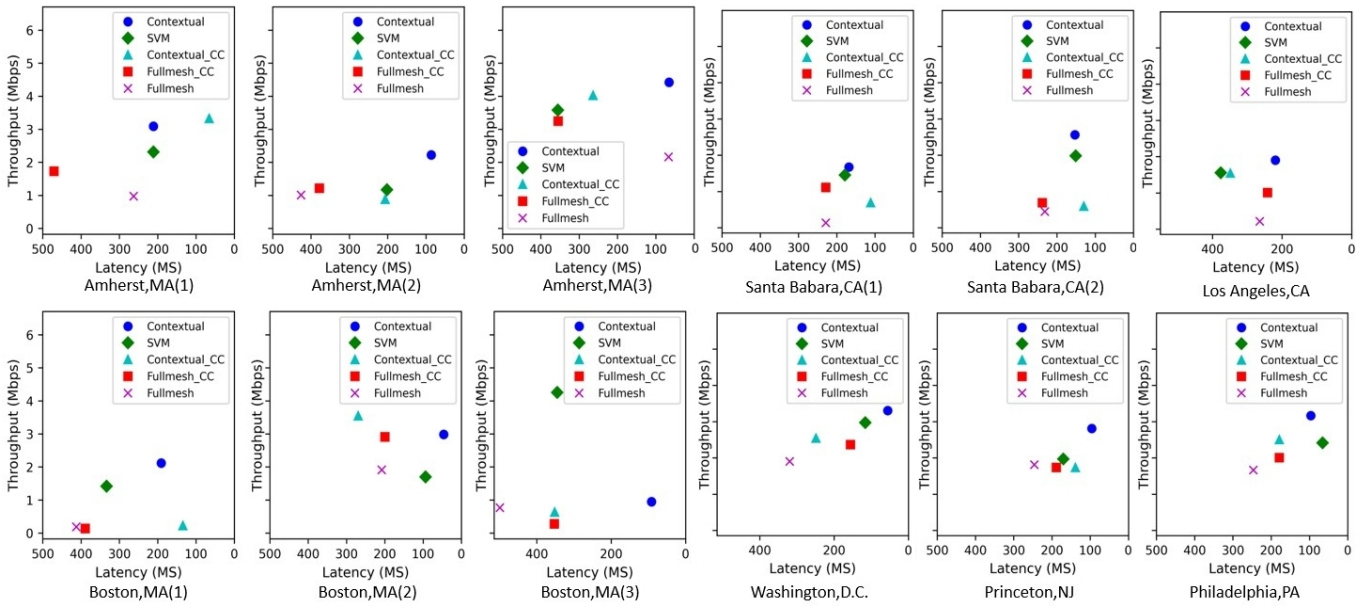

Fig. 7. WiFi and LTE mean throughput vs mean latency over the traced simulated links. The five represented points are: our contextual bandit MPTCP path manager tested with decoupled TCP Reno congestion control, the algorithm of Zhao et al. [14] using SVM to decide the primary path, our method tested with coupled MPTCP congestion control, the default fullmesh MPTCP with coupled congestion control, and the default fullmesh MPTCP with TCP Reno congestion control. The Top-right part of the graph indicate better performance.

the related work of Zhao et al. [14]. The figures show 12 charts for different locations given in the dataset with both downlink and uplink directions. On each chart plot one point per MPTCP path manager type with two congestion control mechanisms, decoupled and coupled. The points correspond to measured average throughput and latency combination, high throughput and low latency are the preferable properties for all (up and to the right).

We found that the contextual bandit path manager had almost lowest or close to lowest latency among all others and across all locations. On average, our contextual bandit path manager almost always outperformed the other baseline algorithms in terms of throughput as well. Finally, we observe that our contextual path manager along with a good congestion control algorithm react well together to achieve higher (the highest among all tested solution) throughput for any given link capacity.

VII. Related Work

The Multi-Path TCP (MPTCP) was introduced by the IETF as an extension for TCP to allow exchanging data via multiple paths concurrently and to maximize network resource utilization [30]. Several efforts have been carried out to improve the MPTCP performance in different applications [14, 31–34]. The throughput of MPTCP relies extensively on its path management mechanism and path characteristics [18, 35]. Typical MPTCP path manager employs a fullmesh mechanism to setup subflows between all available pair of interfaces [17]. Raiciu et. al [16] introduced the Ndiffport path manager, specifically to improve datacenter performance and robustness with MPTCP. With regard to mobile devices, a full mesh might be undesirable as cellular interfaces require energy and are unstable. Paasch et al.[36] proposed the backup and the single path mode alternatives to the full mesh path management for mobile devices, they all establish the WiFi connection first. Deng et al. [18] studied the impact of network selection; their experiments show that the primary selected network path has a major effect on the network throughput degradation. Similarly, we have proposed a proactive path manager, but our solution triggers the establishment of the optimal primary subflow first depending on the radio signal conditions at the connection setup time.

A few solutions have been proposed employing machine learning withing the MPTCP path management. Zhao et. al [14], for example, presented a Support Vector Machine model for MPTCP performance prediction using an SDN controller to monitors and adjusts the paths. Ahmad et. al [37], used reinforcement learning for MPTCP in multicast and wireless scenarios. Xu et. al Xu et al. [20] presented a deep reinforcement learning based architecture for MPTCP by defining the state of the environment using throughput, delay, and jitter. Similar to these approaches, we also used a machine learning in general, a simplified version of reinforcement learning in particular, and we also evaluated LTE and WiFi with the objective of maximizing the throughput of the flows in the past time epoch. However, we used a multi-armed bandit approach. None of the previous solutions considered the problem of primary path selection to tackle the path diversity issues. Finally, multi-armed bandit has been considered to solve other wireless network problems recently [38, 39].

Differently from these existing solutions, our proposed contextual bandit path manager is a learning-based primary path

selection approach with a particular focus on path diversity for heterogeneous networks. We adopted the contextual bandit learning framework to generate the optimal primary MPTCP path selection decisions efficiently.

VIII. Conclusion

This work sought to solve an important concern for Multi-Path TCP: how to automatically decide the primary path for MPTCP connections to deal with the performance degradation caused by rapid wireless signal fluctuations in heterogeneous edge networks. We designed an efficient MPTCP path manager selection strategy for LTE and WiFi. In particular, we proposed a new MPTCP path manager module that employs an online contextual bandits algorithm with binary rewards. Our prototype implemented with Network Simulator 3 (NS3) consists of two novel components: i) An online contextual bandit algorithm using Stochastic Gradient Descend classification as an oracle to decide the optimal primary MPTCP path for each new connection, and ii) a patch to the MPTCP protocol that allows overwrites to the path manager module.

IX. Acknowledgements

This work has been partially supported by NSF under Award Numbers CNS1647084, CNS1836906, and CNS1908574. The work of Aziza Alzadjali was conducted while at Saint Louis University.

References

[1] A. Sacco, M. Flocco, F. Esposito, and G. Marchetto, "An architecture for adaptive task planning in support of iot-based machine learning applications for disaster scenarios," *Computer Communications*, vol. 160, pp. 769 – 778, 2020.

[2] A. Sacco, F. Esposito, G. Marchetto, G. Kolar, and K. E. Schwetye, "On edge computing for remote pathology consultations and computations," *IEEE Journal of Biomedical and Health Informatics (J-BHI)*, vol. PP, July 2020.

[3] A. Sacco, F. Esposito, and G. Marchetto, "Rope: An architecture for adaptive data-driven routing prediction at the edge," *IEEE Transactions on Network and Service Management*, vol. 17, no. 2, pp. 986–999, 2020.

[4] A. V. Ventrella, F. Esposito, and L. A. Grieco, "Load profiling and migration for effective cyber foraging in disaster scenarios with formica," in *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*, 2018, pp. 80–87.

[5] S. Barré, C. Paasch, and O. Bonaventure, "Multipath tcp: from theory to practice," in *International Conference on Research in Networking*. Springer, 2011, pp. 444–457.

[6] C. Raiciu, C. Paasch, S. Barre, A. Ford, M. Honda, F. Duchene, O. Bonaventure, and M. Handley, "How hard can it be? designing and implementing a deployable multipath {TCP}," in *9th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 12)*, 2012, pp. 399–412.

[7] P. Eardley, Y. Nishida, and M. Kühlewind, "Multipath tcp (mptcp)," Available from https://datatracker.ietf.org/wg/mptcp/about/.

[8] N. Mohan, T. Shreedhar, A. Zavodavoski, O. Waltari, J. Kangasharju, and S. K. Kaul, "Redesigning mptcp for edge clouds," in *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, 2018, pp. 675–677.

[9] C. Lee, S. Song, H. Cho, G. Lim, and J.-M. Chung, "Optimal multipath tcp offloading over 5g nr and lte networks," *IEEE Wireless Communications Letters*, vol. 8, no. 1, pp. 293–296, 2018.

[10] X. Corbillon, R. Aparicio-Pardo, N. Kuhn, G. Texier, and G. Simon, "Cross-layer scheduler for video streaming over mptcp," in *Proceedings of the 7th International Conference on Multimedia Systems*, 2016, pp. 1–12.

[11] F. Kaup, M. Wichtlhuber, S. Rado, and D. Hausheer, "Can multipath tcp save energy? a measuring and modeling study of mptcp energy consumption," in *2015 IEEE 40th Conference on Local Computer Networks (LCN)*. IEEE, 2015, pp. 442–445.

[12] R. M. Chirwa and A. P. Lauf, "Performance improvement of transmission in unmanned aerial systems using multipath tcp," in *2014 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*. IEEE, 2014, pp. 000 019–000 024.

[13] R. Withnell and C. Edwards, "Towards a context aware multipath-tcp," in *2015 IEEE 40th Conference on Local Computer Networks (LCN)*. IEEE, 2015, pp. 225–228.

[14] Q. Zhao, M. Chen, P. Du, T. Le, and M. Gerla, "Towards efficient cellular traffic offloading via dynamic mptcp path configuration with sdn," in *2019 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2019, pp. 520–525.

[15] K. Nguyen, M. Golam Kibria, K. Ishizu, F. Kojima, and H. Sekiya, "An approach to reinforce multipath tcp with path-aware information," *Sensors*, vol. 19, no. 3, p. 476, 2019.

[16] C. Raiciu, S. Barre, C. Pluntke, A. Greenhalgh, D. Wischik, and M. Handley, "Improving datacenter performance and robustness with multipath tcp," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4, pp. 266–277, 2011.

[17] O. Bonaventure, C. Paasch, G. Detal *et al.*, "Use cases and operational experience with multipath tcp," *RFC 8041*, 2017.

[18] S. Deng, R. Netravali, A. Sivaraman, and H. Balakrishnan, "Wifi, lte, or both? measuring multi-homed wireless internet performance," in *Proceedings of the 2014 Conference on Internet Measurement Conference*, 2014, pp. 181–194.

[19] M. M. Rosello, "Multi-path Scheduling with Deep Reinforcement Learning," *2019 European Conference on Networks and Communications, EuCNC 2019*, pp. 400–

405, 2019.

[20] Z. Xu, J. Tang, C. Yin, Y. Wang, and G. Xue, "Experience-Driven Congestion Control: When Multi-Path TCP Meets Deep Reinforcement Learning," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1325–1336, 2019.

[21] M. R. Kanagarathinam, S. Singh, V. S. Kumar, and K. J. Moon, "S-mptcp: A smart multipath tcp controller for next generation mobile network," in *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*. IEEE, 2018, pp. 1165–1170.

[22] L. Boccassi, M. M. Fayed, and M. K. Marina, "Binder: A system to aggregate multiple internet gateways in community networks," in *Proceedings of the 2013 ACM MobiCom workshop on Lowest cost denominator networking for universal access*. ACM, 2013, pp. 3–8.

[23] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 661–670.

[24] A. Slivkins, "Introduction to multi-armed bandits," *arXiv preprint arXiv:1904.07272*, 2019.

[25] M. Kheirkhah, I. Wakeman, and G. Parisis, "Multipath-TCP in ns-3," *CoRR*, 2015. [Online]. Available: http://arxiv.org/abs/1510.07721

[26] C. Paasch, S. Barre *et al.*, "Multipath tcp implementation in the linux kernel," Available from http://www.multipath-tcp.org.

[27] S. Deng, R. Netravali, A. Sivaraman, and H. Balakrishnan, "Cell vs wifi," Available from http://web.mit.edu/cell-vs-wifi/.

[28] D. Wischik, C. Raiciu, A. Greenhalgh, and M. Handley, "Design, implementation and evaluation of congestion control for multipath tcp." in *NSDI*, vol. 11, 2011, pp. 8–8.

[29] D. Cortes, "python package contextual bandits," Available from https://contextual-bandits.readthedocs.io/en/latest/id4.

[30] A. Ford, C. Raiciu, M. Handley, S. Barre, J. Iyengar *et al.*, "Architectural guidelines for multipath tcp development," *IETF, Informational RFC*, vol. 6182, pp. 2070–1721, 2011.

[31] J. Zeng, F. Ke, Y. Zuo, Q. Liu, M. Huang, and Y. Cao, "Multi-attribute aware path selection approach for efficient mptcp-based data delivery." *J. Internet Serv. Inf. Secur.*, vol. 7, no. 1, pp. 28–39, 2017.

[32] B. Hesmans, G. Detal, S. Barre, R. Bauduin, and O. Bonaventure, "Smapp: Towards smart multipath tcp-enabled applications," in *Proceedings of the 11th acm conference on emerging networking experiments and technologies*, 2015, pp. 1–7.

[33] S. Chattopadhyay, S. Shailendra, S. Nandi, and S. Chakraborty, "Improving mptcp performance by enabling sub-flow selection over a sdn supported network," in *2018 14th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, 2018, pp. 1–8.

[34] K. Wang, T. Dreibholz, X. Zhou, F. Fa, Y. Tan, X. Cheng, and Q. Tan, "On the path management of multi-path tcp in internet scenarios based on the nornet testbed," in *2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA)*. IEEE, 2017, pp. 1–8.

[35] B. Arzani, A. Gurney, S. Cheng, R. Guerin, and B. T. Loo, "Impact of path characteristics and scheduling policies on mptcp performance," in *2014 28th International Conference on Advanced Information Networking and Applications Workshops*. IEEE, 2014, pp. 743–748.

[36] C. Paasch, G. Detal, F. Duchene, C. Raiciu, and O. Bonaventure, "Exploring mobile/wifi handover with multipath tcp," in *Proceedings of the 2012 ACM SIGCOMM workshop on Cellular networks: operations, challenges, and future design*, 2012, pp. 31–36.

[37] B. Ahmad, A. Khalid Kiani, S. Ur Rehman, Y. Huang, and Z. Yang, "Multicast multipath tcp for reliable communication in wireless scenarios," in *2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, Aug 2019, pp. 2212–2217.

[38] Y. Gai, B. Krishnamachari, and R. Jain, "Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation," in *2010 IEEE Symposium on New Frontiers in Dynamic Spectrum (DySPAN)*. IEEE, 2010, pp. 1–9.

[39] I. Colin, A. Thomas, and M. Draief, "Parallel contextual bandits in wireless handover optimization," in *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2018, pp. 258–265.