Model Predictive Control for Load Balancing

Pham Tran Anh Quang, Youcef Magnouche,
Jeremie Leguay
Huawei Technologies Ltd.
Paris Research Center, France.

Xu Gong, Feng Zeng
Huawei Technologies Ltd.
Dongguan Research Center, China.

ABSTRACT

To improve bandwidth utilization, flow aggregates are typically split over multiple paths. This demonstration shows that load balancing can be enhanced by exploiting traffic predictions. We present a Model Predictive Control (MPC) based load balancing framework that optimizes the maximum link utilization to proactively mitigate congestion.

CCS CONCEPTS

• **Networks → Traffic engineering algorithms**.

ACM Reference Format:
Pham Tran Anh Quang, Youcef Magnouche, Jeremie Leguay and Xu Gong, Feng Zeng. 2020. Model Predictive Control for Load Balancing. In *Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication (SIGCOMM '20 Demos and Posters), August 10–14, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3405837.3411383

1 INTRODUCTION

Load balancing aims at distributing large flow aggregates across multiple available paths [2]. The amount of traffic assigned to each path is determined by a split ratio that may vary over time according to traffic variations. Ideally, the split ratios should follow the demand. However, in practice, re-configurations can only be realized at a slow pace, and they must be modified incrementally to avoid degrading the QoS of ongoing flows. Consequently, the prediction of future traffic may help in smoothly updating split ratios to follow the evolution of traffic. Along these lines, we propose to demonstrate that Model Predictive Control (MPC) can be applied to UCMP load balancing in centralized TE architectures, and it outperforms greedy updates of split ratios (not using predictions) and the legacy approaches ECMP.

The prediction of future traffic can be done using historical data. Its accuracy, nevertheless, depends on the prediction horizon, the traffic pattern, and the algorithm design. One way to tackle prediction errors in MPC is assigning discount factors to further predictions, thus reducing their impacts on the load-balancing decision. Besides the prediction horizon, the control horizon, determining how frequently we change the split ratio, can also impact the load balancing performance. In this demo, we investigate the impact of the prediction horizon, the control horizon, and the types of discount to load balancing.

2 MPC FRAMEWORK

We consider a centralized TE controller that can modify split ratios periodically or when congestion happens. The controller has several origin-destination (OD) tunnels to manage, each one consisting in an aggregate of many micro-flows. At each optimization step, new target split ratios are computed based on traffic predictions and distributed to edge devices. New micro-flows arriving at ingress devices are placed on the candidate paths such that the difference between the actual and the target split ratios is minimized.

Problem Formulation. Given a network with a set of nodes V and a set of links A, let K be the set of tunnels and $\mathbb{T}$ the set of time-slots in the prediction horizon. For each $k \in K$ and $t \in \mathbb{T}$, let $d^k(t)$ be the prediction of the traffic demand at each time-slot and let $\mathbb{P}_k$ be the set of pre-calculated paths associated with tunnel k. Let $x_p^k \in \mathbb{R}_+^{|K|\times|\mathbb{P}_k|}$ be split ratio of traffic on path $p \in \mathbb{P}_k$ for tunnel $k \in K$ and $w^t \in \mathbb{R}_+$ the discount factor. Let $x_{max}^t \in \mathbb{R}_+^{|K|\times|\mathbb{P}_k|}$ be the Maximum Link Utilization (MLU) at time-slot $t \in \mathbb{T}$. The prediction-based load balancing can be found solving the following linear program:

$$\begin{aligned}
\min \quad & \sum_{t\in\mathbb{T}} w^t x_{max}^t \\
\text{s.t.} \quad & \sum_{p\in\mathbb{P}_k} x_p^k = 1, \quad \forall k \in K, && (1)\\
& \sum_{k\in K} d^k(t) \sum_{p\in\mathbb{P}_k, e\in p} x_p^k \le x_{max}^t C_e, \quad \forall t \in \mathbb{T}, e \in A, && (2)\\
& x_p^k \ge 0, \quad \forall k \in K, p \in \mathbb{P}_k.
\end{aligned}$$

Constraints (1) ensure that for each tunnel, all traffic is split on the associated paths and Constraints (2) permit to compute the maximum link utilization for each time-slot. Additional constraints can also be added to enforce smooth variations of split ratios over time.

Traffic predictions. We consider an ideal predictor and a Kalman Filter predictor. For the ideal predictor, we assume the predictor knows traffic demand in advance. The Kalman Filter predictor makes predictions based on historical data, and its accuracy depends on the prediction horizon.

3 DEMONSTRATION

We use NS3 simulator [3] with Open Flow 1.3 module [1]. Applications in NS3 are generating the traffic pattern following the traffic of Tencent's video streams. The transport layer is TCP. The microflow inter-arrival time varies to generate diurnal pattern traffic. As Fig. 2 shows, the topology is an SD-WAN network where 3 branches are

---

SIGCOMM '20 Demos and Posters, August 10–14, 2020, Virtual Event, USA

ACM ISBN 978-1-4503-8048-5/20/08.
https://doi.org/10.1145/3405837.3411383

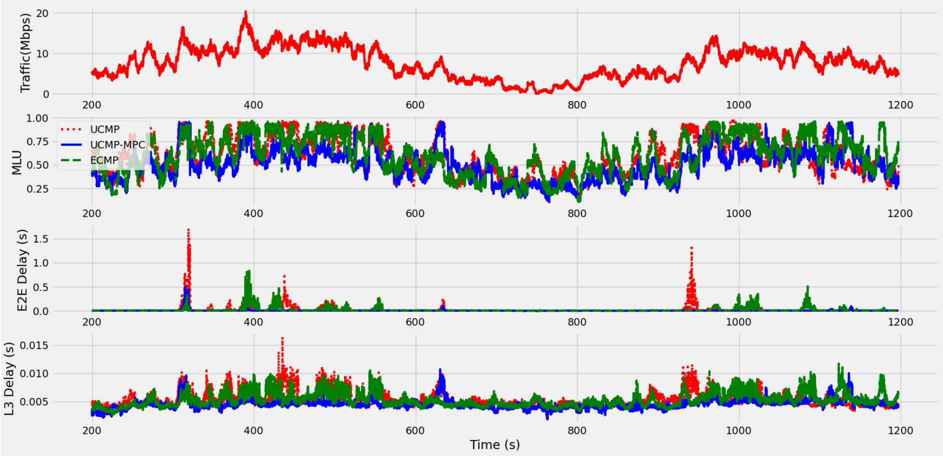

Figure 1: Total traffic over time. QoS factors over time for ECMP, UCMP without prediction and MPC-based UCMP.

connected to a headquarter site using both MPLS and broadband Internet. In this particular scenario, when split ratios are updated every 50s and the prediction horizon is 50s, the average MLU for ECMP, UCMP without predictions, UCMP with ideal predictions, and our MPC-based UCMP are 0.54, 0.54, 0.43, 0.44, respectively (MLU is between 0 and 1, the lower the better).

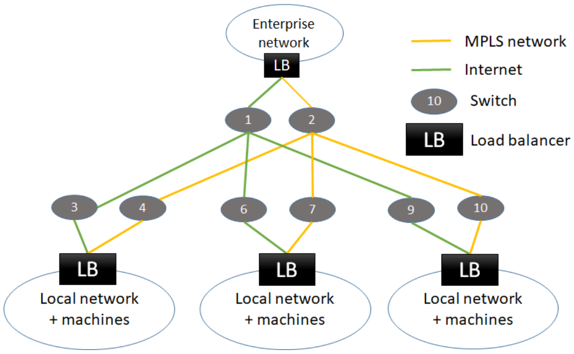

Figure 2: SD-WAN network scenario.

Fig. 1 shows the diurnal evolution over time of the total traffic (sum of the throughput for all traffic aggregates). It also shows the evolution of QoS factors such as the Maximum Link Utilization (MLU) and the end-to-end delay at both network (packet level) and application layers (socket level). Fig. 3 shows the end-to-end network delay over time. We can observe that the MPC solution avoids delay spikes and manages to proactively mitigate the congestion. It outperforms ECMP and the myopic UCMP. In another animation (in the video), we also show the evolution of the traffic for each tunnel and of the split ratios for all load balancing solutions.

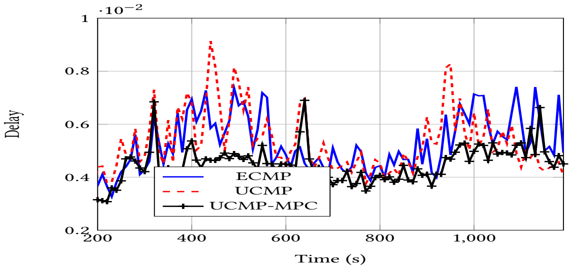

Figure 3: End-to-end delay at network layer.

Pred. Horiz.	50s	100s	300s
MLU [0,1]	0.787	0.721	0.721
Discount	None	$\mathbb{T} - t$	$exp^{-0.1t}$
Ideal	0.792	0.787	0.799
Kalman	0.82	0.786	0.773

Table 1: 95 percentiles MLU for 1) different prediction horizons and 2) different discount types (none, linear, exponential) for Ideal and Kalman predictors.

4 PARAMETER TUNING

Table 1 shows different simulation results varying the length of the prediction horizon and using different discount types (linear, exponential). We can observe that the prediction horizon should be larger than the control horizon but not too much. We also observe that the exponential discount factor helps mitigating prediction errors. Jointly with the demonstration, we will present more numerical results.

The video is available here: https://tinyurl.com/y9tjqfwu

REFERENCES

[1] Luciano Jerez Chaves, Islene Calciolari Garcia, and Edmundo Roberto Mauro Madeira. 2016. OFSwitch13: Enhancing Ns-3 with OpenFlow 1.3 Support. In *Proc. ACM Workshop on Ns-3*. New York, NY, USA.

[2] Sumet Prabhavat, Hiroki Nishiyama, Nirwan Ansari, and Nei Kato. 2011. On load distribution over multipath networks. *IEEE Communications Surveys & Tutorials* 14, 3 (2011), 662–680.

[3] George F. Riley and Thomas R. Henderson. 2010. *The ns-3 Network Simulator.*