Link-based Parameterized Micro-tolling Scheme for Optimal Traffic Management

Extended Abstract

Hamid Mirzaei
University of California, Irvine
Irvine, CA
mirzaeb@uci.edu

Guni Sharon
University of Texas at Austin
Austin, TX
gunisharon@gmail.com

Stephen Boyles
University of Texas at Austin
Austin, TX
sboyles@mail.utexas.edu

Tony Givargis
University of California, Irvine
Irvine, CA
givarigs@uci.edu

Peter Stone
University of Texas at Austin
Austin, TX
pstone@cs.utexas.edu

ABSTRACT

In the micro-tolling paradigm, different toll values are assigned to different links within a congestible traffic network. Self-interested agents then select minimal cost routes, where cost is a function of the travel time and tolls paid. A centralized system manager sets toll values with the objective of inducing a user equilibrium that maximizes the total utility over all agents. A recently proposed algorithm for computing such tolls, denoted $\Delta$-tolling, was shown to yield up to 32% reduction in total travel time in simulated traffic scenarios compared to when there are no tolls. $\Delta$-tolling includes two global parameters: $\beta$ which is a proportionality parameter, and $R$ which influences the rate of change of toll values across all links.

This paper introduces a generalization of $\Delta$-tolling which accounts for different $\beta$ and $R$ values on each link in the network. While this enhanced $\Delta$-tolling algorithm requires setting significantly more parameters, we show that they can be tuned effectively via policy gradient reinforcement learning. Experimental results from several traffic scenarios indicate that Enhanced $\Delta$-tolling reduces total travel time by up to 28% compared to the original $\Delta$-tolling algorithm, and by up to 45% compared to not tolling.

KEYWORDS

Micro-tolling; Policy Gradient; Reinforcement Learning

ACM Reference Format:


PROBLEM DEFINITION AND BACKGROUND

This paper considers a scenario where there is a set of agents that should be routed across a traffic network. Agents are assumed to be self-interested, i.e., they choose the path with minimum cost. The cost of a path is the path’s latency plus the tolls paid on that path. Since toll values might change over time, agents are continually looking for the optimal route and may change en route.

The system manager should assign tolls such that the total travel time is minimized while each agent maximizes its own self-interest. Formally, the micro-tolling assignment problem is defined as follows.

Given: link latencies measured at current time step $i$.

Output: the toll values for each link that should be applied at the next time step.

Objective: Minimize total system travel time.

Assumption: Agents are self-interested. This means that each
agent chooses the route that is at its best interest (minimal travel time).

The methods presented in this paper are based on two existing algorithms: \( \Delta \)-tolling, and policy gradient RL. In the following subsections, each of these algorithms are described briefly.

## 2.1 Delta-tolling

It has been previously proven that charging each agent the marginal cost, i.e. the cost it inflicts on other agents, leads to optimal system performance [8]. However, calculating marginal cost in real-world scenarios is not feasible. In [10, 11], a model-free method, i.e. \( \Delta \)-tolling is proposed to approximate marginal cost. In \( \Delta \)-tolling a toll is calculated for each link at any time step by multiplying the difference of current travel time and the free-flow travel time (\( \Delta \)) by a constant parameter \( \beta \). The actual toll applied on each link is smoothed by another parameter \( \tau \) to remove transient toll value spikes, according to the following equation:

\[
\tau^{i+1} = \tau^i + (1 - R)\tau^i,
\]

where \( \tau^i \) is assigned toll at time step \( i \).

## 2.2 Policy gradient RL

A well-known approach to learn a parameterized policy based on on-line data is to use general purpose Policy gradient RL method. There are different methods to estimate the policy gradient [7]. We have chosen Finite Difference Policy Gradient RL (FD-PGRL) [4]. Unlike other methods that need within-episode rewards or the agent should learn the policy with no domain knowledge, FD-PGRL can leverage an existing policy with a reasonable performance while making small changes to the policy parameters in order to proceed towards the optimal policy and it only uses finite differences to estimate the policy gradient.

## 3 ENHANCED DELTA-TOLLING

In this section we present the Enhanced \( \Delta \)-tolling (EA-\( \Delta \)-tolling) which is an extensions to \( \Delta \)-tolling approach introduced in 2.1. The original \( \Delta \)-tolling has two global parameters \( R \) and \( \beta \) for the whole network. EA-\( \Delta \)-tolling extends the parameter set by assigning different values of \( R \), \( \beta \) or both for each link. Therefore, the number of parameters in EA-\( \Delta \)-tolling can be up to \( 2|E| \) where \( |E| \) is the number of links. The increased number of parameters requires a feasible tuning algorithm since the tuning cannot be done in a brute-force way. We have used FD-PGRL introduced in 2.2. To use FD-PGRL, EA-\( \Delta \)-tolling is defined as a parameterized policy. The policy parameters are \( \beta \), \( R \) or both assigned for each link of the network.

While it is suggested in [11] that there is a correlation between the \( R \) and \( \beta \) parameters, no conclusion was provided regarding this correlation. Therefore, we consider three variants of EA-\( \Delta \)-tolling:

- **EA-\( \Delta \)-tolling\( \beta \)** - this variant uses a global \( R \) parameter and link specific \( \beta \) parameters (\( |E| + 1 \) parameters in total).
- **EA-\( \Delta \)-tolling\( R \)** - this variant uses a global \( \beta \) parameter and link specific \( R \) parameters (\( |E| + 1 \) parameters in total).
- **EA-\( \Delta \)-tolling\( \beta \), \( R \)** - this variant uses link specific \( \beta \) and \( R \) parameters (\( 2|E| \) parameters in total).

### 4 EMPIRICAL STUDY AND RESULTS

In all the experiments, traffic is modeled using the cell transmission model (CTM) [2, 3]. The DTA simulator [1] was used to run CTM and the simulation settings are the same as reported in [11]. Three traffic scenarios were used: Sioux Falls [5], Downtown Austin [6] and Uptown San Antonio which are available at: https://goo.gl/SyvV5m. The FD-PGRL parameters, i.e., step size, perturbation and number of policy runs at each step were set to 0.4, 0.01 and 60 respectively.

Total latency over all trips and total travel cost over all agents (social welfare) are presented in Figure 1. The values are normalized in Table 1.

The results suggest that tuning \( R \) parameter per link while having a global \( \beta \) leads to the best system performance in most cases.

### REFERENCES


