A Hybrid Evolutionary and Multiagent Reinforcement Learning Approach to Accelerate the Computation of Traffic Assignment

(Extended Abstract)

Ana L. C. Bazzan
Univ. Federal do Rio Grande do Sul (UFRGS)
C.P. 15064, 91501-970 P. Alegre, Brazil
bazzan@inf.ufrgs.br

Camelia Chira
Technical University of Cluj-Napoca
Baritiu 26-28, Cluj-Napoca, Romania
camelia.chira@cs.utcluj.ro

ABSTRACT

Traditionally, traffic assignment allocates trips to links in a traffic network. Nowadays it is also useful to recommend routes. Here, it is interesting to recommend routes that are as close as possible to the system optimum, while also considering the user equilibrium. To compute an approximation of such an assignment, we use a hybrid approach in which an optimization process based on an evolutionary algorithm is combined with multiagent reinforcement learning. This has two advantages: first, the convergence is accelerated; second, the multiagent reinforcement learning resembles the adaptive route choice that drivers perform in order to seek the user equilibrium. In short, our hybrid approach aims at incorporating both the system and the user perspectives in the traffic assignment problem. Results confirm that this hybridization accelerates the computation and delivers an efficient assignment.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent Systems*

General Terms

Algorithms

Keywords

Multiagent Reinforcement Learning, Traffic Assignment

1. INTRODUCTION

There are several ways to perform traffic assignment. One of them is based on the fact that drivers perform experimentation in order to assess the utility of a given number of alternative routes: self-interested drivers adapt their route choices for the n-th day based on the travel time (or any other utility function) of the previous days. This is the basis of the the so-called user equilibrium (UE) or Nash equilibrium. Another possibility is to compute an assignment that minimizes the average trip time for all drivers. The assignment resulting from this principle is called system optimum (SO). While in a real, congested, urban traffic network the observed flows are more likely to be close to an user optimum (though this varies greatly), traffic authorities strive to obtain the SO because this is socially more efficient. One way to achieve the SO is via traffic information and recommendation. However, because not everyone has access to such information, part of the drivers will still tend to perform adaptive route choices that lead to their individual best.

In this paper we take advantage of the fact that more and more information is being collected by some actor in the traffic system (be it a traffic authority or an independent actor such as Waze or Google traffic), and that route guidance is becoming a reality. In order to recommend routes, an assignment must be computed. It is then in the interest of the collectivity that such guidance aligns with the SO. However, since this normally means that some drivers are assigned to routes that are not in their particular interests (i.e., are far from the UE for these particular drivers), our approach is a hybrid between the SO and the UE. While fundamentally computing the SO, this approach also considers individual UEs.

In our hybrid traffic assignment method, the SO is computed via a genetic algorithm (GA). To account for drivers experimentation, some solutions of the pool are composed by routes that would have been computed by drivers themselves in their process of seeking the UE. In our case this is computed via Q-learning (QL). Because each choice is likely to affect the reward of many others, this is a typical multiagent reinforcement learning (RL) problem.

Due to lack of space we omit concepts related to the traffic assignment problem (TAP). We only remind about the complexity of computing Nash equilibria in general, and consequently that the UE is computed via approximations. Similarly, analytical methods to compute an exact solution to the SO (e.g., based on convex optimization) are not always feasible. For a discussion on this and an overview on related work, see [1].

2. METHODS

To deal with the computation of the SO, here a GA is employed, in which the objective function is to minimize the average travel time over all drivers. A solution for the TAP is the allocation of given portions of the flows to determined edges of a network. One important point that has motivated our approach is that giving recommendations simply from the point of view of the performance of the system may hit some drivers since their individual travel times may increase. These will tend to unilaterally divert to other routes. Therefore, our approach takes this into account: when assembling the population of solutions that takes part in the selection process of the GA, our approach includes solutions that are computed the

Appears in: *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015), Bordini, Elkind, Weiss, Yolum (eds.), May 4–8, 2015, Istanbul, Turkey.*

Algorithm 1 Pseudo-code for the GA+QL approach

```
INPUT: population size, elite size, mutation and crossover probabilities (for the GA); learning rate, initial value and decay rate on ε (QL)
INPUT: k shortest paths that are available as action selection for each flow
generate GA population
while generation < max_nb_generations do
    evolve (elitism, reproduction, mutation, crossover)
    // QL:
    for each trip or group of trips do
        ε-greedy action selection: action is randomly chosen with probably 1 − ε or route is the one with highest Q-value
    end for
    for each trip or group of trips do
        simulate trip, collect travel time, update Q-table
    end for
    substitute GA's worst individual by one formed by the individual route choices resulting from QL action selection
end while
```

way an individual driver would, if he would learn to select routes individually. As mentioned, QL is used for this purpose. Here, the reward of each agent is its individual travel time, not the average travel time. Each agent has to select an action from its action set. These sets are composed by k routes that are computed using a k shortest path algorithm to travel from the agent's origin to its destination. For action selection we use ϵ-greedy, starting with high exploration. To this aim we initialize ϵ with a high value, and allow it to decay smoothly by a rate d.

The computation of the SO is as follows. Each individual of the population of solutions is a set of shortest paths, one for each trip (i.e., one for each agent). Thus the length of the chromosome (that represents each solution) is the number of trips, and each position can take an integer value between 0 and $k-1$. A chromosome is, for instance, 3 7 0 6 6 4 ...0 1 (here for $k = 8$). Then, given a population containing p chromosomes, the GA evolves this population so that the average travel time is minimized. The evolution is made in a standard way with selection for crossover pairing with probability c and a mutation probability p_m, plus elitism. The pseudo-code for our approach (called GA+QL) is as Algorithm 1.

In order to illustrate the approach, a non-trivial traffic network is used, namely the one suggested in Chapter 10 of [2] (Exercise 10.1), where 1700 trips and four flows are considered: from nodes A and B to nodes B and M. Henceforth this network is referred as OW network. To compute the travel times, the following volume-delay function is used: $t_e = t_{e_0} + 0.02 \cdot q_e$, where t_e is the travel time on edge e, t_{e_0} is the travel time per unit of time under free flow conditions, and q_e is the flow using e. This simple scenario goes far beyond simple two-route scenarios that are commonly used. It captures properties of real-world scenarios, like interdependence of routes with shared edges, heterogeneous demand throughout the complete network, and it has more than a single flow.

3. EXPERIMENTS AND RESULTS

To assess the efficiency of the proposed method, the main performance measure is the same as used in [2]: travel times averaged over all trips as well as over trips in each flow.

Regarding the parameters of the GA, a population of size 100 was used, with elitism (the 5 best solutions were transferred to the next generation without change). For the remaining 95 individuals, selection was done using crossover rate $c = 0.2$ and mutation rate $p_m = 0.001$; $k = 8$.

Table 1: Average Travel Time per Flow: all-or-nothing, incremental assignment, GA-only, and GA+QL ($k = 8$, $p_m = 0.001$, $c = 0.2$, $\alpha = 0.9$, $d = 0.9$)

Flow	All-or-nothing	Incremental	GA (gen. 100)	GA+QL (gen. 100)
AL	114	75.92	78.67	70.86
AM	94	70.18	71.27	65.22
BL	98	77.96	82.43	69.58
BM	71	62.48	69.21	62.42
all	96.35	71.77	75.31	67.14

We do not show the plots of travel time along generations, but note that if the GA is not hybridized with the QL, the convergence happens much later than when the GA+QL approach is used. This can be seen at the last two columns in Table 1: at generation 100 (in case of $\alpha = 0.9$ and $d = 0.9$), travel times computed by the GA are higher than those computed by GA+QL. In particular, when all trips are considered (last line in the table), the travel time of the GA is 11% higher than that of the GA+QL. For sake of comparison, Table 1 also includes the results related to two standard methods to deal with the TAP: all-or-nothing (which disregards congestion), and the incremental method (see [2]). Results referring to the use of QL alone (not shown) are good in many cases, but there is an exploration phase where random actions are selected. Therefore it takes more time for the QL alone to reach the same performance of the GA+QL approach.

As mentioned previously, not only the average travel time over all drivers matters, but also travel times in each of the four flows. Table 1 (lines 1–4) shows these values. In particular, for flow BL, there is a gain of 16% when GA+QL is used (versus GA).

4. CONCLUSIONS AND FUTURE WORK

When recommending routes for drivers, it is interesting that these are aligned with a traffic assignment that approximates the system optimum. However, some self-interested drivers may still perform adaptive route choice at individual level, seeking to minimize their own travel time. To address this, our approach combines GA with QL, where routes that are learned at individual level accelerate the convergence of the GA. Our results show that this hybrid approach is able to find solutions (in terms of average travel time) that are better than other methods, and that this is done faster.

Future works relate to simulating the learning process of heterogeneous trips, as for instance those associated with different learning paces by their drivers, and/or, adherence to route recommendation. This is barely addressed in the literature but is important because in reality, the population of drivers is heterogeneous.

Acknowledgments

Ana Bazzan was partially supported by CNPq.

REFERENCES

[1] A. L. C. Bazzan, D. Cagara, and B. Scheuermann. An evolutionary approach to traffic assignment. In *2014 IEEE Symposium on Computational Intelligence in Vehicles and Transportation Systems (CIVTS)*, SSCI, pages 43–50, 2014.

[2] J. Ortúzar and L. G. Willumsen. *Modelling Transport*. John Wiley & Sons, 3rd edition, 2001.