Extending a Coupling Metric for Characterization of Traffic Networks: an Application to the Route Choice Problem

Thiago B. F. Oliveira, Bruno C. da Silva, F. Stefanello, Arthur Zachow, Ana L. C. Bazzan

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

Abstract. *This paper discusses the use of a coupling metric to characterize traffic networks for an agent based solution to the traffic assignment problem (we call it route choice problems). This metric is based on how often routes share edges with others and how interconnected they are. Since the choice of learning agents may interfere with each other, we extend previous works by biasing the learning process with this metrics and by characterizing other, bigger, instances of traffic networks.*

Resumo. *Esse artigo discute o uso de uma métrica de acoplamento para caracterizar redes de tráfego para uma solução baseada em agentes do Traffic Assignment Problem. Métrica essa, baseada em o quão frequente rotas compartilham arestas entre si e o quão interconectadas elas são. Já que as escolhas de agentes de aprendizagem podem interferir entre si, nós estendemos trabalho anterior enviesando o aprendizado com essa métrica e caracterizando outras instâncias maiores de redes de tráfego.*

1. Introduction

The Traffic Assignment Problem (TAP) deals with selecting routes from a set of options for vehicles in a network. Each vehicle has an origin and a destination, and the objective is to decrease the travel time. In a variant using multi-agent reinforcement learning, each vehicle is an agent and acts independently. Initially, the agents know nothing and must first explore its decision space while potentially interfering with each other with their actions. In this paper we extend previous work and analyze different networks. It is based on the biasing method for the Q table proposed on [Stefanello et al. 2016] which is used with the TAP to provide a shortcut to these agents. We analyze the network in question and the routes generated with a k shortest paths algorithm (KSP) [Yen 1971] generated for its OD pairs, and calculate a coupling metric for the routes. This metric is then used to bias the Q table of the learning agents. The bias would then make agents prefer less coupled routes then the rest. This is a twofold extension of [Stefanello et al. 2016] in which we: (i) provide a newer coupling definition which we believe is better suited for characterizing the network, and (ii) analyze further networks from the Bar-Gera repository [1].

The remainder of this paper is organized as follows. Related work is presented on section 2, section 3 introduces the TAP with multi-agent reinforcement learning. Section 4 defines the coupling metric used and the results obtained are shown on section 5.

[1]Transportation Networks for Research. https://github.com/bstabler/TransportationNetworks

2. Related Work

The TAP is a common research topic for computer science research. In this paper we focus on applying reinforcement learning techniques. An Agent based learning proposal is made in [Tumer and Agogino 2006]. A mixed approach involving the use of both reinforcement learning and an optimization heuristic is proposed in [Bazzan and Chira 2015]. [Ortúzar and Willumsen 2001] is recommended as reference for traditional approaches to the TAP.

This paper extends previous work done on [Stefanello et al. 2016] which also proposes a coupling metric which was further modified for this paper. It presents results demonstrating the use of a biased Q table with the TAP. Furthermore, a formal description of the TAP is also included.

3. The Traffic Assignment Problem With Multi-agent Reinforcement Learning

A traffic network can be defined as a graph G = (V, E) where V is the set of vertices on the network and E the set of edges. Furthermore, this network is associated with a traffic demand, i.e., the trips requirements of the users of this network. This is represented in the form of origin - destination pairs (OD pairs). This matrix establishes that at an origin vertex there will be a specified number of vehicles departing to a destination one. In Figure 1, we show the OW network from [Ortúzar and Willumsen 2001] as an example. It consists of a few vertices and edges along with 4 OD pairs (A to L,A to M, B to L and B to M). For a vehicle to travel from A to L for example it must transverse a series of edges until arriving at the destination. There are several possible routes – which differ in their lengths and travel times – to achieve this.

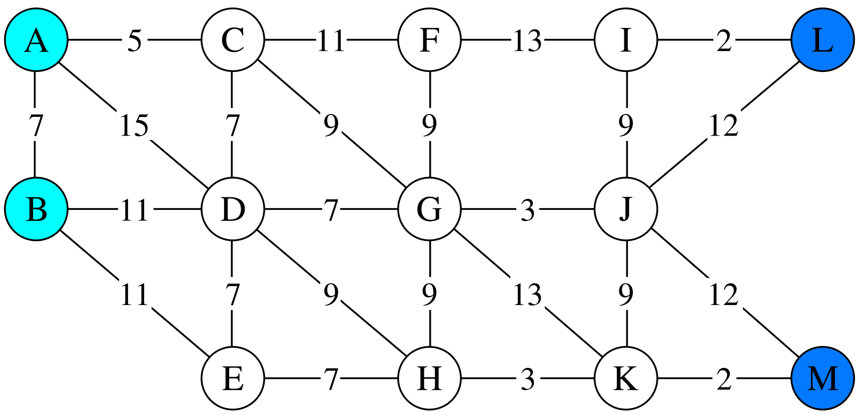

Figure 1. The OW network. Light blue vertices are origins; darker ones are destinations

Each edge has a cost to transverse given by a cost function. This is a way of representing the usage of this edge on the network. If it is congested, the cost will be greater. In this paper we consider a cost function consisting of a constant called the free flow travel time (FFTT) and a variable term that express the level of congestion on that link, i.e., the more vehicles on it, the higher the cost. The free flow travel time is defined as the time it takes for a vehicle to travel through an edge. On the edges on Figure 1, the numbers accompanying them are their respective FFTTs.

Q-learning is a reinforcement learning algorithm based on the Markov decision process. It models the decision making of agents based on rewards for its actions (in our case each route is an action). The agents keeps records of the states it has visited and the rewards it obtained previously when following each action available. At each iteration of the algorithm, the state-action pair $Q(s,a)$, where s is the current state and a is the action taken, is updated according to Equation 1. The α represents the learning rate. The higher it is, the more relevant the recently learned information will be considered. It cannot be too high or the agent will have short memory, nor too low in which case the agent would not learn much.

$$Q(s,a) \leftarrow Q(s,a) + \alpha \cdot (reward + \gamma \cdot \max_{a'} Q(s',a') - Q(s,a)) \tag{1}$$

The agent will then choose one of the possible actions according to its policy. This policy must create a compromise between random exploration of the state space and recollection, i.e., the use of the information the agent has gathered to make a choice based on what it has learned. We use the ϵ-greedy policy where the agent will explore with probability of ϵ. It is initiated with an initial value and at each iteration of the algorithm, it is decreased with a decay value δ as shown in Equation 2.

$$\epsilon \leftarrow \epsilon \cdot \delta \tag{2}$$

In this paper we consider the TAP with a multi-agent reinforcement learning approach. This implies that the decision of which route to take will be taken independently by every individual agent and not centralized.

4. Coupling Metric

When one tries solving the TAP with a multi-agent system and reinforcement learning a problem arises from the interference one agent may cause to another. A group of agents may select a route which would be faster for them, but at doing so, congesting a part of the network another agent must use.

To avoid considering every route possible for each OD pair we use the k shortest paths algorithm [Yen 1971] to restrict the available routes to the k shortests for each pair. We remark that such algorithm only considers the FFTT, i.e., it takes no congestion into consideration. The problem of deciding which route each vehicle follows while minimizing cost is called the TAP.

We propose a way of analysing a network, demands and the routes generated for them with the k shortest paths algorithm of [Yen 1971] with the intent of characterizing the coupling of the routes generated. This metric measures how similar routes are to each other. A strongly coupled route shares a lot of its path with other ones. Therefore, if agents from a OD pair share routes that are highly coupled with another pair, we believe that there is a high likelihood of these groups interfering with each other in the way of congestion, for example. We intend on using this metric to bias the Q-learning algorithm.

In this paper we define a coupling metric that is similar to the one proposed by [Stefanello et al. 2016] but has one major difference. It considers the cost of the free flow travel time instead of treating every edge as having the same cost. The coupling of the route R_i with R_j is defined as:

$$C(R_i, R_j) = \frac{|R_i \cap R_j|}{|R_i|}. \tag{3}$$

In Equation 3, $|R_i \cap R_j|$ is defined as the sum of the FFTT of the edges routes R_i and R_j share, and $|R_i|$ the sum of the FFTT of all the edges of R_i. Note that the coupling is *not* symmetric, i.e., $C(R_i, R_j)$ is not necessarily equal to $C(R_j, R_i)$ due to the value of the denominator in this equation.

Given the just defined coupling between each two routes, the overall coupling of a route is given as in Eq. 4, where σ is the total number of routes.

$$\Psi(R_i) = \frac{\sum\limits_{R_j, j \neq i} C(R_i, R_j)}{\sigma - 1} \times 100 \tag{4}$$

5. Results

In this work, our aim is to extract information from traffic networks previous to the use of the Q-learning algorithm in order to bias the Q-table and, hopefully, accelerate the convergence. In Subsection 5.1 we calculate the coupling value for a set of networks, whereas in Subsection 5.2 we compare the performance of the Q-learning algorithm with and without biasing.

5.1. Calculating the Coupling Values

As mentioned, we extend the previous work of [Stefanello et al. 2016] by proposing a modification in metric definition, as well as by applying it in additional networks. Specifically, we apply the coupling metric to a set of transportation test problems from the Bar-Gera repository [2] and the OW network proposed in [Ortúzar and Willumsen 2001] (see Fig 1).

Table 1 shows the main characteristics of each network used. These networks range form relatively small ones, like the Sioux Falls and OW, to large ones, such as the Berlin-Mitte-Prenzlauerberg-Friedrichshain-Center. Their size, as well as the demand size, is quite important because this severily affects performance since the combinatorial nature of the algorithm as seen in Eq. 4.

We remind that the coupling metric is computed for each of the k routes of each OD pair. Since some networks have a high number of OD pairs (see Table 1), it is not possible to list the resulting coupling for each route, in each OD pair. Thus, we omitted individual results and consolidated them. To give an idea of how these consolidated values are distributed, we also show histograms (see ahead).

We calculated the coupling metric for each of the networks in Table 1 using the value $k = 8$ for the KSP. The consolidated results can be found in Table 2. The KSP and

[2] Transportation Networks for Research. `https://github.com/bstabler/TransportationNetworks`

Table 1. Characteristics of the networks used

network	#vertices	#edges	#OD pairs	#origin	#dest.
Anaheim	416	914	1406	38	38
Berlin-Friedrichshain	224	523	506	23	23
Berlin-Mitte-Center	398	871	1260	36	36
Berlin-Mitte-Prenzlauerberg-Friedrichshain-Center	975	2184	9505	98	98
Berlin-Prenzlauerberg-Center	352	749	1406	38	38
Berlin-Tiergarten	361	766	644	26	26
OW	13	24	4	2	2
Sioux Falls	24	76	528	24	24

coupling algorithms ran on a Ubuntu 16.04 machine with an i7 2600K 3.4 Ghz processor and 8 GiB of RAM. Their code were implemented in Python and are available in our repositories [3].

Table 2. Consolidated coupling values for different networks

network	average	std. dev.	sum	min	max
Anaheim	2.669	0.865	30,022.208	0.267	7.072
Berlin-Friedrichshain	6.514	3.080	26,367.256	0.000	15.230
Berlin-Mitte-Center	4.976	2.558	47,930.113	0.000	14.552
Berlin-Mitte-Prenzlauerberg-Friedrichshain-Center	3.181	1.598	145,803.985	0.000	8.631
Berlin-Prenzlauerberg-Center	6.262	2.918	67,477.957	0.000	15.721
Berlin-Tiergarten	5.895	2.822	25,090.029	0.000	11.630
OW	21.510	5.074	688.322	12.797	30.787
Sioux Falls	6.614	1.020	27,936.899	2.368	9.795

In Table 2, we provide consolidated metrics of different networks. The average column represents the average of the coupling of every route of every OD pair in each network. The standard deviation is calculated from the same data. The sum is the sum of

[3] KSP implementation: `https://github.com/maslab-ufrgs/ksp` and Route coupling implementation: `https://github.com/maslab-ufrgs/routecoupling`

each coupling value. Min and max represents, respectively, the minimum and maximum coupling values for each network.

Considering the eight shortest paths calculated, the Anaheim and Berlin-Mitte-Prenzlauerberg-Friedrichshain-Center are the ones with lower average coupling. This indicates that there is a trend in these networks of more alternative paths and the interference an agent may cause to another is smaller. The fact that these networks also have the lowest maximum coupling also supports this idea.

The networks with a minimum coupling of zero, have some routes that are completely isolated from the rest and don't share any edge of the network with them. Agents following these routes will cause no interference on the rest.

The histograms in Figures 2 and 3 show the distribution of the routes on each network according to their individual coupling values. Notice how the former has a very noticeable normal distribution, while the second is much flatter. In the first case, a large subset of the routes have similar coupling values around the average. The second histogram, however, has no such convergence to a single coupling value which shows that its routes vary significantly more than in the previous case.

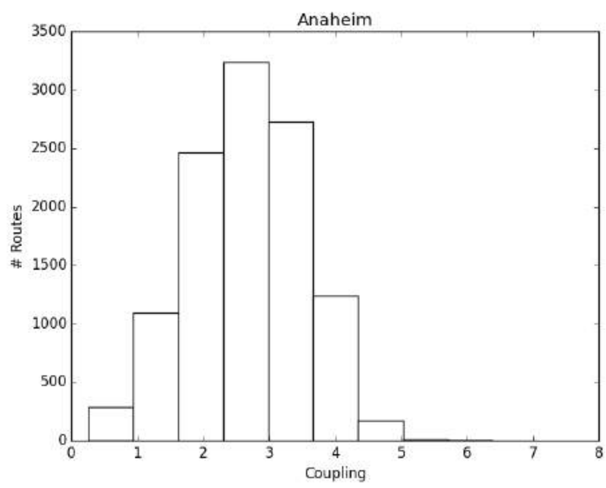

Figure 2. Coupling distribution for the Anaheim Network

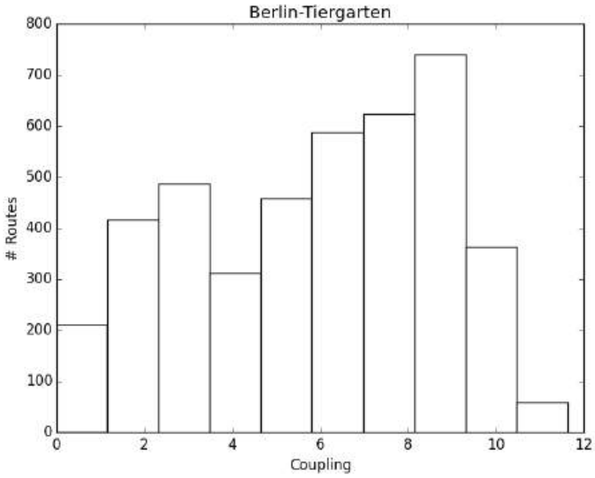

Figure 3. Coupling distribution for the Berlin-Tiergarten Network

With the coupling metric we have been able to obtain interesting characteristics of different networks. More interestingly, these networks present different properties in regards to our metric. We believe this will help to determine the level of difficulty of solving the TAP on different networks.

5.2. Biased Q-learning Performance

Using the coupling metric calculated for the OW network (Fig. 1), we tested its effectiveness as a bias for the Q table. Our results show a small, but noticeable, increase in convergence speed.

Using low alpha values in Eq. 1 to avoid overwriting the coupling information stored on the Q table too fast, we ran the simulations with $\epsilon = 0.1$ and a $\delta = 0.999$ for 1000 episodes.

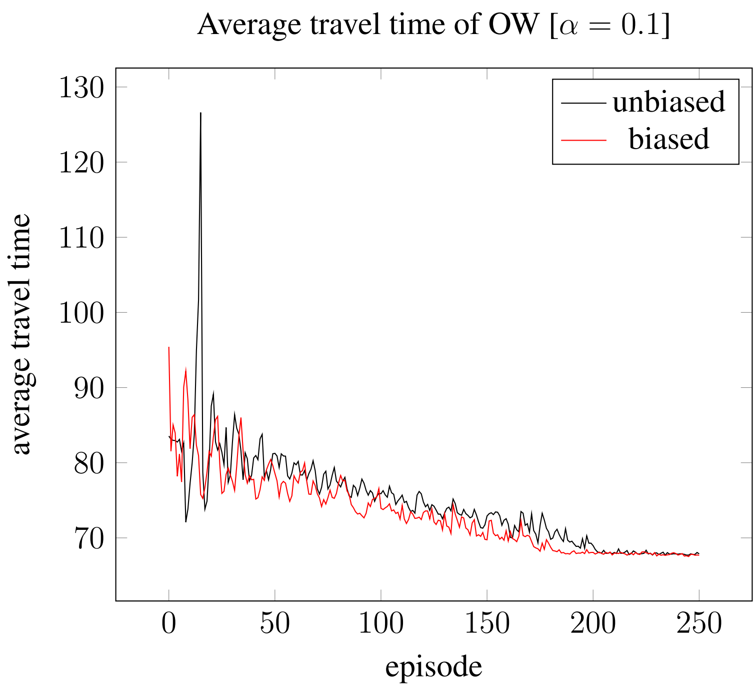

Figure 4. Average travel time at each episode of the Q-learning algorithm for the OW network and $\alpha = 0.1$

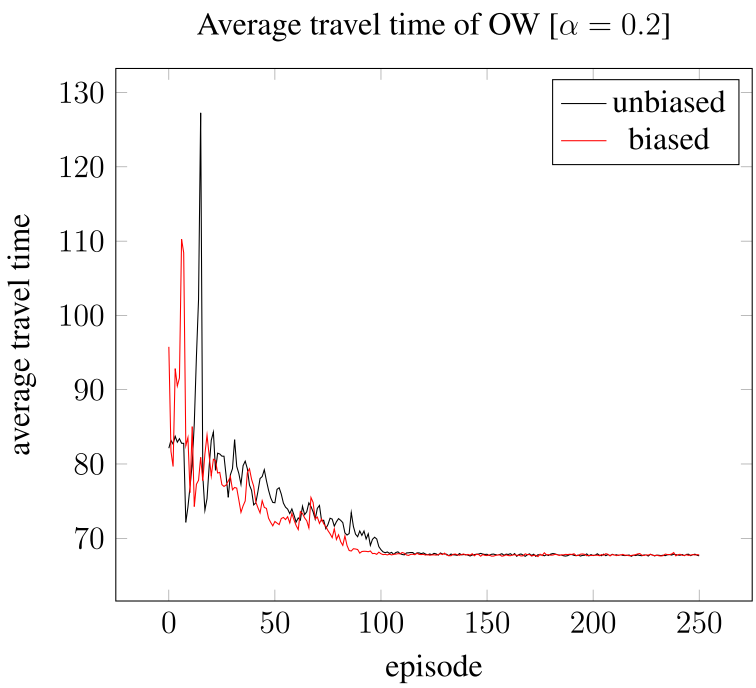

Figure 5. Average travel time at each episode of the Q-learning algorithm for the OW network and $\alpha = 0.2$

In figures 4 and 5 we present a plot of the average travel time (over all agents, a measure of how good agents are deciding) against learning episode. The x axis represents the episode number (which were truncated to 250 for better visualization). The y axis shows the average travel time of all agents.

In the first case with α equalling 0.1, the improved speed of the biased version is more noticeable. This happens because the algorithm will attribute less value to what the agents learn while experimenting and more to older knowledge such as the original Q table data. If the alpha value is increased the reverse would happen and the biased algorithm would lose its advantage.

However, the better performance of the biased algorithm could also be explained by a drop in performance in the original version since the algorithm could be ignoring newly learned information and for the unbiased case new information is much more important.

6. Conclusions and Future Work

In this paper we have proposed a novel coupling metric for traffic networks. It was devised with the intent of using it to bias the learning process of agents when solving the TAP with a reinforcement learning multi-agent system. Furthermore, we extended previous work by applying these techniques to different networks.

Initial experiments involving the use of the coupling metric for biased learning showed interesting results which call for further study. Moreover, evaluating if the coupling metric is capable of helping to determine the difficulty of solving the TAP is also to be determined with future work.

References

Bazzan, A. L. C. and Chira, C. (2015). Hybrid evolutionary and reinforcement learning approach to accelerate traffic assignment (extended abstract). In Bordini, R., Elkind, E., Weiss, G., and Yolum, P., editors, *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015)*, pages 1723–1724. IFAAMAS.

Ortúzar, J. and Willumsen, L. G. (2001). *Modelling Transport*. John Wiley & Sons, 3rd edition.

Stefanello, F., da Silva, B. C., and Bazzan, A. L. C. (2016). Using topological statistics to bias and accelerate route choice: preliminary findings in synthetic and real-world road networks. In *Proceedings of Ninth International Workshop on Agents in Traffic and Transportation*, pages 1–8, New York, USA.

Tumer, K. and Agogino, A. (2006). Agent reward shaping for alleviating traffic congestion. In *Workshop on Agents in Traffic and Transportation*, Hakodate, Japan.

Yen, J. Y. (1971). Finding the k shortest loopless paths in a network. *Management Science*, 17(11):712–716.