Notes on the Optimum Traffic Assignment Problem and Equilibrium

Nils Hallerfelt

April 2023

1 Introduction

In this short study the traffic flows on the highway network in Los Angeles will be analyzed. The graph is represented with a node-link incidence matrix B. The rows of B are associated with the nodes of the network and the columns of B with the links. The i-th column of B has 1 in the row corresponding to the tail node of link e_i and (-1) in the row corresponding to the head node of link e_i.Each node represents an intersection between highways. Each link $e_i \in \{e_1, \ldots, e28\}$, has a maximum flow capacity c_{e_i}. Furthermore, each link has a minimum traveling time l_{e_i}, which the drivers experience when the road is empty.

2 Theory

As mentioned in the introduction, different kinds of flows will be examined in a graph. To be more precise, the report will use theory about system-optimum and user-optimum flows, and highlight the connection between them when the Wardrop equilibrium is used, in particular when additional weights are put on some edges, referred to as tolls.

2.1 System-Optimum Traffic Assignment Problem

When using the theory of system-optimum traffic assignment to solve the problem of maximum flow (SO-TAP), the interest lies in finding the optimum assignment under the assumption that the system communicates internally and that all agents (drivers) altruistic in the sense that they are only concerned with how effective the overall system is without any regard to their own selfish interests.

The cost function is defined via a delay function $\tau_e(f_e)$, where f_e is a flow over the edge e. The following assumption will be useful:

Assumption 1. *For all $e \in \mathcal{E}$: τ_e is non-decreasing, in $\mathbf{C}^2$ on $[0, c_e)$, $\tau_e(f_e) = \infty$ for $f_e \leq c_e$, and*

$$2\tau_e'(f_e) + f_e\tau_e''(f_e) \leq 0 \tag{1}$$

For the SO-TAP, we define the cost function for a single edge as the product between the flow and the delay that the flow induces, that is the total delay over that edge:

$$\psi(f_e) = f_e \tau_e(f_e), \quad f_e \leq 0, \quad e \in \mathcal{E} \tag{2}$$

It is straightforward to show that assumption 1 implies the following:

Assumption 2. *For all $e \in \mathcal{E}$, $\psi_e(0) = 0$, $\psi_e(f_e)$ is non-decreasing, in $\mathcal{C}^2$ and convex in $[0, c_e)$.*

We are thus working with a convex linear program:

$$\mathcal{L}_{SO} = \begin{cases} \inf \sum_{e \in \mathcal{E}} \psi_e(f_e) \\ \text{sub. to } : f \in \mathbb{R}_+^{\mathcal{E}} \\ Bf = \nu \end{cases} \tag{3}$$

that can be simplified to the a finite-only case, under the restriction that $f_e \leq c_e$, as each $\psi(f_e)$ is finite under this assumption and there are a finite number of edges:

$$\mathcal{L}_{SO} = \begin{cases} \min \sum_{e \in \mathcal{E}} \psi_e(f_e) \\ \text{sub. to } : 0 \leq f_e < c_e, \forall e \\ Bf = \nu \end{cases} \tag{4}$$

By convexity, and under the assumption that the program is feasible, a linear programming algorithm can be deployed for solving the optimal flow of such a system.

2.2 User-Optimal Traffic Assignment

In the User-Optimal Traffic Assignment Problem (UO-TAP) a greedy behaviour of the agents are now assumed, instead of the altruistic behaviour in the SO-TAP. This means that each agent (driver, in our case), will choose the path from point A to B that is best for itself, and thus another dynamic comes into play.

To model this, the cost function becomes:

$$\psi_e(f_e) = \int_0^{f_e} \tau_e(s) ds, \quad f_e \geq 0, \quad e \in \mathcal{E}. \tag{5}$$

So the cost function over link e is no longer the total flow over that link, but rather what delay an agent would expect to experience by traversing the edge. Given an origin o and destination d, let $\Gamma_{o,d}$ be the set of all $o-d$ paths, and let $A^{o,d}$ be the link-path incidence matrix. Consider the vector z such that z_p is the aggregate flow along the $o-d$ path p. Then, by the flow decomposition theorem:

$$f = A^{o,d} z \text{ satisfies } f \geq 0, \quad Bf = BA^{o,d} z = v(\delta^{(0)} - \delta^{(d)}) \tag{6}$$

where v is the sum of all flows over the $o-d$ paths (throughput).

Definition 1. *For a given throughput $v \geq 0$, a Wardrop equilibrium is a flow vector*

$$f^{(0)} = A^{(o,d)} z$$

where z is such that $z \geq 0$, $\mathbb{1}^T z = v$, and every path $p \in \Gamma_{o,d}$:

$$z_p > 0 \implies \sum_{e \in \mathcal{E}} A_{ep} \tau_e(f_e^{(0)}) \leq \sum_{e \in \mathcal{E}} A_{eq} \tau_e(f_e^{(0)} \forall q \in \Gamma_{o,d} \tag{7}$$

That is, the total delay of choosing $o - d$ path p is less than or equal to choosing any other $o - d$ path.

This captures well what was intended: the Wardrop equilibrium $f^{(0)}$ is a flow such no agent has reason to change path under their own egoistic interests, as any other path from origin to destination will give a larger delay (How does this differ from a Nash? If i remember correctly the Braess Paradox can happen in Nashes as well?).

One can find the Wardrop equilibrium by deploying a linear program with the revised cost function, see equation 5 and 4.

2.3 Introducing Tolls

A natural question for a central authority is how the behaviour of egoistic agents can be influenced. There could be plenty of reasons for such a question. Maybe it is preferred that the agents choose paths that lead to less noise or omissions in some areas. Another is the observation that the flow one get from egoistic decision making among the agents are suboptimal compared to the system optimum, and the central authority would like a flow closer to the system optimum although the agents act on their own. One way to do this is to introduce some extra costs on the edges, we will refer to these as tolls.

Definition 2. *For a given throughput v, and a vector of tolls $\omega \in \mathcal{R}^{\mathcal{E}+}$, a Wardrop equilibrium with tolls is a flow vector*

$$f^{(\omega)} = A^{(o,d)} z$$

where z is such that $z \geq 0$, $\mathbb{1}^T z = v$, and every path $p \in \Gamma_{o,d}$:

$$z_p > 0 \implies \sum_{e \in \mathcal{E}} A_{ep} (\tau_e(f_e^{(0)}) + \omega_e) \leq \sum_{e \in \mathcal{E}} A_{eq} (\tau_e(f_e^{(0)} + \omega_e \forall q \in \Gamma_{o,d} \tag{8}$$

That is, the total delay of choosing $o - d$ path p is less than or equal to choosing any other $o - d$ path.

This generalization of the Wardrop equilibrium allows for a central authority to choose ω such that a desired behaviour is reached. To find suitable tolls to optimize the system from a system perspektive, the following theorem is useful:

Theorem 1. *For a multigraph $\mathcal{G} = (\mathcal{G}, \mathcal{E}, \theta, \kappa)$, let $o \neq d$ be two nodes such that d is reachable from o. Let each link e be equipped with a non-decreasing differentiable delay function τ_e, such that every cycle in $\mathcal{G}$ contains a link e with $\tau_e(0) > 0$. Let v in $(0, c^*_{(o,d)}$ be a feasible throughput and let $\omega \in \mathcal{R}_+^{\mathcal{E}}$ be vector of tolls. Then, an $o - d$ flow $f^{(\omega)}$ is a Wardrop equilibrium if and only if it is a solution of the network flow optimization problem*

$$f^{(\omega)} \in \underset{f \geq 0, Bf = \nu(\delta^{(o)} - \delta^{(d)})}{\arg\min} \sum_{e \in \mathcal{E}} \int_0^{f_e} \tau_e(s) ds + \omega_e f_e.$$

If f^* is the solution of a network flow optimization problem 3 with strictly convex increasing link costs $\psi_e(f_e)$, and

$$\omega_e^* = \psi_e'(f_e^*) - \tau_e(f_e^*), \quad e \in \mathcal{E} \tag{9}$$

then the corresponding Wardrop equilibrium coincides with the system optimum flow:

$$f^{(\omega^*)} = f^* \tag{10}$$

The proof needs more theory than is introduced here, but the theorem and proof can be found in Como's and Fagnani's "Lecture notes on Network Dynamics".

How is this useful? We begin by noting that the assumptions on τ in the theorem are fulfilled by a delay that fulfills Assumption 1, and the like for the cost-function ψ. Then this tells us that, if we want the Wardrop equilibrium to coincide with the system optimum, we can choose the tolls according to equation 9. Assuming the the cost function is chosen as $\psi_e(f_e) = f_e\tau_e(f_e)$, equation 9 becomes:

$$\omega_e^* = f_e^*\tau_e'(f_e^*) \tag{11}$$

so with this choice of cost function and tolls the Wardrop equilibrium with tolls coincides with the system optimum.

2.4 Specific Delay Function

Throughout the report the delay-function that will be used is the following:

$$\tau_e(f_e) = \begin{cases} \dfrac{l_e}{1-\frac{f_e}{c_e}} \text{ for } f_e \in [0, c_e) \\ \infty \text{ else} \end{cases} \tag{12}$$

This choice of τ_e fulfills assumption 1.

3 Results

3.1 Shortest Path

Given the incidence matrix B, two lists of tail-nodes and head-nodes for each edge can easily be constructed, and a graph can defined. Then, by putting weights on the edges according to the minimum travelling time l, we have a graph with positive weights. Thus Dijkstra's algorithm can be deployed to solve the shortest-path problem. The result was the following path from Vertex 1 (V_1) to Vertex 17 (V_{17}):

$$\text{Dijkstra}(\mathcal{G}, o = V_1, d = V_{17}) \rightarrow (V_1 - V_2 - V_3 - V_9 - V_{13} - V_{17}, dist = 0.5330)$$

3.2 Max Flow

Now the weights of the graph were set to the edge capacities. Then a maxflow algorithm was deployed to find the max flow from V_1 to V_{17}. Unfortunately we were not able to decide which algorithm to use here, but any of Ford-Fulkerson, Goldberg and Tarjan's Push-relabel, or Boykov-Kolmogorov can be used to solve the problem efficiently:

$$\text{maxflow}(\mathcal{G}, V_1, V_{17}) = 22448$$

3.3 External inflow

Given a fixed flow f_{fix} over the edges, we can easily determine the external flows to the system as:

$$Bf_{fix} = external\_inflow$$

as this expression sums all inflows and outflows for each node, and if the summation does not equal zero there is no conservation of mass, and this must enter the system from some external source. Here inflows are positive and outflows are negative, and the flows are represented in Table 1

Vertex:	1	2	3	4	5	6	7	8	9
Inflow:	16806	8570	19448	4957	-746	4768	413	-2	-5671
Vertex:	10	11	12	13	14	15	16	17	end
Inflow:	1169	-5	-7131	-380	-7412	-7810	-3430	-23544	-

Table 1: External inflows for each vertex in the graph. Negative inflows are equivalent with outflows. The first and second row are the node-numbers, while the second and third row are the inflows corresponding to the above rows vertex number.

3.4 SO-TAP with τ_e

In this and the following sections we assume conservation of mass in all nodes except for the source and sink, where there is an assumed inflow of 16806 and an equal outflow. This means that the following constraint is imposed:

$$Bf = \nu = [16806, 0, \dots, 0, -16806]^T \tag{13}$$

Also, in this and the following, the CVX package was used to solve the linear programs.

To solve the SO-TAP using the delay function from equation 12, we use the linear program specified by equation 4. The cost function becomes:

$$\psi_e(f_e) = f_e\tau_e(f_e) = l_e c_e\big(\frac{1}{1 - f_e/c_e} - 1\big).$$

The results can be found in the first column of Table 2.

3.5 Wardrop equilibrium (UO-TAP): $f^{(0)}$

Now the Wardrop equilibrium was calculated for the same delay-function τ_e from equation 12. Now, as the Wardrop equilibrium is to be found, the cost function was defined as the following:

$$\psi_e(f_e) = \int_0^{f_e} \tau_e(s)ds = [0 \leq f_e < 0] = c_e l_e \log\big(\frac{c_e}{c_e - f_e}\big)$$

With this choice of cost-function the program from equation 4 was solved once again. The results can be found in the second column of Table 2.

3.6 UO-TAP with tolls: $f^{(\omega)}$

Now tolls are introduced, and the goal is to compute the Wardrop equilibrium $f^{(\omega)}$. The tolls are introduced according so that

$$w_e = f_e^* \tau_e'(f_e^*) \tag{14}$$

where f_e^* is the system-optimum flows earlier found. Thus, according to Theorem 1 and equation **??**, $f^{(\omega)}$ should coincide with f^*.

We solve the linear program given by 4 , now with:

$$\psi_e(f_e) = \int_0^{f_e} \tau_e(s) + \omega_e ds = \int_0^{f_e} \tau_e(s)ds + f_e\omega_e = c_e l_e \Big(\log\big(\frac{c_e}{c_e - f_e}\big) + \frac{f_e f_e^*}{(c_e - f_e^*)^2} \Big)$$

The results can be seen in the third column of Table 2. By comparing column 1 and column 3 in the table, we see that the flow vector for the SO-TAP and the flow vector for the Wardrop equilibrium with tolls according to 14 mostly coincide. As they should in theory coincide, we believe that the small difference found is due to numercal precision.

3.7 Another Cost-Function

Now we consider another cost function for the SO-TAP problem, namely that:

$$\psi_e(f_e) = f_e(\tau_e(f_e) - l_e). \tag{15}$$

How can we interpret this? It is easier to start with the new "delay-function" $(\tau_e(f_e) - l_e)$. This is the delay under a flow f_e, after subtracting l_e (the empty road time it takes traverse the edge). This would model "how much of a traffic-jam there is". If we rewrite the expression:

$$\tau_e(f_e) - l_e = l_e\big(\frac{f_e}{c_e - f_e}\big) = l_e \frac{\text{flow}}{\text{residual capacity}}$$

The flow over an edge divided by the residual capacity can be interpreted as a utilization measure. So a high utilization measure indicates that there a lot of traffic jams on an infinitesimal part of the edge. The l_e acts as a weight, it is more important to minimize traffic jams on longer roads, i.e. edges with higher l_e values. Then, if we return to the cost-function:

$$\psi_e(f_e) = f_e l_e\big(\frac{f_e}{c_e - f_e}\big) \tag{16}$$

we have multiplied the weighted traffic-jam with the total flow. Considering the flow as drivers, this can be interpreted as the total amount of traffic jam experienced by the agents traversing edge e. So, by minimizing this cost-function, we minimize how much traffic jams that will be experienced in the system.

We minimize this in the usual manner with the linear program from 4, using equation 16 as cost function. The results can be seen in the fourth column of Table 2.

Now turning to the UO-TAP, we are, as a central organisation, interested in finding tolls so that the egoistic driver will choose so that this "as little traffic jams as possible system" found in SO-TAP actually is achieved. Once again, we use tolls, and the theory provided in Theorem 1. Now we choose the tolls as the following:

$$\omega_e^* = f_e^* \frac{\partial}{\partial f_e}(\tau_e(f_e) - l_e)|_{f_e^*} = f_e^* \tau_e(f_e^*). \tag{17}$$

The cost function for the Wardrop is provided in the usual manner:

$$\psi_e(f_e) = \int_0^{f_e} \tau_e(s) - l_e + \omega_e^* ds = c_e l_e \left(\log(\frac{c_e}{c_e - f_e}) + \frac{f_e f_e^*}{(c_e - f_e^*)^2} \right) - f_e l_e \tag{18}$$

And the linear program is solved once again, with the choice of cost function according to 18. The results are presented in the fifth column of Table 2. By comparing column 4 and 5, we conclude that the values for the SO-TAP and Wardrop equilibrium with tolls according to 17 coincide up to numerical precision.

Vertex	f_1^*	$f_1^{(0)}$	$f_1^{(\omega^*)}$	f_2^*	$f_2^{(\omega^*)}$
1	6642.29	6715.64	6642.30	6653.27	6653.27
2	6058.94	6715.64	6058.95	5774.88	5774.89
3	3132.36	2367.40	3132.36	3419.63	3419.63
4	3132.36	2367.40	3132.36	3419.63	3419.63
5	10163.70	10090.35	10163.67	10152.73	10152.73
6	4638.33	4645.39	4638.33	4642.71	4642.71
7	3006.33	2803.84	3006.33	3105.75	3105.75
8	2542.54	2283.55	2542.54	2662.05	2662.04
9	3131.52	3418.48	3131.53	3009.11	3009.11
10	583.35	1.06e-05	583.34	878.39	878.39
11	9.93e-05	176.82	1.18e-05	3.28e-05	1.69e-05
12	2926.57	4171.41	2926.59	2355.26	2355.26
13	1.60e-05	1.62e-06	2.62e-06	3.15e-05	1.62e-05
14	3132.36	2367.40	3132.37	3419.63	3419.63
15	5525.36	5444.95	5525.37	5510.02	5510.02
16	2854.28	2353.16	2854.27	3043.58	3043.58
17	4886.43	4933.34	4886.44	4881.79	4881.79
18	2215.34	1841.54	2215.34	2415.35	2415.35
19	463.79	697.11	463.80	443.70	443.70
20	2337.59	3036.49	2337.59	2008.19	2008.19
21	3318.07	3050.27	3318.07	3487.28	3487.28
22	5655.67	6086.76	5655.66	5495.48	5495.47
23	2373.08	2586.50	2373.07	2203.87	2203.87
24	1.69e-05	2.07e-06	2.62e-06	1.03e-05	4.95e-06
25	6414.11	6918.74	6414.12	6300.71	6300.71
26	5505.44	4953.91	5505.44	5623.50	5623.50
27	4886.43	4933.34	4886.44	4881.79	4881.79
28	4886.43	4933.34	4886.44	4881.79	4881.79

Table 2: Each column represents a flow vector, and each row the corresponding edge. Each computation was done in Long format, but have been rounded up to two decimals. The subscripts in the first rows are to differentiate between the form of cost function, where the (1) uses the delay, and (2) is for the difference between delay and length. In the first column, the result for the SO-TAP with cost function as described in equation 12. In the second column, the Wardrop equilibrium flow vector for the same delay function is shown. Then, in the third column is the Wardrop equilibrium with tolls set according to equation 14. In the fourth column, the flow vector when using the cost-function from equation 16 is presented. Lastly, in the fifth column, the Wardrop equilibrium with tolls according equation 17, and cost function according to 18 can be found.