Perturbed utility stochastic traffic assignment*

Rui Yao[1], Mogens Fosgerau[†2], Mads Paulsen[3], and Thomas Kjær Rasmussen[3]

[1]EPFL, Switzerland
[2]University of Copenhagen, Denmark
[3]Technical University of Denmark, Denmark

Short summary

This paper develops a fast algorithm for computing the equilibrium assignment with the perturbed utility route choice (PURC) model. Without compromise, this allows the significant advantages of the PURC model to be used in large-scale applications. We formulate the PURC equilibrium assignment problem as a convex minimization problem and find a closed-form stochastic network loading expression that allows us to formulate the Lagrangian dual of the assignment problem as an unconstrained optimization problem. To solve this dual problem, we formulate a quasi-Newton accelerated gradient descent algorithm (qN-AGD*). Our numerical evidence shows that qN-AGD* clearly outperforms a conventional primal algorithm as well as a plain accelerated gradient descent algorithm. qN-AGD* is fast with a runtime that scales about linearly with the problem size, indicating that solving the perturbed utility assignment problem is feasible also with very large networks.

Keywords: Perturbed utility, stochastic traffic assignment, dual algorithm, closed-form network loading, network route choice

1 Introduction

Traffic assignment deals with the problem of allocating travel demands between a set of origin-destination (OD) pairs onto a congestible transportation network under specific behavioral assumptions (Sheffi, 1985). This problem is central to transportation network planning and analysis.

The perturbed utility route choice (PURC) model (Fosgerau et al., 2022, 2023) is a link-based model that predicts the demand of a traveler as the network flow vector by solving a certain convex optimization problem. The model has a number of features that are very attractive for applications. It generates realistic predictions of network flows with substitution patterns induced directly by the network structure, while allowing a priori any physically possible route in the transportation network without the need for choice set generation, and while predicting zero flow in irrelevant parts of the network. Not least, the PURC model can be estimated by linear regression.

However, to the best of our knowledge, the perturbed utility based traffic assignment problem has neither been formulated nor solved in the literature, while crucial for applying PURC for traffic flow predictions and network planning. This paper formulates the PURC traffic assignment problem and develops a fast approach for computing the PURC traffic equilibrium, thus making it relevant for large-scale applications. Specifically, we exploit properties of the perturbed utility stochastic traffic assignment model, and formulate it as an unconstrained Lagrangian dual problem that facilitates the development of an efficient quasi-Newton accelerated gradient descent (qN-AGD*) algorithm (Nesterov, 1983; Beck & Teboulle, 2009; Chambolle & Dossal, 2015).

*We refer to the full paper (under review) at https://dx.doi.org/10.2139/ssrn.4652411.

†Corresponding author: mogens.fosgerau@econ.ku.dk.

We test the performance of our qN-AGD* algorithm with a range of networks of various sizes and find very satisfactory runtimes. Importantly, we find that the runtime scales about linearly with the size of the problem, i.e., the number of nodes times the number of origin-destination pairs, suggesting that our algorithm will perform well with very large problems.

Perturbed utility route choice model

The perturbed utility model (Fosgerau & McFadden, 2012; Allen & Rehbeck, 2019) in its general form, describes consumer choice as a vector x that maximizes a concave function $v^\top x - F(x)$, where v is a vector of utility indexes and F is the convex perturbation function. The consumption vector x is constrained to lie in some budget set B. In words, the perturbed utility route choice model (Fosgerau et al., 2022) is a perturbed utility model in which the budget set for a traveler is the set of network flow vectors that satisfy flow conservation of one unit demand traveling from the origin to the destination. Moreover, the convex perturbation function incorporates the network structure by being constructed as a sum of convex terms, one for each link in the network. As a consequence, correlation between alternatives is directly induced by the physical network structure.

We now formally introduce PURC. A network is given by $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes with typical element v, and $\mathcal{E}$ is the set of directed edges (or links) with typical element (i, j) for a link from node i to node j. We assume there exists at least one path between any two network nodes, i.e.

Assumption 1 *The network $(\mathcal{V}, \mathcal{E})$ is connected.*

The node-link incident matrix $A \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{E}|}$ has entries

$$a_{v,ij} = \begin{cases} -1, & v = i, \\ 1, & v = j, \\ 0, & \text{otherwise.} \end{cases}$$

A unit demand vector $b \in \mathbb{R}^{|\mathcal{V}|}$ is given as

$$b_v = \begin{cases} -1, & \text{if } v \text{ is the traveler's origin,} \\ 1, & \text{if } v \text{ is the traveler's destination,} \\ 0, & \text{otherwise.} \end{cases}$$

A network link flow vector $x \in \mathbb{R}_+^{|\mathcal{E}|}$ satisfies flow conservation if $Ax = b$.

A traveler in the PURC model is associated with a demand vector b, a vector of positive link costs[1] $c \in \mathbb{R}_{++}^{|\mathcal{E}|}$, and link-specific convex perturbation functions F_e, where

Assumption 2 *The perturbation function $F_e : \mathbb{R}_+ \to \mathbb{R}_+, \forall e \in \mathcal{E}$, is continuously differentiable, strictly convex, and strictly increasing, with $F_e(0) = F_e'(0) = 0$ and range equal to $\mathbb{R}_+$. Define $(F_e')^{-1}(y) = 0$ for $y < 0$ such that the inverse function of the derivative of the perturbation function has domain equal to $\mathbb{R}$.*

For a flow vector $x \in \mathbb{R}_+^{|\mathcal{E}|}$, we define

$$F(x) = \sum_{e \in \mathcal{E}} F_e(x_e) \tag{1}$$

[1]For simplicity, we deviate slightly from Fosgerau et al. (2022) by not making explicit the dependence of link costs on link lengths. This is just a question of notation.

for the sum across links of perturbations $F_e(x_e)$.[2]

The PURC model assumes that the traveler chooses link flow vector x to minimize a perturbed cost function $c^\top x + F(x)$, under the flow conservation constraint.[3] Thus, the traveler's demand solves the following convex program.

$$\min_{x\in\mathbb{R}_+^{|\mathcal{E}|}} c^\top x + F(x) \tag{2a}$$

$$\text{s.t.} \quad Ax = b. \tag{2b}$$

It is an important feature of the program (2) that the objective (2a) is convex and separable by links. The coupling across links arises only through the linear conservation constraint (2b). Fosgerau et al. (2022) exploit this property to derive an estimation procedure that requires only linear regression.

Stochastic traffic assignment models and algorithms

In this subsection, we briefly review link-based stochastic traffic assignment models and algorithms. In contrast to path-based models, link-based models, as in the case of PURC, have the advantage of avoiding the challenging path set generation. We refer detailed discussion on path-based models to our full paper (Yao et al., 2023). In essence, a stochastic traffic assignment model predicts network flows in stochastic user equilibrium (SUE), under which no traveler can reduce their perceived travel cost by unilaterally changing their routing decision (Daganzo & Sheffi, 1977).

For example, link-based recursive route choice models has been applied in SUE, which avoids choice set generation by assuming travelers sequentially choose the next link at each node. This family of models includes the recursive logit (RL) (Fosgerau et al., 2013), nested RL (Mai et al., 2015), and recursive network GEV (Mai, 2016). However, estimation of these recursive models is challenging (Oyama, 2023; Mai & Frejinger, 2022). In contrast, the perturbed utility route choice model (Fosgerau et al., 2022) requires only linear regression for model estimation, while it is still capable of capturing correlations and avoids choice set generation.

Correspondingly, a number of stochastic traffic assignment models have been proposed based on these route choice models. Akamatsu (1997) propose a link-based assignment formulation with infinite path set, which decomposes the path entropy term with respect to link flow. The traveler route choice behavior in Akamatsu (1997) is indeed characterized by the RL model (Fosgerau et al., 2013). The Akamatsu (1997) formulation will generally predict cyclic flows. This is caused by the Markov property in combination with the fact that it assigns positive probability to all out-going links (Akamatsu, 1996). For the same reason, positive flow is assigned on all links in the network, which might be behaviorally questionable and computationally challenging. Oyama et al. (2022) has demonstrated the efficiency of accelerated gradient descent (AGD) method (Nesterov, 1983; Beck & Teboulle, 2009; Sutskever et al., 2013) in solving the recursive NGEV assignment problem, which accumulates the past gradients to guide the current iterate (Polyak, 1964). However, the original AGD method often exhibits an oscillatory convergence pattern, which detracts from its performance. Chambolle & Dossal (2015) proposed a modification of the AGD algorithm (termed AGD* in this paper) with superior practical performance and an improved theoretical convergence rate (Attouch & Peypouquet, 2016).

[2]Fosgerau et al. (2022) defines link-specific perturbation functions by multiplying a basic perturbation function by link lengths. This ensures that the overall perturbation function F is invariant with respect to link splitting.

[3]In the route choice context, it is equivalent but more natural to talk about cost minimization rather than utility maximization.

In this paper, contrasting to classic Markovian traffic assignment models, we propose a stochastic traffic assignment model for a generic convex perturbation with a condition that induces the optimal flow to be zero on most links (Fosgerau et al., 2022), i.e. those that are not in the set of optimal paths. In addition, we propose a quasi-Newton extension of the AGD* method that automatically scales the gradient with the Hessian diagonal, which not only improves convergence but also avoids any line search for step sizes.

2 The perturbed utility-based traffic assignment problem

We now set up the traffic assignment problem for the perturbed utility route choice model. We consider a general assignment setting with multiple traveler types and allow for arbitrary heterogeneity. We denote the set of traveler types by $\mathcal{W}$ with typical element w and the volume of travelers of each type w by q^w.

The network and the link travel times are common across types. Congestion in the network causes interaction between travelers. Otherwise, each type is assumed to behave according to its own perturbed utility route choice model (2) as described in the previous section. We index the link cost functions, the perturbation functions, and the demand vectors by the type, c_{ij}^w, F_{ij}^w, and b^w. The link cost function c_{ij}^w is a type-specific function of link travel time t_{ij}. This allows e.g. type-specific preferences against tolls, while the link travel time is common to all types. We make the following assumptions.

Assumption 3 *The link cost functions $c_{ij}^w : \mathbb{R}_+ \to \mathbb{R}_+, (i,j) \in \mathcal{E}, w \in \mathcal{W}$ are positive, continuously differentiable, and increasing with respect to link flow, and convex, $c_{ij}^w(0) > 0$, $c_{ij}^{w\prime} > 0$.*

Assumption 4 *The link travel time functions $t_{ij} : \mathbb{R}_+ \to \mathbb{R}_+, (i,j) \in \mathcal{E}$ are positive, differentiable, increasing, and strictly convex, $t_{ij} > 0, t_{ij}' > 0$.*

The travel time on link (i,j) depends on the link flow $x_{ij} = \sum_{w \in \mathcal{W}} q^w x_{ij}^w$ and the link cost for type w is then $c_{ij}^w(t_{ij}(x_{ij}))$. Assumptions 3 and 4 combine to ensure that the link costs are convex functions of link flow.

Primal formulation

We will formulate a convex minimization problem, whose solution is the Wardrop/Nash equilibrium where all travelers make individually optimal choices according to (2), taking link costs as given. Link costs, in turn, depend through the common travel time on the link flow, which is the aggregated demand from the individual travelers. This is our primal perturbed utility-based stochastic traffic assignment problem (TAP).

[TAP]

$$\min_{\boldsymbol{x}} Z = \sum_{(i,j)\in\mathcal{E}} \sum_{w\in\mathcal{W}} \left[\int_0^{\sum_{w'\in\mathcal{W}} q^{w'} x_{ij}^{w'}} c_{ij}^w(t_{ij}(m))dm + q^w F_{ij}^w(x_{ij}^w) \right] \tag{3a}$$

$$s.t. \quad x_{ij}^w \geq 0, \ \forall (i,j) \in \mathcal{E}, w \in \mathcal{W} \tag{3b}$$

$$q^w(A_v \boldsymbol{x}^w - b_v^w) = 0, \ \forall v \in \mathcal{V}, w \in \mathcal{W}. \qquad (\eta_v^w) \tag{3c}$$

In this expression, $\boldsymbol{x}^w = (x_e^w)_{e\in\mathcal{E}}$ is the link flow vector for type w, and $\boldsymbol{x} = \{x_e^w\}_{e\in\mathcal{E},w\in\mathcal{W}}$. A_v is the row vector corresponding to node v of the incidence matrix A. Eq. (3c) is the conservation constraints with corresponding Lagrange multipliers η_v^w for each node and type. We denote $\boldsymbol{\eta} =$

$(\eta_v^w)_{v\in\mathcal{V},w\in\mathcal{W}}$.

Our proposed primal TAP formulation extends Beckmann et al. (1956)'s deterministic UE formulation with the addition of a perturbation term. The PURC is similar to the recursive NGEV formulation of Oyama et al. (2022) with the important difference that PURC allows corner solutions, i.e. links with zero flow. In particular, because of the requirements on the PURC perturbation function, the optimal flow for a given OD concentrates on a relatively small number of paths, while it is zero in the rest of the network.

We first show that the equilibrium link flow pattern in the primal TAP equals the PURC demand for each w. We kindly refer detail proof in the full paper.

Proposition 1 *The stochastic user equilibrium condition in TAP* (3) *is equivalent to the optimality condition in PURC* (2)*, such that traveler route choice behavior is in accordance with the PURC model.*

Proposition 2 (Existence and uniqueness of TAP) *The primal traffic assignment problem (TAP) admits a unique solution.*

Dual formulation

The primal TAP involves a large number of flow conservation constraints, one for each combination of type and network node. In this section, we will develop a closed-form expression for the dual problem corresponding to the primal TAP, and we will show that the dual problem is unconstrained. This will be useful for finding a fast solution algorithm.

The Lagrangian function for the primal TAP (3) is

$$\begin{aligned} L(\boldsymbol{x},\boldsymbol{\eta}) = &\sum_{(i,j)\in\mathcal{E}}\sum_{w\in\mathcal{W}}\left[\int_0^{\sum_{w'\in\mathcal{W}} q^{w'}x_{ij}^{w'}} c_{ij}^w(t_{ij}(m))dm + q^w F_{ij}^w(x_{ij}^w)\right] \\ &- \sum_{(i,j)\in\mathcal{E}}\sum_{w\in\mathcal{W}} q^w\left(\eta_i^w - \eta_j^w\right)x_{ij}^w - \sum_{v\in\mathcal{V}}\sum_{w\in\mathcal{W}} q^w\eta_v^w b_v^w &\quad (4a) \\ \text{s.t.}\quad & x_{ij}^w \geq 0,\ \forall (i,j)\in\mathcal{E}, w\in\mathcal{W}, &\quad (4b) \end{aligned}$$

where $\eta_i^w, w\in\mathcal{W}, i\in\mathcal{V}$ are the Lagrangian multipliers for the flow conservation constraints. As for the individual traveler, we are free to impose the normalization that $\eta_{d^w}^w = 0$, such that these Lagrangian multipliers can be interpreted as the minimum perceived cost from node i to the destination node d^w of each type w.

The Lagrangian function (4) has simple constraints, but adds extra decision variables, the dual variables η_{ij}^w. Using the following closed-form expression for the flow variables x_{ij}^{w*} as a function of the dual variables allows us to reduce the number of decision variables considerably.

Proposition 3 (Perturbed utility-based network loading) *The optimal link flow x_{ij}^{w*} for given η is*

$$x_{ij}^{w*} = x_{ij}^{w*}(\eta_i^{w*},\eta_j^{w*},c_{ij}^{w*}) = (F_{ij}^{w'})^{-1}\left(\eta_i^{w*} - \eta_j^{w*} - c_{ij}^{w*}\right), \tag{5}$$

where

$$c_{ij}^{w*} = c_{ij}^w\left(t_{ij}\left(\sum_{w\in\mathcal{W}} q^w x_{ij}^{w*}\right)\right). \tag{6}$$

Having determined x_{ij}^{w*} as a function of η_i^{w*}, η_j^{w*} and c_{ij}^{w*} also in the primal TAP, we can substitute that into the Lagrangian (4) to obtain the corresponding Lagrangian dual and the dual traffic assignment problem (DTAP).

[DTAP]

$$\max_{\boldsymbol{\eta}} G = \sum_{(i,j)\in\mathcal{E}} \sum_{w\in\mathcal{W}} \left[\int_0^{\sum_{w'\in\mathcal{W}} q^{w'} x_{ij}^{w'*}} c_{ij}^w(t_{ij}(m))dm + q^w F_{ij}^w(x_{ij}^{w*}) \right] - \sum_{(i,j)\in\mathcal{E}} \sum_{w\in\mathcal{W}} q^w \left(\eta_i^w - \eta_j^w\right) x_{ij}^{w*} - \sum_{v\in\mathcal{V}} \sum_{w\in\mathcal{W}} q^w \eta_v^w b_v^w \tag{7}$$

We note that DTAP is unconstrained with the node potentials $\boldsymbol{\eta}$ being the only decision variables. This allows us to adapt existing fast algorithms for solving the DTAP. The closed-form network loading expression (5), allows us to directly obtain the corresponding individual flows. The following lemma shows that the strong duality condition holds, such that solving the dual problem is equivalent to solving the primal TAP (3).

Lemma 1 (Strong duality) *The duality gap between the primal problem TAP* (3) *and the corresponding dual problem at their optimal solutions is zero.*

Finally, we need to verify that the DTAP admits a solution, such that we can always use the unconstrained DTAP to solve the perturbed utility-based stochastic traffic assignment problem.

Lemma 2 (Existence of DTAP) *The DTAP* (7) *admits at least one solution* η^*.

3 Solution method

In this section, we propose a fast algorithm for solving the dual assignment problem (7). We propose a quasi-Newton accelerated gradient descent algorithm (qN-AGD*), which uses the AGD* scheme (Chambolle & Dossal, 2015) to reduce oscillation, and uses the Hessian diagonal to automatically scale the gradient in a quasi-Newton manner with fixed step size, without the need for a backtracking procedure.

Recall that the optimal link flows (5) depend on the type-specific link costs c_{ij}^{w*}. However, these in turn depend on the link flows through the link travel time functions t_{ij}. To tackle this problem, existing primal algorithms adapt the Gauss-Seidel approach, which decomposes the assignment problem for each origin-based network flow and iteratively solves these subproblems (e.g., Dial, 2006; Y. M. Nie, 2010). In contrast, we consider all the node potential variables as one block of variables, while the link travel times are considered as another separate block of variables. Thus the assignment problem is decomposed into only two subproblems. This exploits that all node potentials η can be updated in parallel, while the link travel times t can be updated subsequently.

We cast the link travel time problem as an auxiliary fixed-point problem for given η.

$$t_{ij} = t_{ij}\left(\sum_{w\in\mathcal{W}} q^w \cdot (F_{ij}^{w'})^{-1} \left(\eta_i^{w*} - \eta_j^{w*} - c_{ij}^{w*}\right) \right)$$

with corresponding residual function $U_{ij}(x_{ij}^*, t_{ij}^*) = t_{ij}\left(\sum_{w\in\mathcal{W}} q^w x_{ij}^{w*}\right) - t_{ij}^* = 0$

We find that the fixed-point t^* exists.

Proposition 4 *Under Assumption 4, the fixed point* $t_{ij}^*, (i,j) \in \mathcal{E}$ *exists.*

We now propose the following dual assignment algorithm, combining the AGD* algorithm with a quasi-Newton gradient scaling, as well as a Newton step for updating link travel times.

Algorithm 1 Dual assignment algorithm – qN-AGD*

Step 0: Initialization. Input initial points $\eta^{(0)}$ and $c^{*(0)}$, and step sizes γ_1, γ_2, set iteration counter $m = 0$, momentum acceleration variable $r_0 = 1$ and momentum acceleration parameter $\alpha > 1$.

Step 1: Iteration.

Step 1.1. - *PURC assignment*:

$$x_{ij}^{w*(m+1)} = \min\left\{1, (F_{ij}^{w\prime})^{-1}\left(\eta_i^{w(m)} - \eta_j^{w(m)} - c_{ij}^{w*(m)}\right)\right\}. \tag{8}$$

Step 1.2. - *Update Lagrangian multipliers η, with Nesterov's momentum acceleration*:

$$\tilde{\eta}_j^{w(m+1)} = \eta_j^{w(m)} + \gamma_1 \tilde{\nabla}_{\eta_j^w} G(\eta^{(m)}), \tag{9a}$$

$$\eta_j^{w(m+1)} = \tilde{\eta}_j^{w(m+1)} + \frac{m}{m+\alpha}\left(\tilde{\eta}_j^{w(m+1)} - \tilde{\eta}_j^{w(m)}\right). \tag{9b}$$

Step 1.3. - *Update auxiliary link travel times t^* and c^{w*}, by one Newton step*:

$$t_{ij}^{*(m+1)} = t_{ij}^{*(m)} - \gamma_2 \frac{U_{ij}\left(x_{ij}^{w*(m+1)}, t_{ij}^{*(m)}\right)}{\nabla_{t_{ij}^*} U_{ij}\left(x_{ij}^{w*(m+1)}, t_{ij}^{*(m)}\right)}. \tag{10}$$

In addition, update link cost:

$$c^{w*(m+1)} = c^w\left(t_{ij}^{*(m+1)}\right). \tag{11}$$

Step 2: Convergence test. If the stopping criteria hold, **stop**. Otherwise, set $m = m + 1$ and go to **Step 1**.

Next, we will show how to compute the gradient of the Lagrangian dual $G(\boldsymbol{\eta})$. By the envelope theorem (Milgrom & Segal, 2002), for the given link costs c^* and x_{ij}^{w*} (by Eq. 5), the gradient of $G(\boldsymbol{\eta})$ w.r.t. η_k^w is

$$\begin{aligned}\nabla_{\eta_k^w} G(\eta^{(m)}) &= q^w\left[\sum_{i:(i,k)\in\mathcal{E}} x_{ik}^{w*(m+1)} - \sum_{j:(k,j)\in\mathcal{E}} x_{kj}^{w*(m+1)} - b_k^w\right] \\ &= q^w(A_k x^{w*(m+1)} - b_k^w).\end{aligned} \tag{12}$$

We further propose to scale the gradient with an upper bound of the Hessian diagonal to speed up convergence (Y. Nie, 2012). This is often considered as a quasi-Newton method in primal deterministic assignment algorithms (e.g., Bar-Gera, 2002; Y. M. Nie, 2010). The corresponding scaled gradient is

$$\tilde{\nabla}_{\eta_k^w} G(\eta^{(m)}) = \frac{\nabla_{\eta_k^w} G(\eta^{(m)})}{q^w A_k\left(\nabla_{\eta_k^{w(m)}}((F_{ij}^w)')^{-1}\right)^{|\mathcal{E}|\times 1}}, \tag{13}$$

where the division is performed element-wise. Note that, in the qN-AGD* algorithm, the scaled gradient (13) is used to update auxiliary node potentials $\tilde{\eta}$ in Eq. (9a), which makes it a quasi-Newton method. This is in contrast to the AGD and AGD* algorithms which are first-order methods that do not utilize any Hessian.

For the auxiliary fixed-point problem, we apply a Newton step (10) for updating the travel times. The gradient of U_{ij} is

$$\nabla_{c^*_{ij}} U_{ij} = \frac{\partial c_{ij}}{\partial x_{ij}} \sum_{w \in \mathcal{W}} q^w \nabla_{c^{w*(m)}_{ij}} ((F^w_{ij})')^{-1}. \tag{14}$$

We consider the algorithm to have converged when the following two criteria both hold:

$$R_1^{(m+1)} = \sum_{w \in \mathcal{W}} \frac{q^w}{Q} \frac{||Ax^{w*(m+1)} - b^w||_1}{|\mathcal{V}|} \leq \epsilon_1 \text{ and } R_2 = \frac{||t\left(x^{w*(m+1)} \cdot q^{|\mathcal{W}| \times 1}\right) - t^{*(m+1)}||_1}{|\mathcal{E}|} \leq \epsilon_2,$$

where $||\cdot||_1$ denotes the L_1-norm, total demand is $Q = \sum_{w \in \mathcal{W}} q^w$, and $t(\cdot)$ is the vector function for link travel times depending on link flows. R_1 is the mean absolute error of the first-order condition for η^* weighted by demands, and R_2 is the mean absolute error of t^* to the auxiliary fixed point.

4 Numerical experiments

In this section, we demonstrate the performance of our proposed dual assignment algorithm through a series of numerical experiments.

We use the entropy perturbation function

$$F^w_{ij}(x^w_{ij}) = (1 + x^w_{ij}) \ln(1 + x^w_{ij}) - x^w_{ij}, \forall (ij) \in \mathcal{E}, w \in \mathcal{W}, \tag{15}$$

we initialize $\eta^{(0)}, c^{*(0)}$ with the costs from on all-or-nothing assignment not including the perturbation term, set the momentum acceleration parameter to $\alpha = 10$, and the stopping criteria parameters to $\epsilon_1 = \epsilon_2 = 10^{-5}$. We specify the link cost function simply as $c_{ij}(x_{ij}) = 0.5 t_{ij}(x_{ij})$, where t_{ij} is the Bureau of Public Roads (BPR) volume-delay function. For the proposed qN-AGD* algorithm, we fix the step sizes as $\gamma_1 = 0.5, \gamma_2 = 1$, which provide satisfactory performance in all numerical experiments. For reference, we also tested the performances of the AGD method (Beck & Teboulle, 2009; Oyama et al., 2022), the AGD* method (Chambolle & Dossal, 2015). For each reference algorithm, we report the best convergence result among step size choices of $\{10^{-4}, 10^{-5}, 10^{-6}\}$. All algorithms are implemented in PyTorch to enable GPU computation and run on a HPC cluster with an A100 GPU. The real networks are obtained from Bar-Gera (2016), with the default volume-delay functions and demand matrix as provided.

Convergence performance at different demand levels

We begin by comparing the convergence performance of the proposed qN-AGD* algorithm to the qN-AGD, AGD* and AGD algorithms, under three demand levels (1q, 1.5q, 2q). For this comparison, we use the Sioux Falls network, which has 76 links, 24 nodes, 528 OD pairs and 360,600 trips (1q).

Figure 1 shows the convergence of the first-order gap R_1 for the four algorithms, under three demand levels. The proposed qN-AGD* algorithm is the fastest of the four algorithms at any demand level: the qN-AGD* under 2q demand is even faster than AGD* at 1q demand. In addition, the qN-AGD* runtime at different demand levels only has small variations, which suggests the proposed qN-AGD* has the potential to work well also for congested networks. The qN-AGD is generally second best, which indicates that the quasi-Newton scaling is important. Among AGD* and AGD, AGD* is faster for higher precision solutions (e.g., with $R_1 = 10^{-5}$), while AGD is

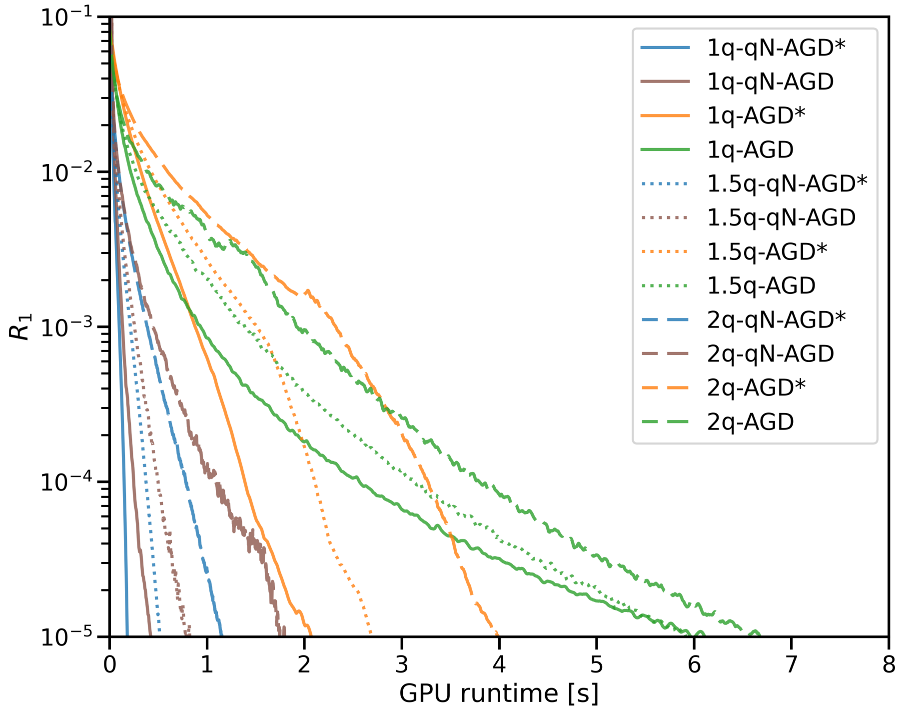

Figure 1: Convergence performance of the proposed quasi-Newton AGD* (qN-AGD*), qN-AGD, AGD*, and the original AGD, with three demand levels (1q, 1.5q, 2q). (GPU runtimes in sec vs. mean absolute error of the first-order condition.)

faster for lower precision (e.g., with $R_1 = 10^{-2}$). This is because AGD* avoids oscillation with a more conservative updating scheme ($\alpha > 3$), which might be slower at the earlier iterations than AGD (approximated by $\alpha = 3$) (Liang et al., 2022).

Runtime performance at different network sizes and demand levels

In this subsection, we systematically evaluate the computational efficiency of the proposed qN-AGD* algorithm and the qN-AGD (which exhibits similar performance) in real-size networks at different demand levels. We here do not consider AGD* and AGD as they clearly showed inferior convergence.

As shown in Figure 2, we construct the grid test networks as proposed in Oyama et al. (2022) by joining blocks of grids, to exemplify the effects of network size and demand levels on runtime performance. We assume the BPR function $t_{ij} = t_{ij,0}\left(1 + 0.15(\frac{x_{ij}}{\kappa_{ij}})^4\right)$ with free-flow travel time $t_{ij,0} = 1$, link capacity $\kappa_{ij} = 5{,}000$, and link length $l_{ij} = 1$, for all links $(i,j) \in \mathcal{E}$. For each origin o, the demand for each destination $d \in \mathcal{D}$ is assumed to follow the gravity model:

$$q_o^d = q \cdot \frac{\exp(t_{od,0})}{\sum_{d \in \mathcal{D} \setminus o} \exp(t_{od,0})}, \forall o \in \mathcal{O},$$

where $t_{od,0}$ is the shortest free-flow travel time between OD pair od, and q is the total trips generated from each origin.

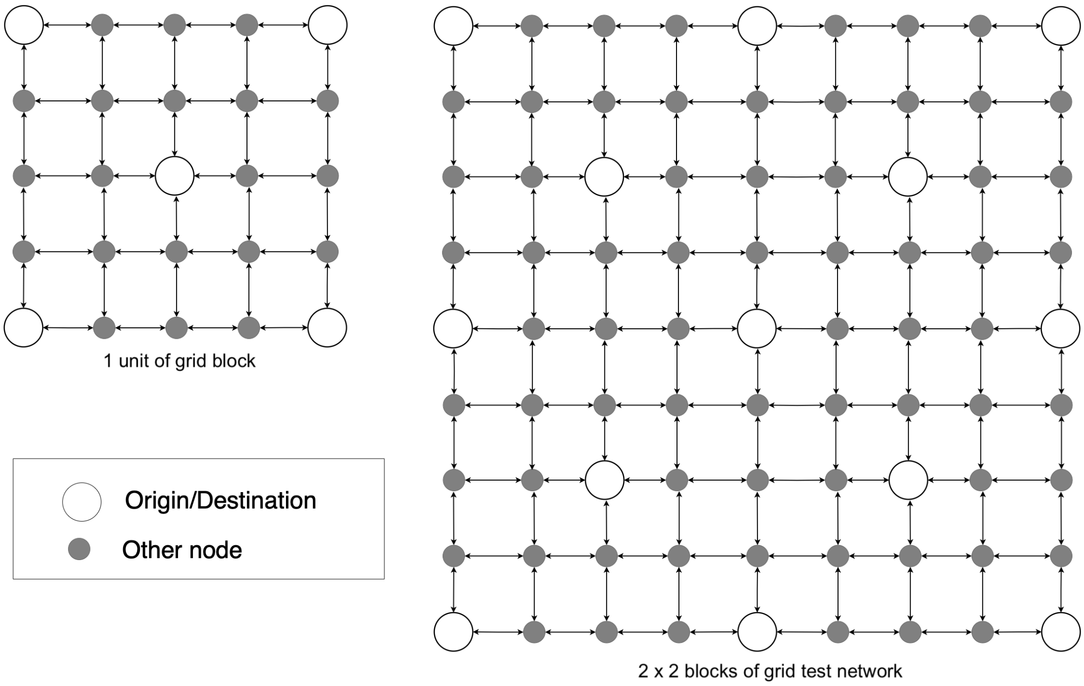

Figure 2: Bidirectional $k \times k$ blocks of grid test network.

An indicator for the size of the assignment problem is the number of decision variables, i.e., the number of nodes times the number of traveler types. As shown in Figure 3, we find the runtime depends about linearly on the problem size. This suggests that our proposed algorithm will scale well to larger networks. Furthermore, the runtime only increases slightly with increasing demands. Note that the grid test networks are congested under our setting. This result further suggests that our proposed algorithm is suitable also for congested networks.

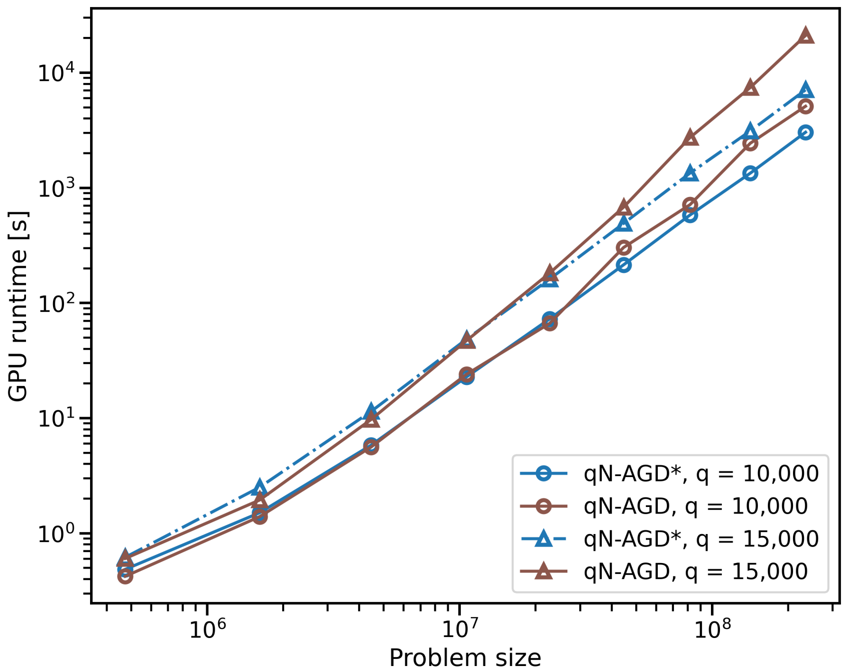

Figure 3: Runtime performance to problem size (k ranging from 4 to 12)

5 Conclusion

This paper has proposed the perturbed utility stochastic traffic assignment model and an accompanying accelerated gradient-based algorithm based on the equivalent dual formulation of the traffic assignment problem. The dual assignment problem is unconstrained with closed-form stochastic network loading, which helps to make our proposed qN-AGD* algorithm very fast.

Our simulation evidence suggests that our proposed algorithm will scale well to larger problems. This is important for making the perturbed utility route choice model competitive with other route choice models. This is also valuable for applying the novel perturbed utility route choice model in real applications.

References

Akamatsu, T. (1996). Cyclic flows, markov process and stochastic traffic assignment. *Transportation Research Part B: Methodological*, *30*(5), 369–386.

Akamatsu, T. (1997). Decomposition of path choice entropy in general transport networks. *Transportation Science*, *31*(4), 349–362.

Allen, R., & Rehbeck, J. (2019, 5). Identification With Additively Separable Heterogeneity. *Econometrica*, *87*(3), 1021–1054. Retrieved from `https://www.ssrn.com/abstract=3073945` doi: 10.3982/ecta15867

Attouch, H., & Peypouquet, J. (2016). The rate of convergence of nesterov's accelerated forward-backward method is actually faster than 1/k^2. *SIAM Journal on Optimization*, *26*(3), 1824–1834.

Bar-Gera, H. (2002). Origin-based algorithm for the traffic assignment problem. *Transportation Science*, *36*(4), 398–417.

Bar-Gera, H. (2016). Transportation networks for research. Retrieved from `https://github.com/bstabler/TransportationNetworks`

Beck, A., & Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, *2*(1), 183–202.

Beckmann, M., McGuire, C. B., & Winsten, C. B. (1956). *Studies in the economics of transportation* (Tech. Rep.).

Chambolle, A., & Dossal, C. (2015). On the convergence of the iterates of the "fast iterative shrinkage/thresholding algorithm". *Journal of Optimization theory and Applications*, *166*, 968–982.

Daganzo, C. F., & Sheffi, Y. (1977). On stochastic models of traffic assignment. *Transportation science*, *11*(3), 253–274.

Dial, R. B. (2006). A path-based user-equilibrium traffic assignment algorithm that obviates path storage and enumeration. *Transportation Research Part B: Methodological*, *40*(10), 917–936.

Fosgerau, M., Frejinger, E., & Karlstrom, A. (2013). A link based network route choice model with unrestricted choice set. *Transportation Research Part B: Methodological*, *56*, 70–80.

Fosgerau, M., Lukawska, M., Paulsen, M., & Rasmussen, T. K. (2023). Bikeability and the induced demand for cycling. *Proceedings of the National Academy of Sciences*, *120*(16).

Fosgerau, M., & McFadden, D. L. (2012, 3). A theory of the perturbed consumer with general budgets. *NBER Working Paper*, 1–27. Retrieved from `http://www.nber.org/papers/w17953http://www.nber.org/papers/w17953.pdf` doi: 10.3386/w17953

Fosgerau, M., Paulsen, M., & Rasmussen, T. K. (2022). A perturbed utility route choice model. *Transportation Research Part C*, *136*(103514). Retrieved from `https://arxiv.org/abs/2103.13784` doi: https://doi.org/10.1016/j.trc.2021.103514

Liang, J., Luo, T., & Schonlieb, C.-B. (2022). Improving "fast iterative shrinkage-thresholding algorithm": faster, smarter, and greedier. *SIAM Journal on Scientific Computing*, *44*(3), A1069–A1091.

Mai, T. (2016). A method of integrating correlation structures for a generalized recursive route choice model. *Transportation Research Part B: Methodological*, *93*, 146–161.

Mai, T., Fosgerau, M., & Frejinger, E. (2015). A nested recursive logit model for route choice analysis. *Transportation Research Part B: Methodological*, *75*, 100–112.

Mai, T., & Frejinger, E. (2022). Undiscounted recursive path choice models: Convergence properties and algorithms. *Transportation Science*, *56*(6), 1469–1482.

Milgrom, P., & Segal, I. (2002). Envelope theorems for arbitrary choice sets. *Econometrica*, *70*(2), 583–601.

Nesterov, Y. E. (1983). A method of solving a convex programming problem with convergence rate $O\left(\frac{1}{k^2}\right)$. In *Doklady akademii nauk* (Vol. 269, pp. 543–547).

Nie, Y. (2012). A note on bar-gera's algorithm for the origin-based traffic assignment problem. *Transportation Science*, *46*(1), 27–38.

Nie, Y. M. (2010). A class of bush-based algorithms for the traffic assignment problem. *Transportation Research Part B: Methodological*, *44*(1), 73–89.

Oyama, Y. (2023). Capturing positive network attributes during the estimation of recursive logit models: A prism-based approach. *Transportation Research Part C: Emerging Technologies*, *147*, 104014.

Oyama, Y., Hara, Y., & Akamatsu, T. (2022). Markovian traffic equilibrium assignment based on network generalized extreme value model. *Transportation Research Part B: Methodological*, *155*, 135–159.

Polyak, B. T. (1964). Some methods of speeding up the convergence of iteration methods. *Ussr computational mathematics and mathematical physics*, *4*(5), 1–17.

Sheffi, Y. (1985). *Urban transportation networks* (Vol. 6). Prentice-Hall, Englewood Cliffs, NJ.

Sutskever, I., Martens, J., Dahl, G., & Hinton, G. (2013). On the importance of initialization and momentum in deep learning. In *International conference on machine learning* (pp. 1139–1147).

Yao, R., Fosgerau, M., Paulsen, M., & Rasmussen, T. K. (2023). Perturbed utility stochastic traffic assignment. *arXiv preprint arXiv:2312.00848*.