Particle-based and stochastic traffic assignment

Gunnar Flötteröd

Communications and Transport Systems, Linköping University, Sweden
Swedish National Road and Transport Research Institute, Sweden

January 28, 2022

1 Short summary

The particle-based and stochastic traffic assignment problem is considered. The increasing deployment of "agent-based" traffic assignment models in transport planning renders this problem relevant. Lacking a comprehensive mathematical characterization, the computation of an assignment within this model class is bound to heuristics. One of these approaches is the "sorting technique" of Sbayti et al. (2007). The present article offers a derivation of this method from first principles, works out the effect of model stochasticity, and uses these results to propose a suitable simulation noise filtering rule. Experimental results covering a wide range of variations of the method under different congestion levels illustrate its advantageous performance.

2 Introduction

The traffic assignment problem amounts to solving a model system comprising a travel behavioral model that interacts with a network loading model. The travel behavioral model describes how travelers choose routes and possibly other aspects of travel (such as departure time or mode), and the network loading model describes the physical realization of the intended travel of possibly many travelers in a capacitated network. The *dynamic* traffic assignment (DTA) problem includes a within-day time dimension along which at least the network loading takes place. Wang et al. (2018) give ample references to concrete models. The arguably most intensively studied DTA problem class is that of deterministic continuum flow assignment. However, problem instances including realistic spatiotemporal flow dynamics have so far eluded a complete solution theory (Han et al., 2019), even though progress is being made under increasingly mild assumptions (Friesz et al., 2021).

The present article considers a *particle-based* and *stochastic* DTA problem. Particle-based means that the travel demand is composed of integral, decision making entities, and that the network loading is realized as a vehicle-discrete dynamic network flow simulator. Stochastic means that travel choices and/or network conditions are random variables. Due to the intricate interactions between travel choices and network flows, these distributions are in general not available in closed form; instead, the model system only allows to simulate realizations thereof. This setup is representative for many modern “agent-based” traffic assignment simulators and hence of increasing practical relevance (Barcelo, 2010).

The considered class of DTA simulators can be characterized as process-based: a scenario is composed of individual objects (travelers, vehicles, roads, ...), and the scenario is evaluated by computing the interactions of these objects according to model-based process rules. The possibility to freely select these processes renders them highly versatile but also is the cause of their mathematical intractability. Different amounts of information may be attached to a particle, and different travel behavioral degrees of freedom may be considered.

- When assigning time-dependent origin/destination matrices, each particle becomes a trip maker (representing one trip in the matrix) that only is annotated with origin, destination and departure time. Travel behavior is represented in terms of a route, possibly a travel mode (car, public transport ...), and possibly a departure time.

- When assigning a synthetic population (Farooq et al., 2013), each particle represents a synthetic traveler that is annotated with socio-demographic information such as age, income or car ownership. Travel behavior is represented in terms of an all-day travel plan that comprises a sequence of trips (annotated with route, departure time, travel mode) connecting the traveler’s activity locations.

The following presentation considers the travel plan assignment of a synthetic population. This means that every particle represents a traveler, and every travel plan represents the all-day travel (trip sequence) of that traveler. The route assignment of an origin/destination matrix is a special case thereof. In reasonable generality, a DTA simulation then functions according to Algorithm 1.

The present work considers instances of Algorithm 1 where each traveler computes for each available plan a real-valued utility. The utility may depend on past network conditions (it may, for instance, include travel time as a result of network congestion), but given the traveler, the plan, and the past network conditions, utility is deterministically determined. The specification is compatible with arbitrary random utility plan choice models as long as the random utility terms are computed prior to the DTA simulation and held constant throughout that simulation. This implies the behavioral assumption that unobserved preferences and attributes of the alternatives do not vary between assignment iterations.

It is assumed that travelers aim to maximize expected utility, but it is not assumed that they dispose of an effective mechanism for doing so. It is assumed that the travel behavioral model computes every assignment iteration for every traveler in a new *candidate* travel plan. The requirements on how this new plan is generated are very mild, they range from a random variation of the previous plan to a best response plan that would maximize the traveler’s expected utility given that all other travelers retain their previously selected plans. Travelers want to switch to the candidate plan if this leads to an expected utility gain. Informally (this is made concrete below), if no traveler

Algorithm 1 Blueprint of a DTA simulation

1. Represent the scenario. This comprises the transport network and the travel demand. The network may be multi-modal. The travel demand is a collection of individual travelers.

2. Repeat the following process until a convergence criterion is satisfied (Nagel and Flötteröd, 2012).

   (a) Every traveler chooses a travel plan. In a discrete choice framework, this step comprises both the possible update of a plan choice set and the choice of one plan from that set (Ben-Akiva and Lerman, 1985).

   (b) All travelers are loaded onto the network. This means running a particle-based network flow simulator until all travelers have completed their daily travel. Traffic flow simulators come in many different guises (Barcelo, 2010).

   (c) Every traveler processes the information gathered during the most recent network loading. This may be as simple as memorizing past travel times, possibly averaged over multiple past iterations (Cascetta, 1989).

3. Extract solution statistics of interest. If variability in the process ceases and a point solution is attained, only the last iteration is of interest. If the process is stochastic and attains stationarity, one may compute means and variances over many stationary iterations.

can increase expected utility by switching to the candidate plan, then the assignment problem is considered solved.

DTA simulation practice has frequently resorted to variations of the method of successive averages (MSA; Blum, 1954), which is well-known for guaranteeing convergence when assigning certain classes of continuum flow models (Sheffi, 1985; Liu et al., 2007). In the particle-discrete case, it can be used to control the replanning probability per traveler or the total number of replanners per assignment iteration; in either case the frequency of replanning falls with one over the assignment iteration number. The studies of Levin et al. (2015) and Ameli et al. (2020) compare DTA assignment techniques including a range of MSA variations. Both present evidence that assigning i.i.d. replanning probabilities to each particle is inferior to the "sorting technique" of Sbayti et al. (2007) that replans in every assignment iteration the (MSA-controlled) share of particles with the highest expected utility improvement.

Summing the expected utility improvement over all particles gives rise to a gap function that is strictly non-negative and attains a zero value at the model's solution point. Mahut et al. (2008) and Himpe and Tampère (2016) discuss gap functions that are applicable in dynamic and even particle-discrete settings. Lu et al. (2009) tackle the assignment problem of Sbayti et al. (2007) within this setting. Smith (1984) adopts a dynamical systems approach to the flow swapping process and minimizes a Lyapunov (gap) function. Mounce and Carey (2011) analyze further developments of this approach. Most notable in the context of the present article is their interpretation of the projection method of Wu et al. (1998) as a path flow swapping rule that aims to dampen oscillations between iterations of the assignment process.

3 Method

Consider a population of $n = 1, \ldots, N$ travelers. Let $\mathcal{X}_n$ be the set of all possible travel plans available to traveler n, and let $\mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_N$ be the set of all possible population travel plans. Denote by $x_n \in \mathcal{X}_n$ a concrete travel plan of traveler n and by $x = [x_1, \ldots, x_N] \in \mathcal{X}$ the fixed-size string of all concrete travel plans in the population. The Hamming distance $d_H : \mathcal{X} \times \mathcal{X} \to \mathbb{N}_0$ between two population plans $x, y \in \mathcal{X}$ is measured by counting the number of individuals with different plans in x and y:

$$d_H(x, y) = \sum_{n=1}^{N} \mathbf{1}(x_n \neq y_n) \tag{1}$$

with $\mathbf{1}(\cdot)$ being the 0/1 indicator function. The population plans $\mathcal{X}$ are hence a subset of a metric space.[1]

Let $\mathcal{Z}$ be the set of all possible network conditions and let $Z(x) : \mathcal{X} \to \mathcal{Z}$ represent the network conditions resutling from a (possibly stochastic) network loading of the

[1] Extending the set of travel plans available to each agent to $\mathcal{X}_* = \bigcup_n \mathcal{X}_n$, a Hamming distance is defined over all size-N strings in $\mathcal{X}_* \times \ldots \times \mathcal{X}_*$. The fact that each individual traveler n only considers a subset $\mathcal{X}_n \subseteq \mathcal{X}_*$ merely means that the distance function d_H exists over a larger domain ($\mathcal{X}_* \times \ldots \times \mathcal{X}_*$) than what is considered here ($\mathcal{X}_1 \times \ldots \times \mathcal{X}_N$).

population plans x onto the network. Let $\nu_n(r_n, z) : \mathcal{X}_n \times \mathcal{Z} \to \mathbb{R}$ be the real-valued utility receive by traveler n when executing travel plan $r_n \in \mathcal{X}_n$ in fixed network conditions z. Let

$$u_n(r_n, x) = E_{Z(x)}\{\nu_n(r_n, Z(x))\} \tag{2}$$

be the expected utility received by traveler n when executing the travel plan r_n over the distribution $Z(x)$ of network conditions that results from loading the population plans x onto the network. Let

$$u_n^*(x) = \max_{r_n \in \mathcal{X}_n} u_n(r_n, x) \tag{3}$$

be the largest expected utility individual n anticipates from unilaterally changing plans.

Given $x \in \mathcal{X}$, let x_{-n} denote x with x_n omitted, and let $[r_n, x_{-n}] \in \mathcal{X}$ denote the population plans that result from replacing x_n by r_n in the nth position of x and leaving the reminder of x (i.e. x_{-n}) unchanged.

Assumption 1. *Let $x = [x_n, x_{-n}]$ and $y = [y_n, y_{-n}]$. There exist real-valued constants $C, D \geq 0$ such that the expected utility function u_n has the following properties, for all $n = 1, \ldots, N$ and $x, y \in \mathcal{X}$:*

$$|u_n(x_n, x) - u_n(x_n, y)| \leq C d_H(x, y) \tag{4}$$

$$|u_n^*(x) - u_n^*(y)| \leq D d_H(x, y) \tag{5}$$

where $|\cdot|$ denotes the absolute value function.

Assumption 1 resembles but deviates from that of Lipschitz continuity because d_H only takes integer values. This integrality implies that Assumption 1 is not stronger than assuming a finite value range of u_n: For $x = y$, both inequalities collaps into the expression $0 \leq 0$. If at least one plan differs between x and y, one has $d_H(x, y) \geq 1$ meaning that a sufficient condition to satisfy (4),(5) is to replace their right-hand sides by C resp. D, and then the existence of finite C, D is ensured by finite utility values on the left-hand sides. – Assumption 1 encodes the intuition that reducing the change in (population plans generating the) network conditions may also reduce the change in the expected utility of given plans, cf. (4), and of the utilities resulting from optimal reactions to the network conditions, cf. (5).

Let

$$g_n(r_n, x) = u_n(r_n, x) - u_n(x_n, x) \tag{6}$$

be the utility gain expected by traveler n from switching from plan x_n to plan r_n given that all other travelers keep their plans. The individual best-response gap $g_n^*(x) : \mathcal{X} \to \mathbb{R}$ and the population best-response gap $g^*(x) : \mathcal{X} \to \mathbb{R}$ are defined through

$$g_n^*(x) = u_n^*(x) - u_n(x_n, x) \tag{7}$$

$$g^*(x) = \sum_n g_n^*(x). \tag{8}$$

$g_n^*(x)$ measures the maximum expected utility gain that traveler n anticipates to achieve by unilaterally changing its travel plan. $g^*(x)$ is the corresponding population sum.

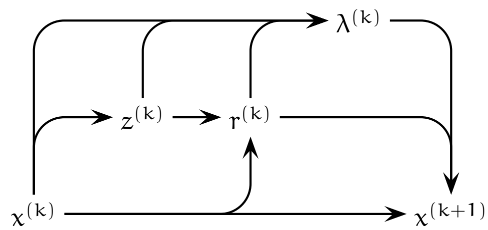


Figure 1: Stochastic process assignment

Definition 1. *The population plans $x^{**} \in \mathcal{X}$ exactly solve the traffic assignment problem if the resulting population best response gap is zero:*

$$g^*(x^{**}) \quad = \quad 0. \tag{9}$$

This means that no traveler anticipates an improvement of its expected utility from unilateraly switching to a different travel plan. The existence of such a solution depends on the concrete modeling assumptions encoded in the simulator. For broad applicability, and to bypass the need to validate the existence of a solution in in the sense of Definition 1, the following, weaker formulation is adopted.

Definition 2. *The population plans x^* optimally approximate a solution to the traffic assignment problem if the resulting expected population best response gap is minimal:*

$$g^*(x^*) \quad = \quad \min_{x \in \mathcal{X}} g^*(x). \tag{10}$$

Since g^* is non-negative and attains zero at exact solutions, this gap minimization recovers exact solutions if they exist. An operational problem formuation is subsequently constructed that aims to minimize the expected population best response gap in the sense of Definition 2. From Assumption 1 follows

$$|g_n^*([x_n, x_{-n}]) - g_n^*([x_n, y_{-n}])| \quad \leq \quad (C + D) d_H(x_{-n}, y_{-n}), \tag{11}$$

which implies

$$g_n^*([x_n, x_{-n}]) \quad \leq \quad g_n^*([x_n, y_{-n}]) + (C + D) d_H(x_{-n}, y_{-n}). \tag{12}$$

The iterative traffic assignment process may now be considered as a time-inhomogeneous Markov chain with state space $\mathcal{X}$ and discrete time index k, cf. Figure 1: Given the population plans $x^{(k)}$ in assignment iteration k, a (possibly stochastic) network loading yields the realized network conditions $z^{(k)}$. The new population plans $x^{(k+1)}$ result from a subsequently detailed two-stage procedure of (i) receiving new candidate plans $r^{(k)} = (r_n^{(k)})$ for all travelers and (ii) deciding based on binary indicator variables $\lambda^{(k)} = (\lambda_n^{(k)})$ which of the new plans to take over into $x^{(k+1)}$.

Specifically, letting $r_n^{(k)} \in \mathcal{X}_n$ be a new plan computed by traveler n in iteration k, the following mechanism for generating $x^{(k+1)} = (x_n^{(k+1)})$ is postulated:

$$x_n^{(k+1)} \quad = \quad \begin{cases} r_n^{(k)} & \text{if } \lambda_n^{(k)} = 1 \\ x_n^{(k)} & \text{if } \lambda_n^{(k)} = 0 \end{cases} \tag{13}$$

with the replanning indicator $\lambda_n^{(k)} \in \{0, 1\}$. This means that traveler n keeps the old plan $x_n^{(k)}$ if $\lambda_n^{(k)} = 0$ and otherwise switches to $r_n^{(k)}$. Inserting (6) and (13) into (12) yields

$$g^*(x^{(k+1)}) \;\leq\; -\sum_n \lambda_n^{(k)} g_n(r_n^{(k)}, x^{(k)}) + (C + D)(N - 1) \sum_n \lambda_n^{(k)} + g^*(x^{(k)}). \tag{14}$$

The subsequently pursued assignment logic aims, in every iteration k of the assignment process, to select replanning indicators that minimize the upcoming iteration's gap $g^*(x^{(k+1)})$, i.e. the left-hand side of (14). Since $g^*(x^{(k+1)})$ is not yet known in iteration k, the approach is to reduce the right-hand-side of (14), which constitutes an upper bound on $g^*(x^{(k+1)})$.

Omitting the respect to the replanning indicators constant term $g^*(x^{(k)})$, a weighted sum of two terms remains on the right-hand side:

- $\sum_n \lambda_n^{(k)} g_n(r_n^{(k)}, x^{(k)})$ represents the expected utility gains of all plan-switching travelers. Its negative sign in (14) means that minimizing the right-hand side encourages replanning.

- $\sum_n \lambda_n^{(k)}$ is the total number of replanners. It is weighted with $(C + D)(N - 1)$ and acts in (14) as a penalty term that discourages plan switching.

The parameters C and D are unknown. When estimating them, Assumption 1 is satisfied as long as they are chosen sufficiently large. However, functioning as a replanning penalty weight, setting them to arbitrarily large values could impede all replanning. One may hence opt for gradually increasing estimators C and D over the assignment iterations k, expecting this to eventually reach values that comply with (4) while still allowing for replanning.

Observing that the only effect of increasing C and D is to reduce the number of replanners, this effect can just as well be achieved by explicitly limiting that number. This leads to the following formulation:

$$\begin{aligned} \min_{\lambda^{(k)} \in \{0,1\}^N} \quad & -\sum_n \lambda_n^{(k)} g_n(r_n^{(k)}, x^{(k)}) \\ \text{s.t.} \quad & \sum_n \lambda_n^{(k)} \leq \Delta^{(k)} \end{aligned} \tag{15}$$

with $\Delta^{(k)}$ the maximum allowed number of replanners in assignment iteration k.

The problem (15) is solved by (i) sorting all travelers according to their expected gain $g_n(r_n^{(k)}, x^{(k)})$ and (ii) letting the $\Delta^{(k)}$ travelers with the largest (positive) gains replan. This logic coincides (apart from using the expected gap instead of a realization thereof) with the "sorting technique" of Sbayti et al. (2007). Beyond this, it has been derived from mild and operational mathematical assumptions.

Given the unavailability of closed-form distributional information, a numerical approximation of the expected value $g_n(r_n^{(k)}, x^{(k)})$ in (15) is needed. With $z^{(k-r)}$ a realization of $Z(x^{(k-r)})$ and $\mathcal{R} = \{0, \ldots, R-1\}$, the following estimator is constructed:

$$\hat{u}_n(r_n, x; \mu, \mathcal{R}) = \sum_{r \in \mathcal{R}} \frac{\exp\left(-\mu d_H(x, x^{(k-r)})\right)}{\sum_{s \in \mathcal{R}} \exp\left(-\mu d_H(x, x^{(k-s)})\right)} v_n(r_n, z^{(k-r)}) \quad (16)$$

$$\hat{g}_n(r_n, x; \mu, \mathcal{R}) = \hat{u}_n(r_n, x; \mu, \mathcal{R}) - \hat{u}_n(x_n, x; \mu, \mathcal{R}). \quad (17)$$

This is a weighted sum of utility functions v_n, with the weights decaying exponentially with the distance between the population plans having generated the previous network condition realization (i.e. $x^{(k-r)}$ having generated $z^{(k-r)}$) and those plans for which an expectation is to be approximated (i.e. x). The function $\hat{g}_n(r_n, x)$ is a Kernel smoother over $\mathcal{X}$ that additionally is parameterized by $r_n \in \mathcal{X}_n$.

The parameter $\mu^{(k)}$ is estimated in every assignment iteration k as follows:

$$\begin{aligned} \mu^{(k)} = \arg\min_{\mu \geq 0} \sum_{n=0}^{N-1} \sum_{r \in \mathcal{R}} \Big(& v_n(r_n^{(k-r)}, z^{(k-r)}) - v_n(x_n^{(k-r)}, z^{(k-r)}) \ldots \\ & - \hat{g}_n(x_n^{(k-r)}, x^{(k-r)}; \mu, \mathcal{R} \backslash \{r\}) \Big)^2 . \end{aligned} \quad (18)$$

Noting that $v_n(r_n^{(k-r)}, z^{(k-r)}) - v_n(x_n^{(k-r)}, z^{(k-r)})$ is the realization of a random variable with expectation $g_n(r_n^{(k-r)}, x^{(k-r)})$, this identifies a $\mu^{(k)}$ value that, on average over all travelers n and past iterations $k-r$, reproduces this expectation as closely as possible. This is a leave-one-out estimator.[2]

4 Results and discussion

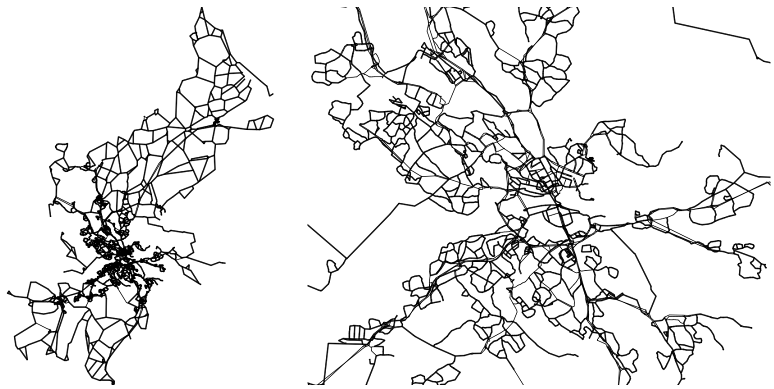
The above method development is the main result of this article. The present section experimentally illustrates this result. As a case study, the Greater Stockholm region in Sweden is considered. Figures 2(a) and (b) display the network. Only car traffic is considered; a 24-hour dynamic network loading is computed by a queueing simulation that approximates a particle-discrete kinematic wave model. The conflict resolution logic of its intersection model is stochastic. Travel behavioral degrees of freedom are route and departure time choice. Proposal routes are generated by computing time-dependent shortest paths against the previous assignment iteration; proposal departure times are generated by random variations.

We adopt from the recent study of Ameli et al. (2020) the following methods, which are applicable in the considered, particle based setting:

I.i.d. Given a replanning rate λ, each particle independently switches to its proposal plan with identical probability λ.

Gap-prop. Each particle computes the ratio of its current gap (i.e. expected cost reduction) over the current cost (negative utility) of travel, multiplies this ratio

[2] If one evaluated $\hat{g}_n(x_n^{(k-r)}, x^{(k-r)}; \mu, \mathcal{R} \backslash \{r\})$ based on all past realizations $\mathcal{R}$, i.e. without excluding the lag r, then trivial solution $\mu = \infty$ would yield a zero objective function in (18) but not yield any smoothing over the simulation noise.



(a) entire network (b) central network

Figure 2: Stockholm network

with the replanning rate λ, and uses the resulting number as the probability of switching to its proposal plan.

Sbayti(mem) The proposed adaptation of Sbayti et al. (2007)'s algorithm, with "mem" representing the memory variable R that defines over how many averaging iterations the expected gap is approximated.

These methods are organized in the columns of Figure 3. All algorithms are evaluated for three different congestion levels: *low*, *medium* (the result of an approximate model calibration), and *high*. The congestion levels are organized in the rows of the following figures. All combinations of algorithms and congestion levels are evaluated for two different replanning rate schedules: *MSA* (λ falls with one over the iteration number) and *sqrt* (λ falls with one over the square root of the iteration number). A discussion of the *exp* schedule is omitted due to space restrictions. The schedules are differentiated by color coding. The following trends can be observed: Not surprisingly, gap levels increase with congestion levels. The *MSA* step size rule performs systematically unfavorably compared to the *sqrt* rule. *Sbayti* reaches lower gaps than *I.i.d.* and *Gap-prop*. Increasing the *Sbayti* memory from 1 to 10 iterations yields a clear improvement, but going beyond that and using a memory of 20 iterations has no clear advantage.

5 Conclusion

An exact treatment of the particle-based and stochastic traffic assignment problem is, without simplifying assumptions that would no longer cover existing simulator implementations, unavailable. Still, the increasing use of simulators in transport planning calls for further analysis. Given the wide spectrum of possible simulators, such an analysis can only be based on general, broad and hence rather simple assumptions. The present article delivers such an analysis for the widely used "sorting technique" of Sbayti et al. (2007) and demonstrates how such an analysis can support further al-

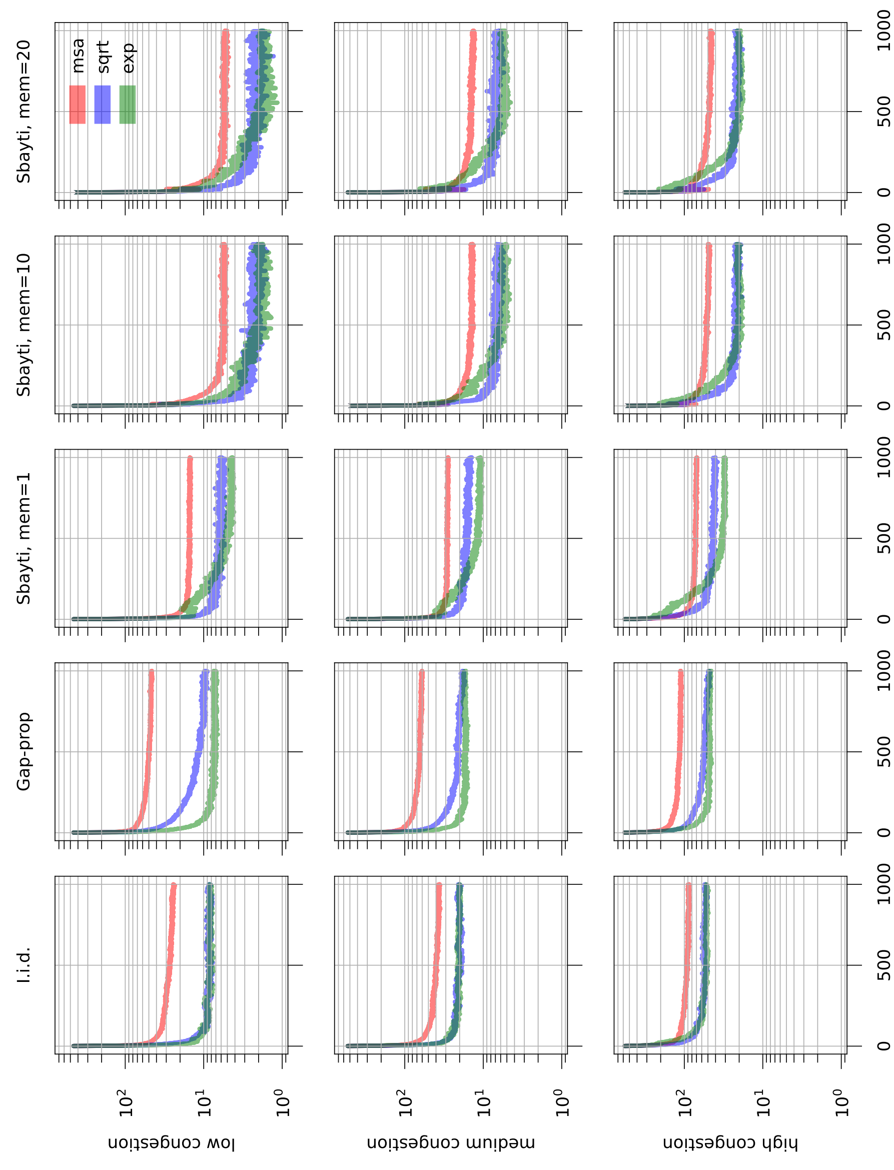


Figure 3: Gap over assignment iteration, 1000 iterations

gorithmic development. Experimental results indicate the practical relevance of the approach.

References

Ameli, M., Lebacque, J.-P. and Leclercq, L. (2020). Cross-comparison of convergence algorithms to solve trip-based dynamic traffic assignment problems, *Computer-Aided Civil and Infrastructure Engineering* **35**(3): 219–240.

Barcelo, J. (2010). *Fundamentals of Traffic Simulation*, Springer.

Ben-Akiva, M. and Lerman, S. (1985). *Discrete Choice Analysis*, MIT Press series in transportation studies, The MIT Press.

Blum, J. R. (1954). Multidimensional stochastic approximation methods, *Ann. Math. Statist.* **25**(4): 737–744.

Cascetta, E. (1989). A stochastic process approach to the analysis of temporal dynamics in transportation networks, *Transportation Research Part B* **23**(1): 1–17.

Farooq, B., Bierlaire, M., Hurtubia, R. and Flötteröd, G. (2013). Simulation based population synthesis, *Transportation Research Part B: Methodological* **58**(C): 243–263.

Friesz, T. L., Han, K. and Bagherzadeh, A. (2021). Convergence of fixed-point algorithms for elastic demand dynamic user equilibrium, *Transportation Research Part B: Methodological* **150**: 336–352.

Han, K., Eve, G. and Friesz, T. L. (2019). Computing dynamic user equilibria on large-scale networks with software implementation, *Networks and Spatial Economics* **19**(3): 869–902.

Himpe, W. and Tampère, C. (2016). A dynamic user equilibrium algorithm that exploits warm starting capabilities of the iterative link transmission model, *International Symposium on Dynamic Traffic Assignment, Sydney, Australia*.

Levin, M., Pool, M., Owens, T., Juri, N. and Waller, S. (2015). Improving the convergence of simulation-based dynamic traffic assignment methodologies, *Networks and Spatial Economics* **15**(3): 655–676.

Liu, H., He, X. and He, B. (2007). Method of successive weighted averages (MSWA) and self-regulated averaging schemes for solving stochastic user equilibrium problem, *Networks and Spatial Economics* **9**: 485–503.

Lu, C.-C., Mahmassani, H. S. and Zhou, X. (2009). Equivalent gap function-based reformulation and solution algorithm for the dynamic user equilibrium problem, *Transportation Research Part B* **43**: 345–364.

Mahut, M., Florian, M. and Tremblay, N. (2008). Comparison of assignment methods for simulation-based dynamic-equilibrium traffic assignment, *Proceeding of the Transportation Research Board 87th Annual Meeting*, WASHINGTON DC, United States.

Mounce, R. and Carey, M. (2011). Route swapping in dynamic traffic networks, *Transportation Research Part B: Methodological* **45**(1): 102–111.

Nagel, K. and Flötteröd, G. (2012). Agent-based traffic assignment: going from trips to behavioral travelers, *in* R. Pendyala and C. Bhat (eds), *Travel Behaviour Research in an Evolving World*, Emerald Group Publishing, Bingley, United Kingdom, chapter 12, pp. 261–293.

Sbayti, H., Lu, C.-C. and Mahmassani, H. (2007). Efficient implementation of method of successive averages in simulation-based dynamic traffic assignment models for large-scale network applications, *Transportation Research Record* **2029**: 22–30.

Sheffi, Y. (1985). *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*, Prentice-Hall.

Smith, M. (1984). The stability of a dynamic model of traffic assignment-an application of a method of lyapunov, *Transportation Science* **18**(3): 245–252.

Wang, Y., Szeto, W., Han, K. and Friesz, T. L. (2018). Dynamic traffic assignment: A review of the methodological advances for environmentally sustainable road transportation applications, *Transportation Research Part B: Methodological* **111**: 370–394.

Wu, J., Florian, M., Xu, Y. and Rubio-Ardanaz, J. (1998). A projection algorithm for the dynamic network equilibrium problem, *Proceedings of the 1998 Iternational Conference on Traffic and Transportation Studies*, American Society of Civil Engineers, Beijing, China.