




dyntapy: dynamic and static traffic assignment in Python

Paul Ortmann ¹ and Chris M. J. Tampère ¹

¹ KU Leuven, Belgium  Corresponding author

DOI: [10.21105/joss.04593](https://doi.org/10.21105/joss.04593)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Hugo Ledoux](#)  

Reviewers:

- [@liberostelios](#)
- [@EricaEgg](#)

Submitted: 13 June 2022

Published: 13 September 2022

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Statement of need

Macroscopic Traffic Models, based on the idea that traffic in a city can be modelled much like a fluid, have been used to guide transport policy and mitigate congestion for more than 50 years. They find a fixed point in the interaction between two phenomena: (i) drivers make routing decisions based on the observed travel times and try to minimize their experienced travel times; (ii) route flows determine the congestion pattern and affect travel times. This fixed point may be further generalized to include choice of departure time, transport mode, activity location, and other behavioral dimensions. Building useful transport models is a balancing act between capturing realistic behavior and managing the computational complexity of models that are routinely used to simulate the journeys of thousands of cars simultaneously. Authors have created a variety of flavors of these models that emphasize preferences for one or another aspect of realism at the cost of others, see Bliemer et al. (2017) for a recent overview and classification.

Independent of the algorithm used, there are commonalities between the software that is used to model car traffic macroscopically. Namely, road network- and travel demand processing, representation of the same as efficient computational entities, and the visualization and processing of results. This is part of what dyntapy provides. Additionally, a selection of static and dynamic traffic assignment algorithms have been implemented and can be used by researchers in transport modeling to conduct studies in which the interaction with an assignment model and/or the accurate modeling of congestion is of concern. Commercial offerings are often expensive and do not allow one to make adjustments to the core of the algorithms. dyntapy is a useful starting point for studies that require such manipulation.

dyntapy also lends itself to educational use and has already been used in traffic engineering courses at KU Leuven.

Summary

dyntapy is a python package for modeling car traffic macroscopically. It provides a few key features to researchers in traffic modeling: (i) extraction and formatting of road network graphs from OpenStreetMap using OSMnx (Boeing, 2017), following the GMNS standard (General Modeling Network Specification: <https://github.com/zephyr-data-specs/GMNS>); (ii) parsers for road network and travel demand information; (iii) fast shortest path computations; (iv) both static and dynamic traffic assignment algorithms; (v) selected link analysis¹; and (vi) flexible visualization tools.

The toolkit is designed to be extensible. The provided object-oriented specification of the road network, travel demand, and results are generic and allow for other assignment algorithms to

¹as of now, merely for some static assignments

be added without much additional effort. Because there is a shared definition of the modeling results, visualization tools can be used without modification for different algorithms.

dyntapy utilizes Networkx (Hagberg et al., 2008) for managing the road network graph. Numpy (Harris et al., 2020) is used for fast array calculations and Numba (Lam et al., 2015) is used to accelerate computations that involve graph traversal via sparse matrices.

Bokeh² powers the flexible visualization of static and dynamic³ assignment results, see Figure 1 and Figure 2 below.

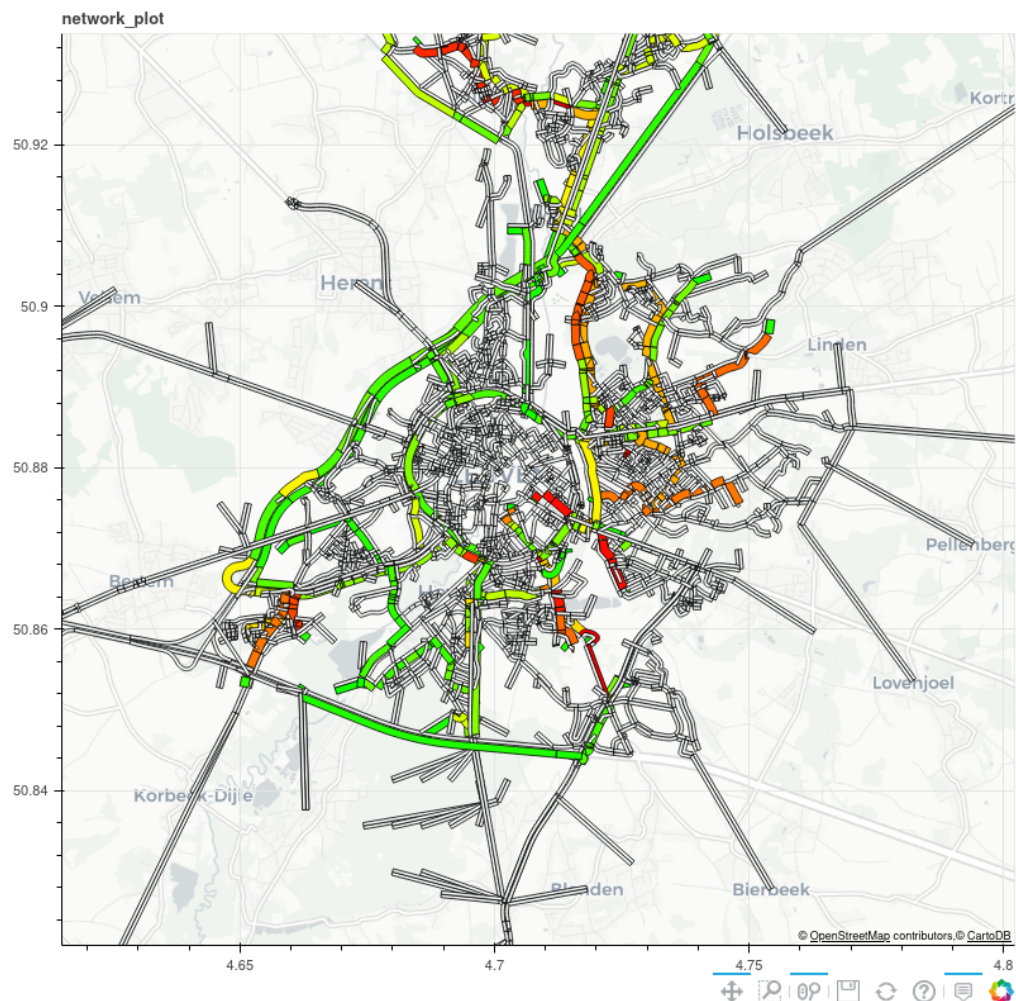


Figure 1: Visualizing the solution to a deterministic Traffic Assignment Problem.

²<https://bokeh.org>

³Examples on the visualization of dynamic attributes can be found in the repository.

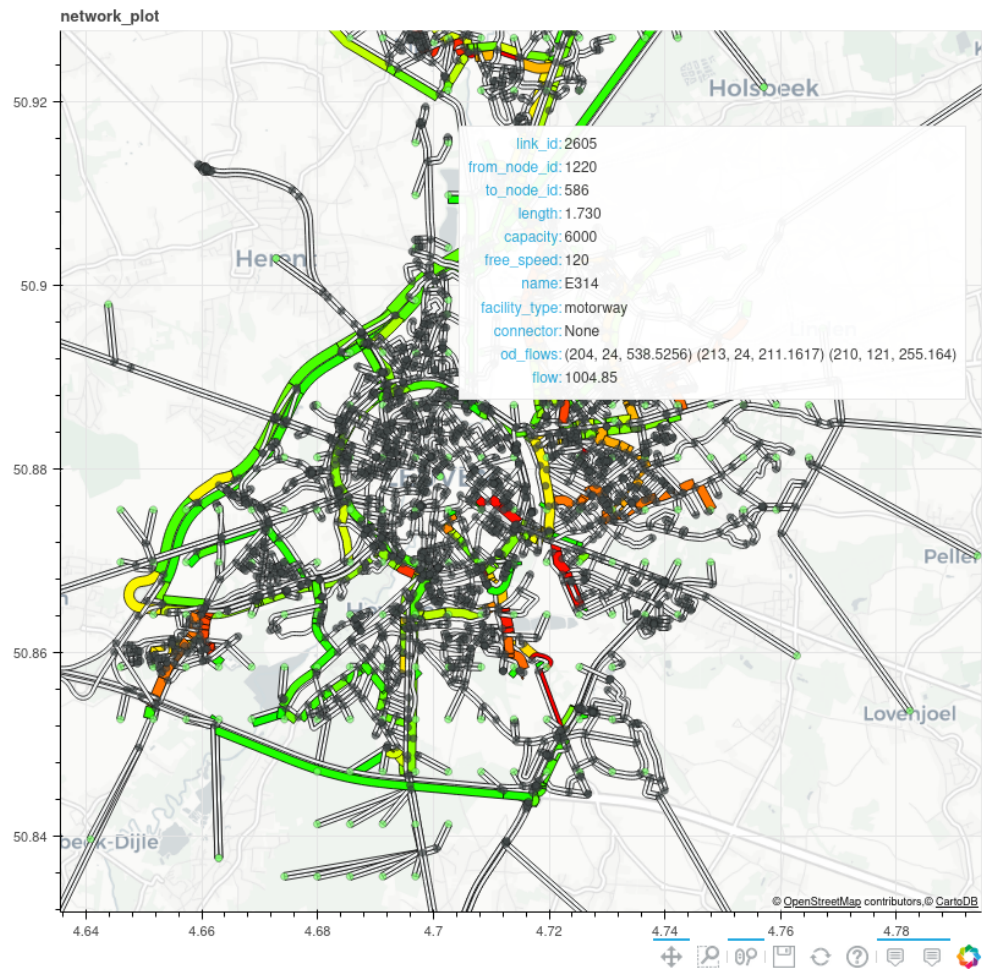


Figure 2: Link and node attributes can be inspected by hovering. The user can define which attributes to display.

We provide two algorithms for solving the deterministic Traffic Assignment Problem, namely, Dial's Algorithm B (Dial, 2006) and the Method of Successive Averages. For solving the stochastic uncongested Traffic Assignment Problem, we use Dial's Algorithm (Dial, 1971). For the deterministic Dynamic Assignment Problem we utilize the iterative link transmission model (Himpe et al., 2016), as dynamic network loading component, an iterative route choice component introduced in Himpe (2016) based on quasi-reduced projection (Mahut et al., 2008) and travel time calculations of the MLI type as introduced in Long et al. (2011).

We welcome contributions by other authors to add to this selection. In the short term, the stochastic Dynamic Traffic Assignment model that was introduced in Verstraete & Tampère (2021) will be added to the list of available algorithms.

dyntapy is being used for scenario analysis as part of DUET (Raes et al., 2021), a platform for digital twins of cities in which policy makers can explore the impact of road network changes on congestion and emissions.

For usage examples and tutorials we refer the reader to the [dyntapy documentation](#).

Related packages

[AequilibraE](#) is a traffic modeling toolkit that offers a set of static assignments (Method

of Successive Averages and different variants of the Frank-Wolfe Algorithm) paired with OpenStreetMap parsing. It is an open source alternative to commercial transport modeling software and provides comprehensive visualization and network management capabilities in QGIS. dyntapy differs from AequilibraE in functionality by its inclusion of dynamic models and command-driven visualization capabilities within Jupyter notebooks.

Similar to other popular graph packages, such as igraph (Csardi & Nepusz, 2006) and NetworkX, we offer shortest path computations. Our implementations are more than twice as fast as igraph and more than 20x faster than Networkx for road networks.

Outlook and use cases

In its current state, dyntapy is most interesting for assignment researchers, transport modelers, and students who want to get hands-on experiments with assignments.

Macro- and mesoscopic commercial modeling software offer some functionalities that are beyond what dyntapy provides and make it less suitable for studies in which realism is of chief concern. Namely, (i) support for specifying intersections in more details and calculating delays based on signals and priorities; and (ii) calibration routines for supply and demand for both static and dynamic assignments⁴.

Support for intersection data both in the parser and as constraints in the models is a long term goal for dyntapy; however, data for intersections is currently not widely available. OpenStreetMap's encoding of intersections is not suitable for use in traffic models, partly because of its ambiguity, but also because it does not allow for specifying signal timings. GMNS fills this gap; however, the number of datasets available in this standard is still very limited.

Regarding (ii), there are currently no plans to add calibration to dyntapy. Intersections are the main causes of queuing in traffic models. Before we can hope to calibrate models we need to have more certainty about the specification of intersections.

Acknowledgements

This research has been funded by DUET - Digital Urban European Twins as part of the Horizon 2020 funding program (Grant ID 870607). Many thanks to Jeroen Verstraete and Willem Himpe for their feedback on some of the design decisions that have been made in dyntapy.

References

- Bliemer, M. C., Raadsen, M. P., Brederode, L. J., Bell, M. G., Wismans, L. J., & Smith, M. J. (2017). Genetics of traffic assignment models for strategic transport planning. *Transport Reviews*, 37(1), 56–78. <https://doi.org/10.1080/01441647.2016.1207211>
- Boeing, G. (2017). OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems*, 65, 126–139. <https://doi.org/10.1016/j.compenvurbsys.2017.05.004>
- Csardi, G., & Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal, Complex Systems*, 1695. <https://igraph.org>
- 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. <https://doi.org/10.1016/j.trb.2006.02.008>

⁴note that in dynamic assignment, calibration remains an open research question

- Dial, R. B. (1971). A probabilistic multipath traffic assignment model which obviates path enumeration. *Transportation Research*, 5(2), 83–111. [https://doi.org/10.1016/0041-1647\(71\)90012-8](https://doi.org/10.1016/0041-1647(71)90012-8)
- Hagberg, A. A., Schult, D. A., & Swart, P. J. (2008). Exploring network structure, dynamics, and function using NetworkX. In G. Varoquaux, T. Vaught, & J. Millman (Eds.), *Proceedings of the 7th python in science conference* (pp. 11–15).
- Harris, C. R., Millman, K. J., Walt, S. J. van der, Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., Kerkwijk, M. H. van, Brett, M., Haldane, A., Río, J. F. del, Wiebe, M., Peterson, P., ... Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825), 357–362. <https://doi.org/10.1038/s41586-020-2649-2>
- Himpe, W. (2016). *Integrated algorithms for repeated dynamic traffic assignments* [PhD thesis]. KU Leuven.
- Himpe, W., Corthout, R., & Tampère, M. C. (2016). An efficient iterative link transmission model. *Transportation Research Part B: Methodological*, 92, 170–190. <https://doi.org/10.1016/j.trb.2015.12.013>
- Lam, S. K., Pitrou, A., & Seibert, S. (2015). Numba: A LLVM-based Python JIT compiler. *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC - LLVM '15*, 1–6. <https://doi.org/10.1145/2833157.2833162>
- Long, J., Gao, Z., & Szeto, W. Y. (2011). Discretised link travel time models based on cumulative flows: Formulations and properties. *Transportation Research Part B: Methodological*, 45(1), 232–254. <https://doi.org/10.1016/j.trb.2010.05.002>
- Mahut, M., Florian, M., & Tremblay, N. (2008). Comparison of assignment methods for simulation-based dynamic-equilibrium traffic assignment. *Proceeding of the Transportation Research Board 87th Annual Meeting (DVD), Washington, DC*.
- Raes, L., Michiels, P., Adolphi, T., Tampere, C., Dalianis, T., McAleer, S., & Kogut, P. (2021). DUET: A framework for building secure and trusted digital twins of smart cities. *IEEE Internet Computing*. <https://doi.org/10.1109/mic.2021.3060962>
- Verstraete, J., & Tampère, C. M. J. (2021). Consistent iterative algorithm for stochastic dynamic traffic assignment with a stable route set. *Computer-Aided Civil and Infrastructure Engineering*, n/a(n/a). <https://doi.org/10.1111/mice.12781>