

# Infrastructure adaptation and emergence of loops in network routing with time-dependent loads

Alessandro Lonardi<sup>1</sup>, Enrico Facca<sup>2</sup>, Mario Putti<sup>3</sup> and Caterina De Bacco<sup>1</sup>

1. Max Planck Institute for Intelligent Systems, Cyber Valley, Tübingen 72076, Germany
2. Laboratoire Paul Painlevé, UMR No. 8524, CNRS, Inria, Université Lille, 59000 Lille, France
3. Department of Mathematics “Tullio Levi-Civita”, University of Padua, Via Trieste 63, 35131 Padua, Italy

Efficient and resilient public transportation that meet the ever-changing demands of users is critical for human-centric cities of the future. Nevertheless, routing algorithms often neglect the time-varying nature of passengers inflows in transport networks. We propose a method that, given time-dependent entry and exit rates of passengers in nodes, outputs edge flows which jointly optimize energy travelling expenditure and infrastructure cost. In particular, we leverage the formalism of capacitated networks, hence we treat passengers’ flows as electrical currents, to find efficient updates of the edge capacities—measuring the infrastructure size needed to allocate passengers. Our key hypothesis is that capacities evolve on a slower timescale than loads of passengers. The physical intuition is that a network manager, who is responsible to build the network infrastructure, has a coarser observation scale of a transportation system than the users, whose paths rapidly fluctuate. In practice, this means that modifications in the network infrastructure occur on a much larger timescale than that of passengers’ fluctuations. Leveraging this assumption, we translate the minimization problem into a closed-form system of differential equations that governs the evolution of capacities and passengers flows, given their time-varying entry and exit rates.

In our study, we formally prove that if passengers’ loads are periodic, the evolution of the system is controlled by a matrix obtained with the Fourier coefficients of the input loads. Remarkably, we find a sufficient condition on these coefficients to determine if the resulting networks are trees or have loops. Numerically, we display how the coefficients can be used as a proxy to evaluate the robustness of the outputted transport network. Furthermore, we propose a candidate Lyapunov functional for our dynamical system, which allows us to interpret stationary topologies as optimal networks, i.e., structures minimizing the global cost to build the graph. Not only this result is of theoretical significance, but it also offers a remarkable computational speedup. In fact, our algorithm efficiently extracts optimal networks that can otherwise only be obtained by letting the capacities evolve with the same timescale of passengers’ inflows, and averaging their values at large times. We validate our method on numerous synthetic topologies, and demonstrate its practical relevance by applying it to the Bordeaux bus network.

N-STARK (open-source algorithm implementation): <https://github.com/aleable/N-STARK>

The published manuscript can be found at: <https://journals.aps.org/pre/abstract/10.1103/PhysRevE.107.024302>

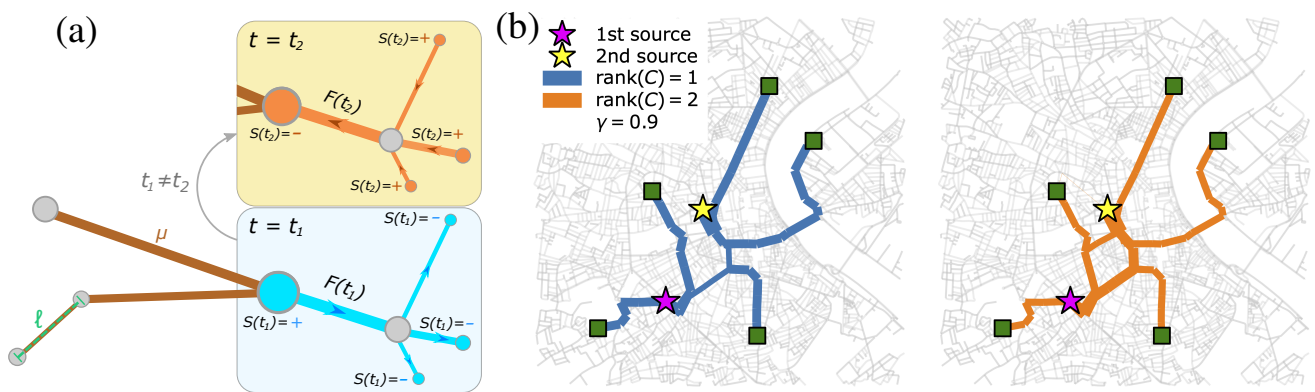


Figure 1: **Model setup and results on the Bordeaux bus network.** (a) Schematic visualization of the problem. Brown capacities ( $\mu$ ) are outputted by the model based the edge lengths ( $\ell$ ), and on the passengers’ loads ( $S(t)$ ). The latter change in time and generate time-dependent edge flows ( $F(t)$ ). In the two scenarios in the blue and yellow panels, fluxes move from the central part of the network to its periphery for  $t = t_1$ , and vice versa for  $t = t_2 \neq t_1$ . (b) The optimal transport network of Bordeaux buses for two origin nodes (stars) and five destination nodes (squares) changes based on the nature of the time-varying and periodic passengers’ loads. Particularly, for two configurations of the loads’ Fourier’s coefficients (blue and orange), buses reroute differently on the roads of Bordeaux. Here, edges are proportional to edge flows.