Simultaneous Localization and Mapping: Through the Lens of Nonlinear Optimization

Chih-Yuan Chiu, Amay Saxena, Joseph Menke, Shankar Sastry

https://cps-vo.org/group/forces

Motivation: Simultaneous Localization and Mapping (SLAM) algorithms perform visual-inertial estimation via filtering or batch optimization methods. Here, we present a unified optimization-based framework for landmark-based SLAM. We mathematically prove that state-of-the-art filtering methods (EKF SLAM, MSCKF, etc.) correspond to specific design choices in our generalized framework. Finally, we reformulate the MSCKF using our framework, implement the reformulation on challenging image sequences in a baseline SLAM dataset in simulation, and use the proposed re-interpretation to contrast the performance characteristics of these two classes of algorithms.

Challenge settings for the SLAM problem:

1. Agile robots or dynamic environments
2. Robust performance
3. Resource awareness

Scientific impact:

Deployment of SLAM-algorithmic methods to important civilian and military applications (e.g., map construction, search-and-rescue missions)

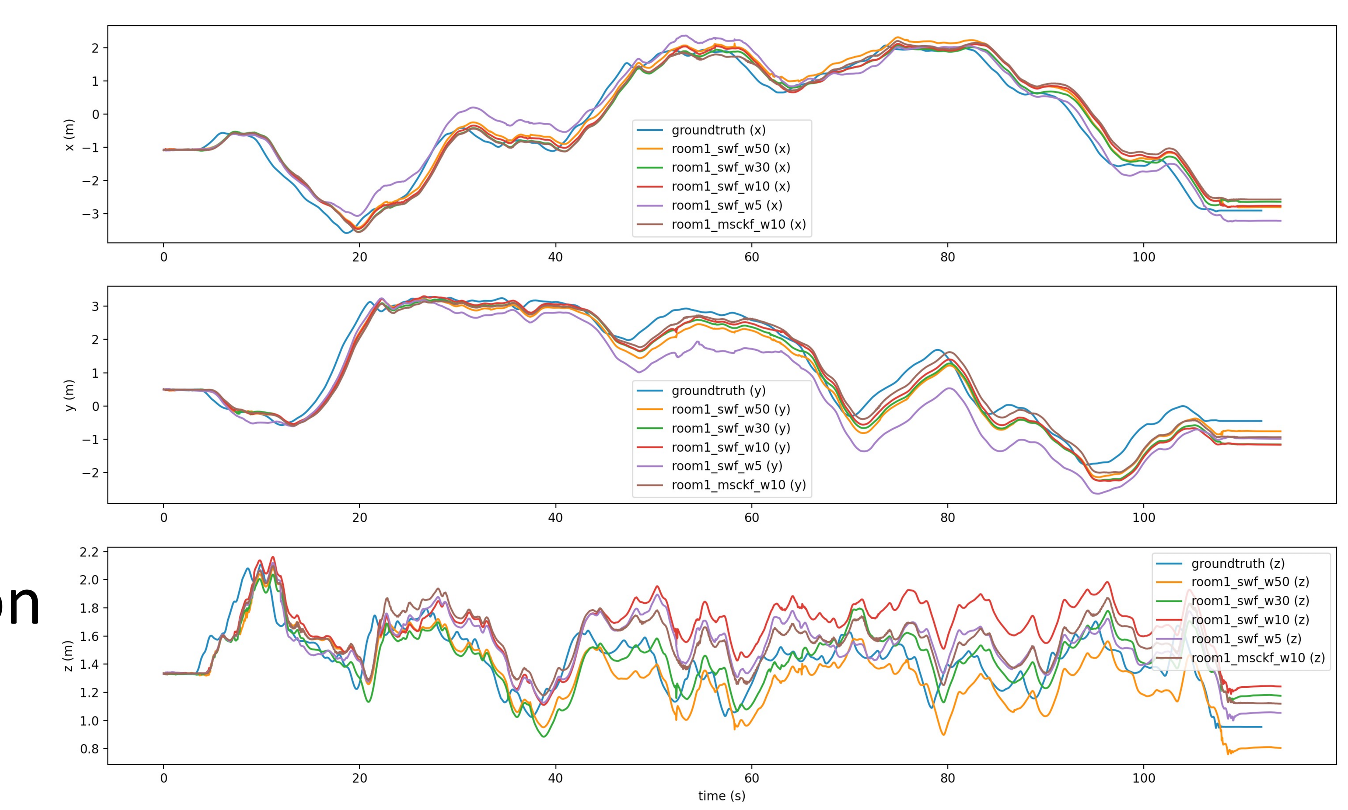
In the SLAM problem, a robotic agent attempts to localize itself within an unknown environment while mapping said environment.

Methodology:

- ❑ Generalized sliding window nonlinear optimization-based framework for SLAM algorithms, on Euclidean spaces and on manifolds.
- ❑ 3 steps --- Cost construction, Gauss-Newton steps, Marginalization steps
- ❑ Popular filtering-based SLAM algorithms (e.g., EKF SLAM and MSKCF) are sliding window filters with specific choices of marginalization schemes.
- ❑ Filtering and optimization-based SLAM algorithms can be directly compared.
- ❑ Our framework allows the design of new algorithms whose computational speed and performance flexibly interpolate the state-of-the-art.

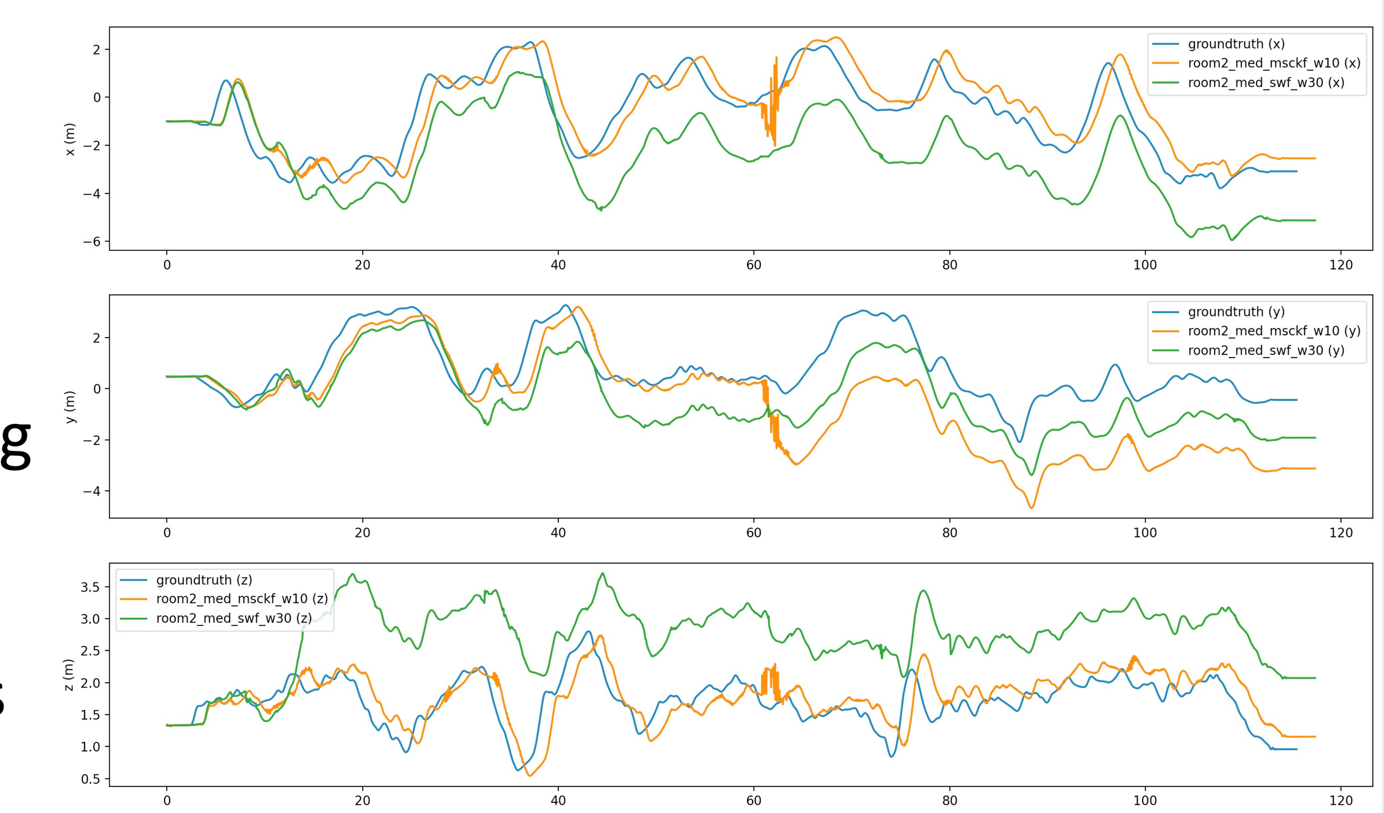
Simulation Setup:

- ❑ Dataset --- Euroc MAV of images taken from a micro aerial vehicle.
- ❑ Dynamics model --- post-integration of data from an on-board IMU.
- ❑ Image measurement model --- stereo camera pair + standard pinhole projection
- ❑ Backend --- GTSAM with C++.



Simulation Results:

- ❑ Localization results for sliding window filters and MSCKF.
- ❑ Data --- Vicon rooms 2 01 (easy, top figure) and 2 02 (harder, bottom figure).
- ❑ Conclusion --- MSCKF outperforms sliding window filters with large window size, despite not performing multiple nonlinear feature measurement updates through Gauss-Newton steps.



Conclusion and ongoing work

- ❑ Nonlinear optimization framework generalizes and allows flexible implementation of SLAM algorithms.
- ❑ Dynamic SLAM – Applying our framework to highly dynamic environments (e.g., rapidly moving features).

Potential broader impact (societal):

- ❑ Improved safety and efficiency of robotic agents deployed in the real world.
- ❑ Unifying framework for understanding, comparing, and improving existing state-of-the-art SLAM algorithms.

Education and Outreach:

Researchers in this project provided SLAM tutorials to the following courses to the EECS department at UC Berkeley --- EECS 106B / 206 B (Spring 2019, Spring 2021), EE 290-005 (Spring 2021).

2021 NSF Cyber-Physical Systems Principal Investigators' Meeting

June 2-4, 2021

Decentralized Dynamic Motion Planning for Nonholonomic Vehicles

Valmik Prabhu, S Shankar Sastry

FORCES: Foumndations of Resilient Cyber Physical Systems

NSF Grant CNS 1545126

https://cps-vo.org/group/forces

Challenge:

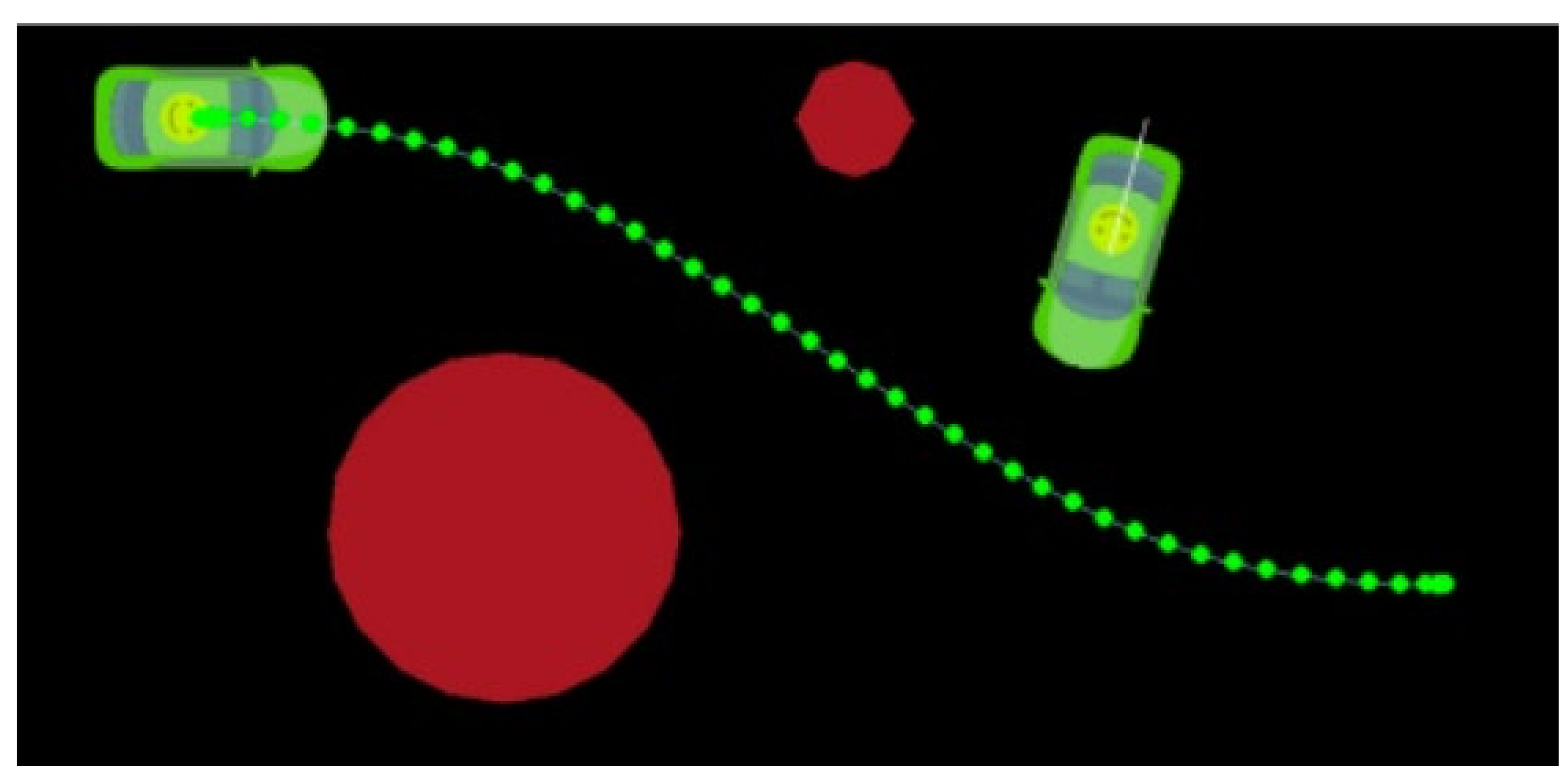
- Decentralized motion planning for unstructured heavy traffic
- Time-parameterized trajectory generation to account for moving obstacles
- Nonholonomic dynamics with state and input bounds
- Accommodate large vehicles like buses and vehicles with trailers.
- Scale to high-DOF systems (cars with multiple trailers)

Future Goals:

- Intent prediction of nearby vehicles
- Resilience to sensor uncertainty
- Multimodal traffic
- Communication through turn signaling, horns, and indicative maneuvers
- Implementation on 1/8 scale ROAR cars
- From micro to macro traffic models

Unstructured Heavy Traffic in Paris, France

Incentivizing good collective behavior by local driving protocols. Motion Planning makes a huge difference to collective behavior



Simulator by Nicholas Eichenberger, Sunay Poole, Ritika Srivastava, and Ashwin Vangipuram

Preliminary Insights on Bicycle Model:

- Polynomial paths on flat variables are poor local plans
  - Fitting polynomials on flat coordinates requires sampling many extra variables
  - Required time cannot be found efficiently via sampling or search
  - Geometric polynomials like eta splines cannot handle dynamic obstacles or input constraints

Next Steps:

- Simulate output of a feedback linearizing controller as a local planner
  - Implicitly account for state and input saturation
  - No additional state variables to sample
  - Controller can follow final path

Education and Outreach:

- Researchers from the project served as project mentors for EECS 106B / 206B, Berkeley's mezzanine level robotics course.

Dynamic Traffic Routing with Iterative Optimization

Kshiitij Kulkarni, Chinmay Maheshwari and Shankar Sastry

NSF CPS Grant FORCES: grant number CNS-1545126

In societal scale CPS local interactions make for complex global behavior. In FORCES we have been interested in exploring how to incentivize good global behavior. Traffic routing between cities has been studied in the static and dynamic settings. There is a need for dynamic traffic assignment (DTA) but at the same time is not as naive as static traffic assignment (STA). We focus on *iterative optimization* as a toolfor dynamic routing.

Challenge problems:

- Beyond STA: network dynamics/residual flow.
- How to overcome limitations of DTA?
  - Computationally tractable algorithm.
  - Adding game theoretic significance to dynamic schemes.
- Convergence guarantees for the relevant dynamics.
- Dynamic incentive mechanism design for improving global societal efficiency of network traffic flows.

Methodology:

Our iterative routing scheme for potential games :

- We update the current flow (denoted x(t)) on any link of the network by keeping track of the incoming and outgoing flows:

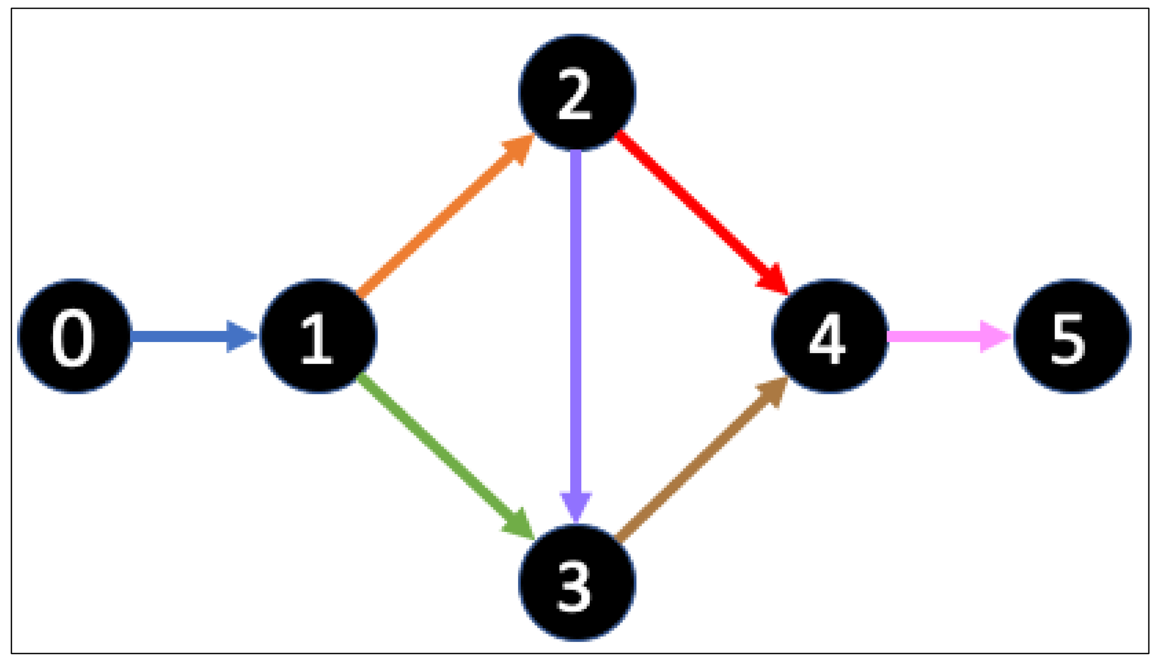
$$\textbf{current flow}(t+1) = \textbf{current flow}(t) + \textbf{inflow}(t) - \textbf{outflow}(t)$$

- The outflows at the edges are combined to determine demands.
- Assuming each node is a player with its corresponding demand, we deploy a STA optimization algorithm to compute how the demand at the nodes needs to be routed:

$$z(t+1) = \arg\min_{z \in \mathcal{Z}(x(t))} \Phi(z, x(t)) \qquad \textbf{inflow}(t) = \Gamma z(t)$$

where $\Phi(z,x) = \sum_{e \in E} \int_{x_e}^{x_e+Z_e} c_e(\tau)d\tau$ **s.t.** $Z = \Lambda z$

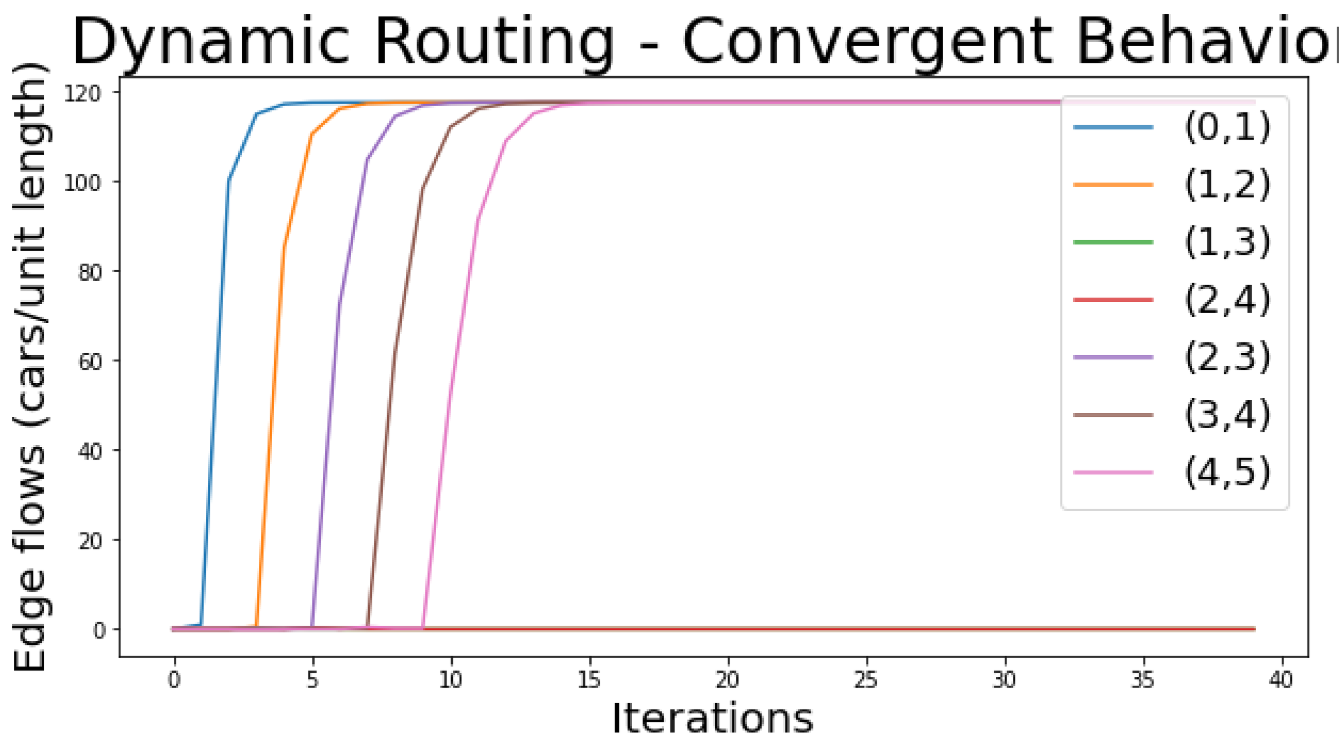
is the potential function for the routing game at every iterate.

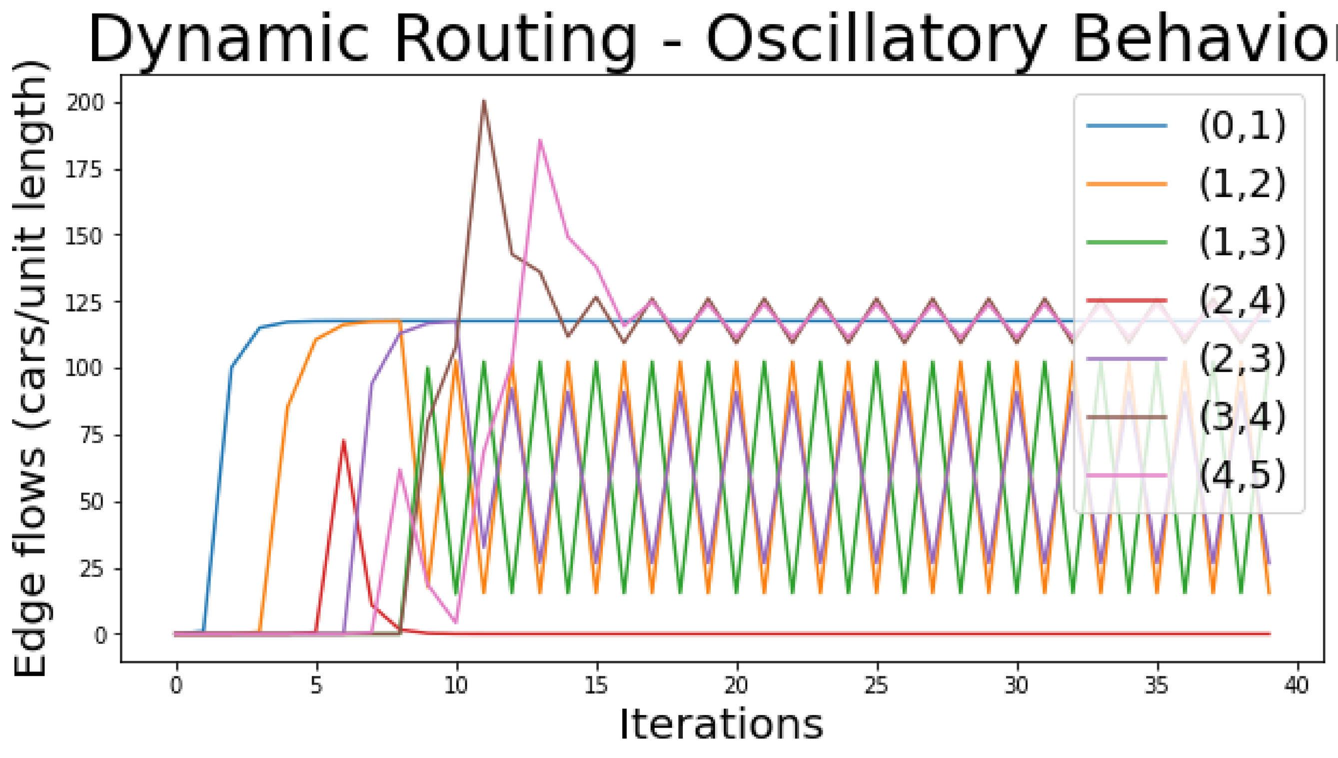


A sample graph with 6 vertices and 7 edges, also known as the Braess network.

Experimental Results:

- High capacity (low costs) on route (0-1-2-3-4-5) leads to convergent flows
- We recover the expected behavior of the Braess paradox in static routing (i.e. all flows are routed through the middle edge (2-3))
- Low capacity (high costs) and high demands lead to oscillations in the route choice.
- Links further down in the network (i.e. (4-5)) are affected later, indicating *causation*.





Ongoing work:

- Convergence proof of iterative optimization algorithm to model dynamic routing as a sequence of static routing games while keeping track of residual flows.
- Computing the efficiency of converged solution
- Incentivizing societal efficiency through local routing protocols