ALEXANDER UBALDO GUTIÉRREZ
PSU ID: 979329711
aug5020@psu.edu

MS in Civil and Environmental Engineering
Transport Engineering
Fall 2025

The Pennsylvania State University
Department of Civil and Environmental Engineering
Xianbiao Hu
Fall 2025

TABLE OF CONTENTS

The Pennsylvania State University
Department of Civil and Environmental Engineering
Xianbiao Hu
Fall 2025

CE 521: TRANSPORTATION NETWORK AND SYSTEM ANALYSIS

HOMEWORK 2

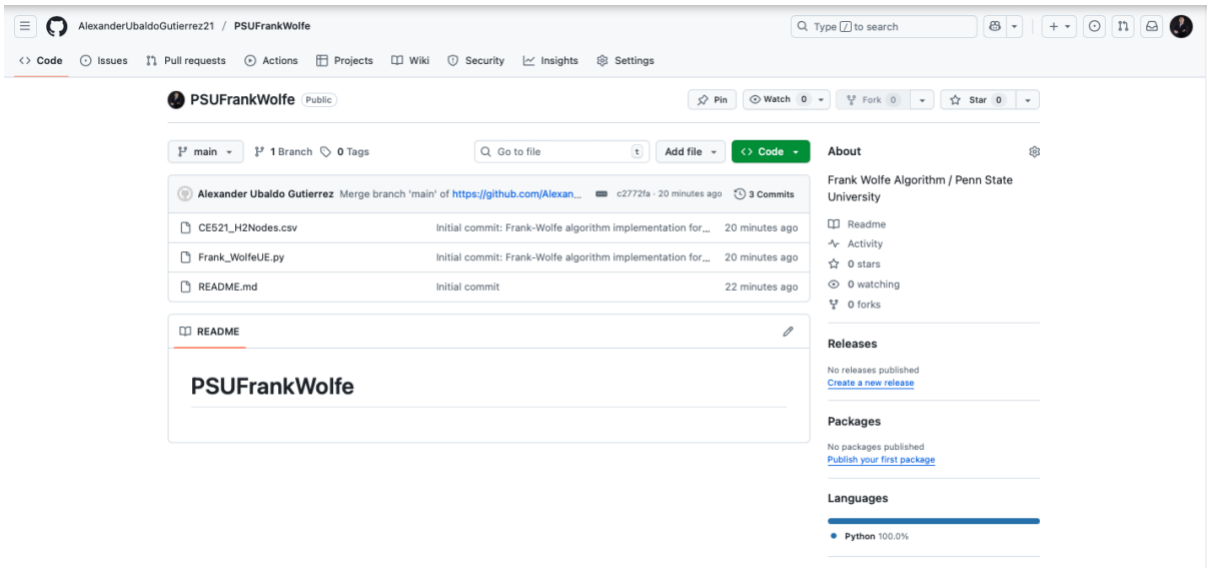
FRANK WOLFE ALGORITHM

PHASE 1

CODING PROCESS

For this network problem, a software solution was developed employing Python as the main programming language, and the libraries NumPy for numerical calculations and NetworkX, which specializes in the data management and analysis of networks structured by nodes. Also, the image network was transformed into a small database, CE521_H2Nodes.csv, to provide quantitative context. The database is divided into two columns, one for network nodes and the other for links.

Frank_WolfeUE.py, our software solution, defines the known variables (Node-1 – Node 3: 8000 or $t(x)$), simple paths for each OD (Origin-Destination) pair, and the all-or-nothing traffic assignment as main steps. After these definitions, the system is ready to generate quantitative data, for instance, iteration values, compute SPTT and TSTT, calculate the gap, and update flows based on vehicles shift. To provide transparency and accountability, a GitHub Repository (image) was created with public access through the link below.



GitHub Repository: PSUFrankWolfe

The Pennsylvania State University
Department of Civil and Environmental Engineering
Xianbiao Hu
Fall 2025

PHASE 2

RUNNING ON TERMINAL
After running our code in the MAC Terminal, the algorithm stopped at the first iteration because the initial all-or-nothing assignment already represents the User Equilibrium (UE) principle.

For this specific network:

- OD 1-3: Direct path (Link 1) has free-flow time 15 min, alternative path (Links 2-3-4) has total free-flow time 45 min (15+15+15). So all 8000 vehicles take the direct path.
- OD. 2-4: Direct path (Link 7) has free-flow time 15 min, alternative path (Links 5-3-6) has total free-flow time 45 min (15+15+15). So all 6000 vehicles take the direct path.

```
Network Parsed:
Nodes: [1, 3, 5, 6, 2, 4]
Links: {1: (1, 3), 2: (1, 5), 3: (5, 6), 4: (6, 3), 5: (2, 5), 6: (6, 4), 7: (2, 4)}
OD Pairs: {(1, 3): 8000, (2, 4): 6000}
Paths: {(1, 3): [[1, 3], [1, 5, 6, 3]], (2, 4): [[2, 5, 6, 4], [2, 4]]}
Converged At Iteration 1, Max Rel Change: 0.0, Relative GAP: 3.9682539682539684

Final Link Flows:
Link 1: 8000.00 Vehicles
Link 2: 0.00 Vehicles
Link 3: 0.00 Vehicles
Link 4: 0.00 Vehicles
Link 6: 0.00 Vehicles
Link 5: 0.00 Vehicles
Link 7: 6000.00 Vehicles

Travel Times:
Link 1: 81.67 Minutes
Link 2: 15.00 Minutes
Link 3: 15.00 Minutes
Link 4: 15.00 Minutes
Link 6: 15.00 Minutes
Link 5: 15.00 Minutes
Link 7: 65.00 Minutes

Shortest Path Travel Time (SPTT): 210000.00 Vehicle-Minutes
Total System Travel Time (TSTT): 1043333.33 Vehicle-Minutes
Relative Gap: 3.968254
```

Since there are no competitive alternative paths (all 3x times longer), the flows don't change, and the relative gap remains high due to system congestion. The gap being non-zero indicates the inefficiency due to traffic density. As shown in the image above, once we run the command **python3 Frank_WolfeUE.py**, we identify the following outputs:

- SPTT: 210000.00 Vehicles-Minutes
- TSTT: 1043333.33 Vehicles-Minutes
- Relative GAP: 3.968254

Considering that γ (gamma)/gap and λ (lambda) are convergence thresholds that measure how close the current solution is to the theoretical optimum, both values influence the number of iterations. Despite that relative gap was 3.968254, which is larger than $\gamma = 0.001$, convergence was triggered by $\lambda = 0.001$ (no change in flows). Taking into consideration that

our code checks both conditions: “if max_rel_change < lambda_thresh or relative_gap < gamma” (line 112), the algorithm stopped because λ it was satisfied; there's no flow redistribution needed.