10 weeks

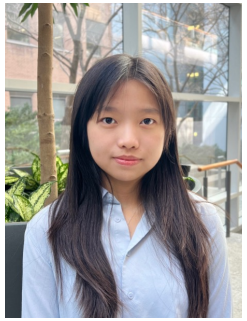
to search for a home [1]

50%

of homebuyers have at least one regret [2]

Group 55 – ECE297

Neighbourhood Explorer



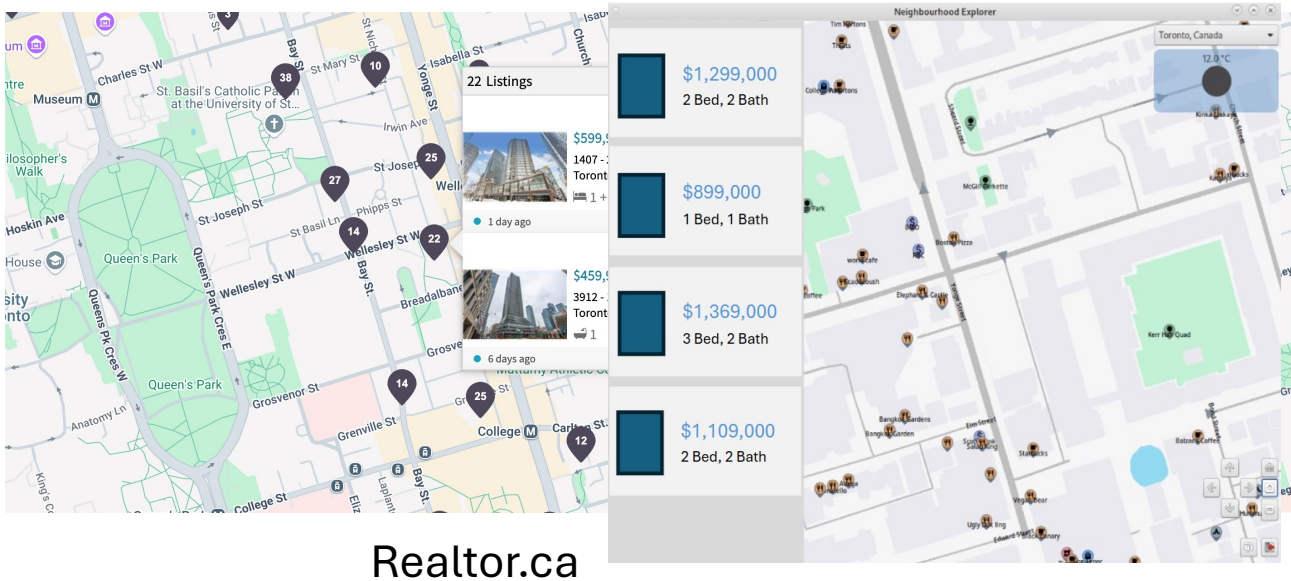
Xiao Nan (Hannah) Ye

Raffael Gil Dela Cruz

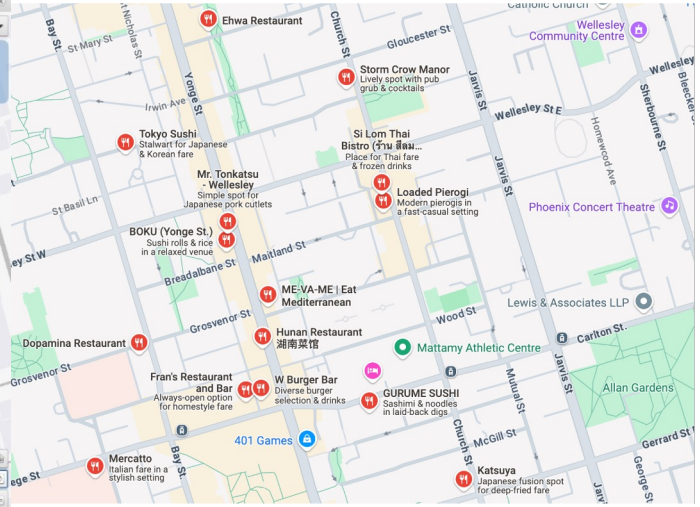
Herman Ho

Neighborhood Explorer’s current features **lay the foundation** for its future as a powerful tool for both **area exploration** and **real estate searching**

Real-Estate Search | **Neighbourhood Explorer** | **ore Features, Services, Amenities in an Area**



Realtor.ca



Google Maps

Table Of Contents

Visual Design

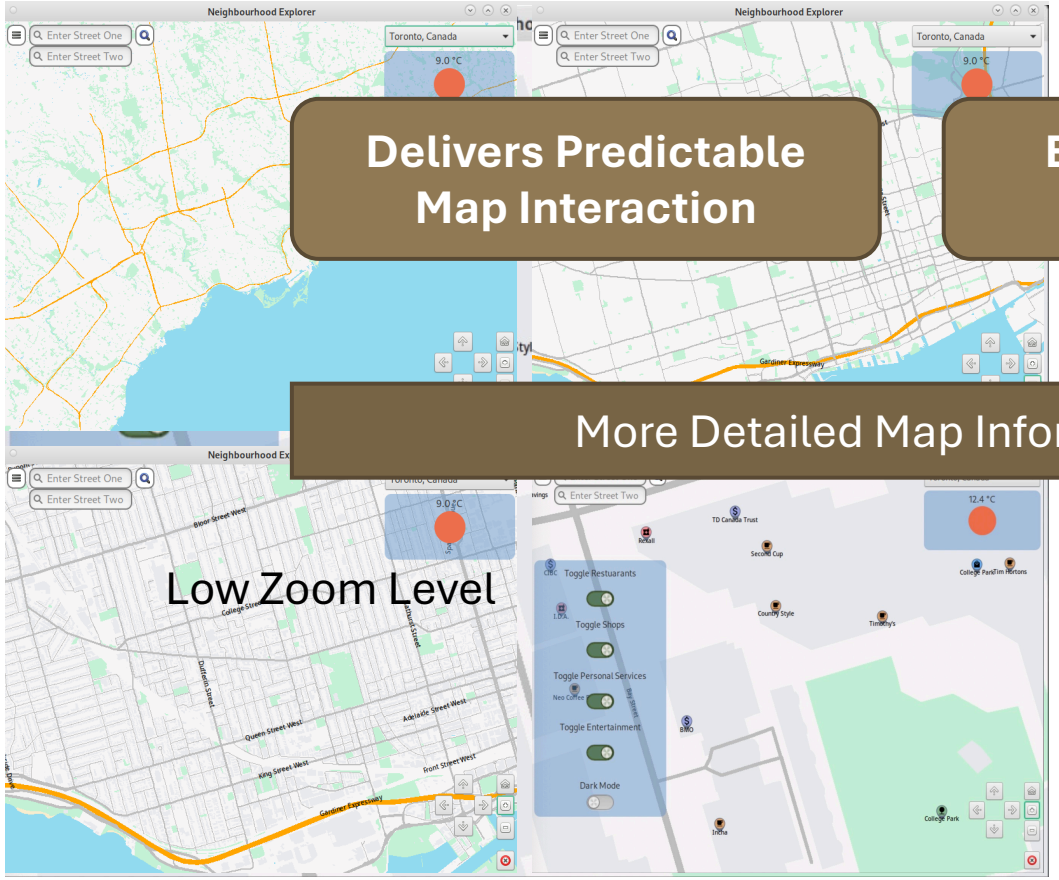
Path Searching

Algorithms

Future Vision

1. Visual Design

Progressive display of information **reduces clutter** and helps users **focus on relevant details** [3]



Low Zoom Level

Delivers Predictable Map Interaction

Enhances System Usability

More Detailed Map Information

Major Roads

Detailed Map Features

High Zoom Level

POIs

Quadrant- and category- based POI groupings offer a **clear overview** and **minimizes screen clutter**

Restaurants

Shops

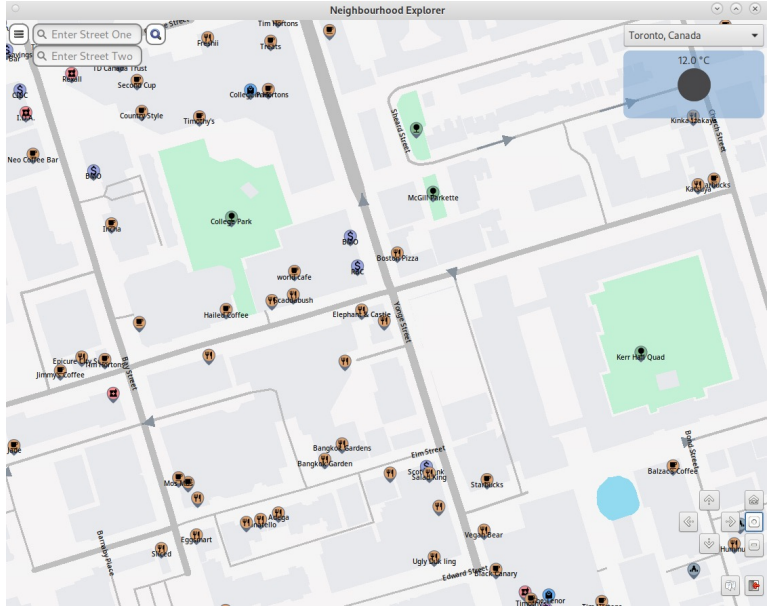
Personal Services

Entertainment

Quadrant- and category- based POI groupings offer a **clear overview** and **minimizes screen clutter**

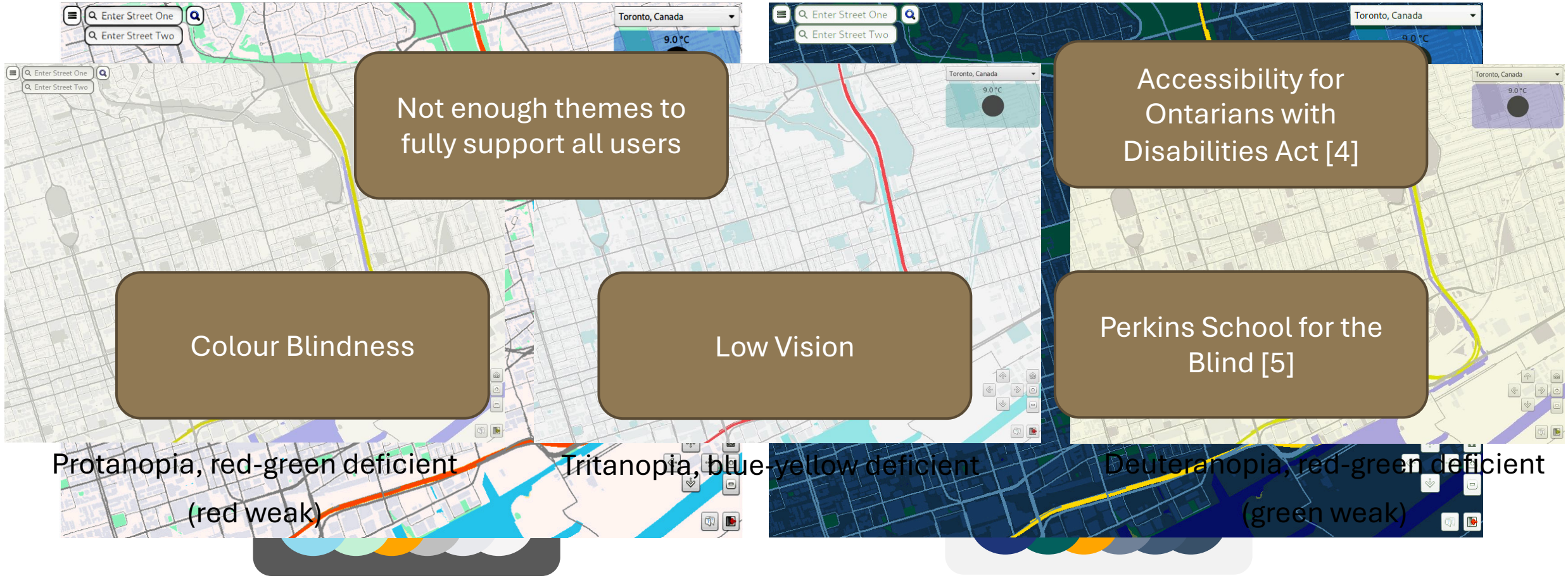
Lower Zoom Level

Higher Zoom Level

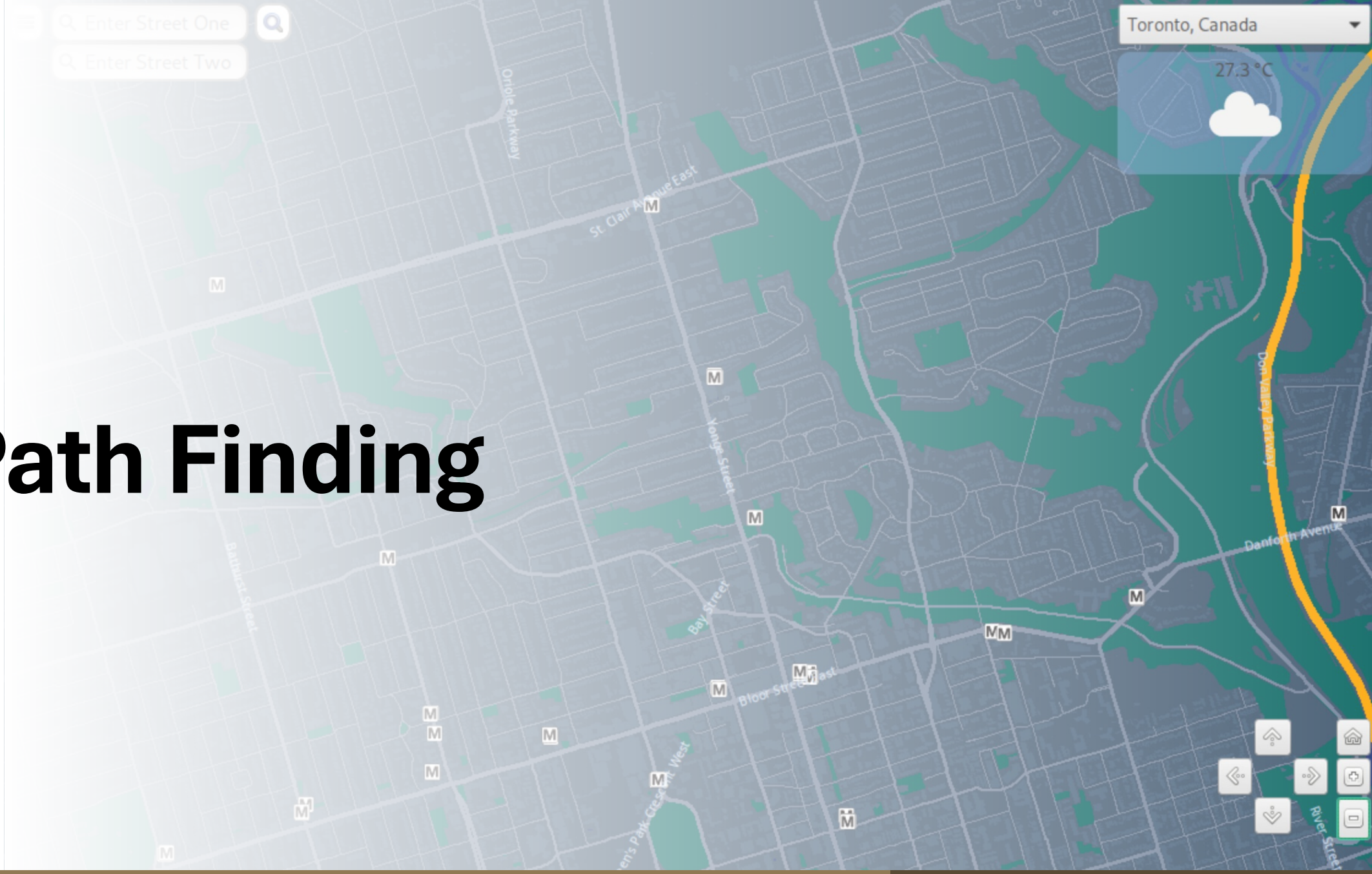


Highest Zoom Level

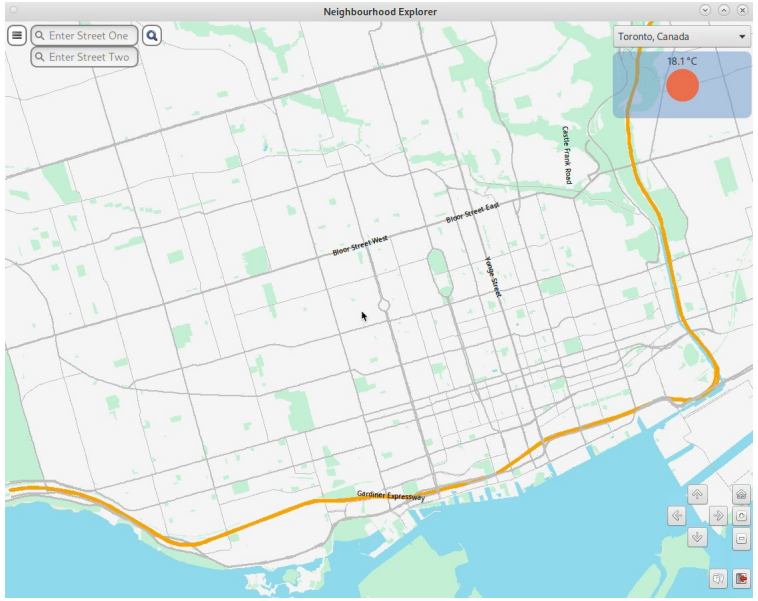
Different colour schemes are used to **improve accessibility** for a **diverse range of users**



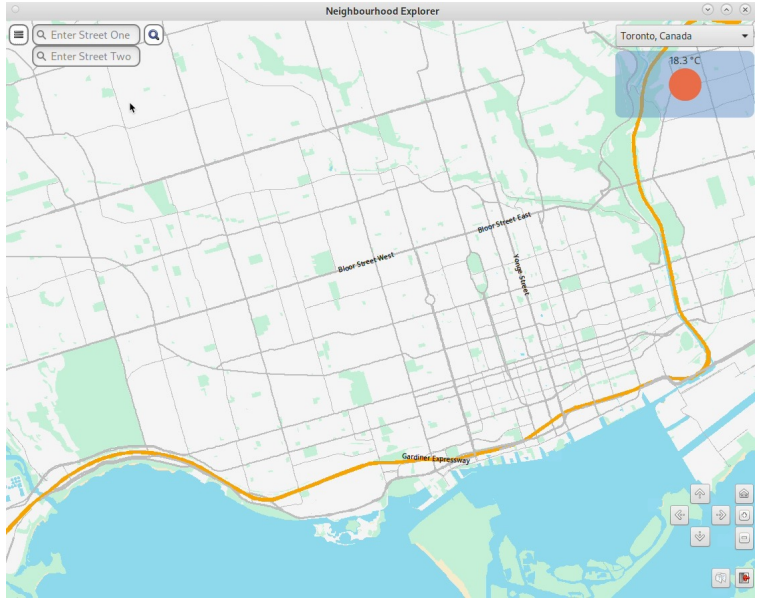
2. Path Finding



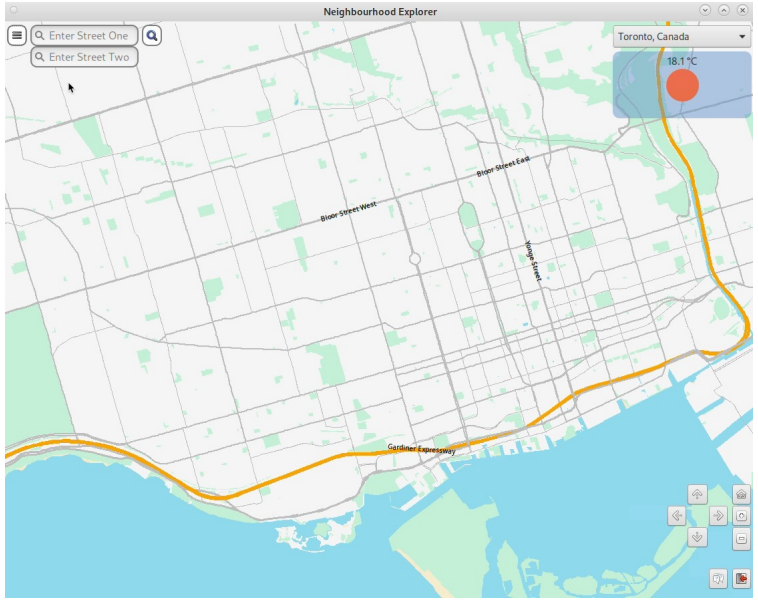
Three methods to perform pathfinding **enhances flexibility** and **prioritizes user preference**



Right Click to Select Origin and Destination



Pathfinding Mode: Click on Origin and Destination



Pathfinding Mode: Enter 4 Street Names

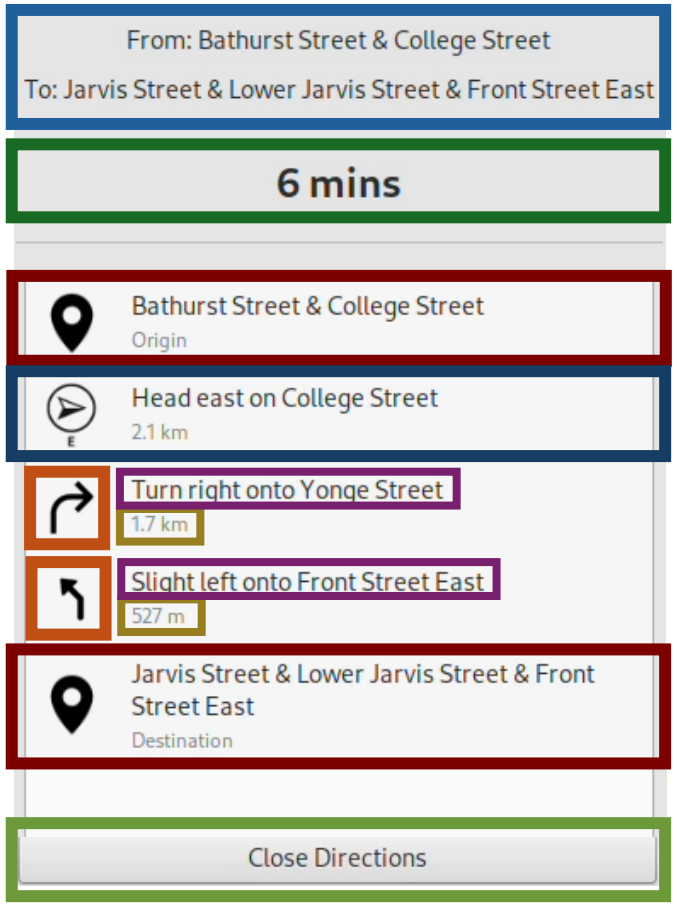
Autocomplete **speeds up** the street name entry process and **offers real-time suggestions** [6]

Reduces cognitive load

Automatically displays suggestions

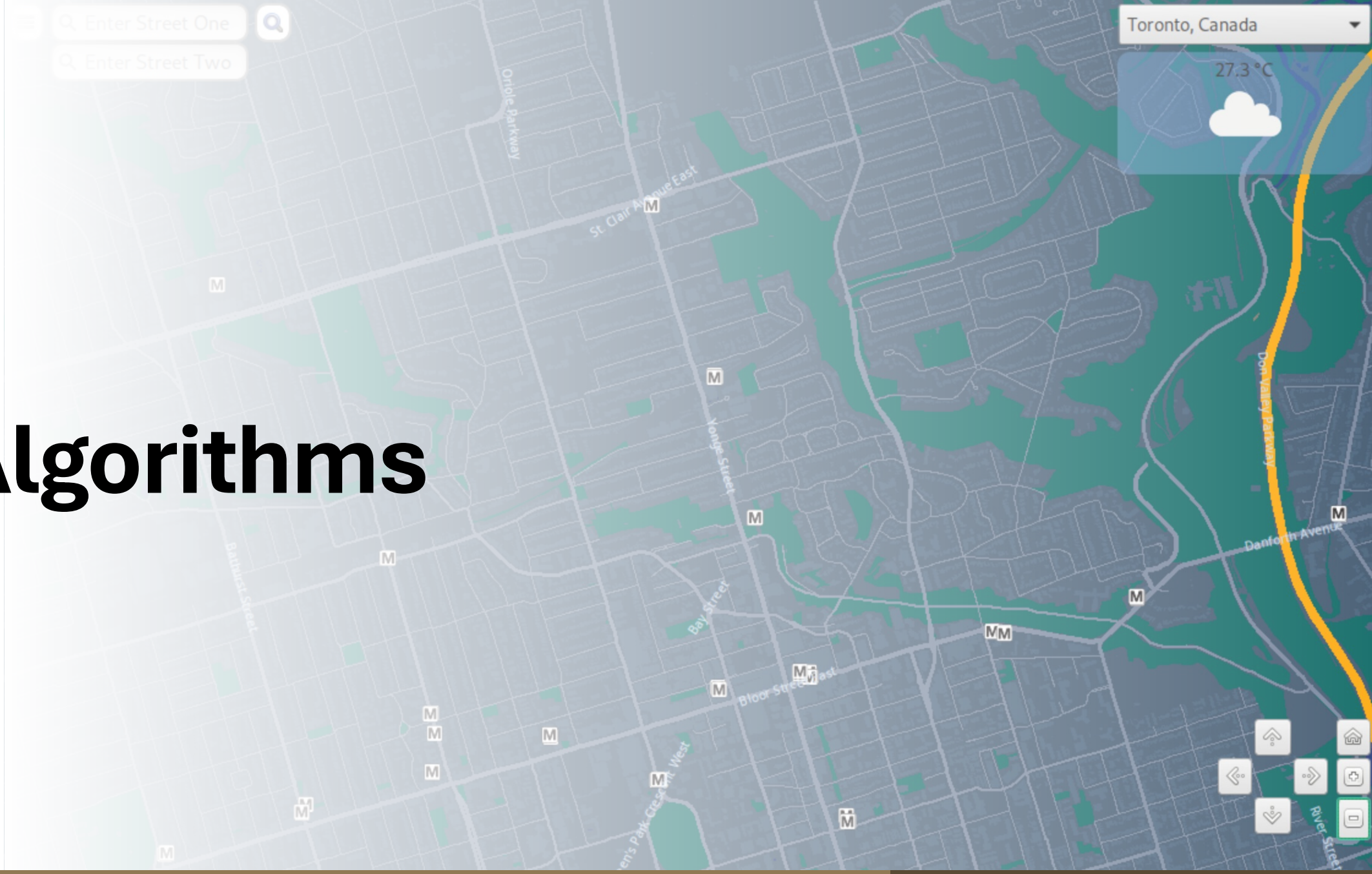
Faster than a **blink of an eye** - delivers results in 0.0005s

The directions panel presents **key pathfinding details** and maintains a **concise and minimalistic design**



- Route Overview
- Travel Time
- Indication of Origin & Destination
- Initial Travel Direction
- Street & Turn Instructions (Text)
- Turn Direction Icon
- Travel Distance [m / km]
- Close Directions Button

3. Algorithms

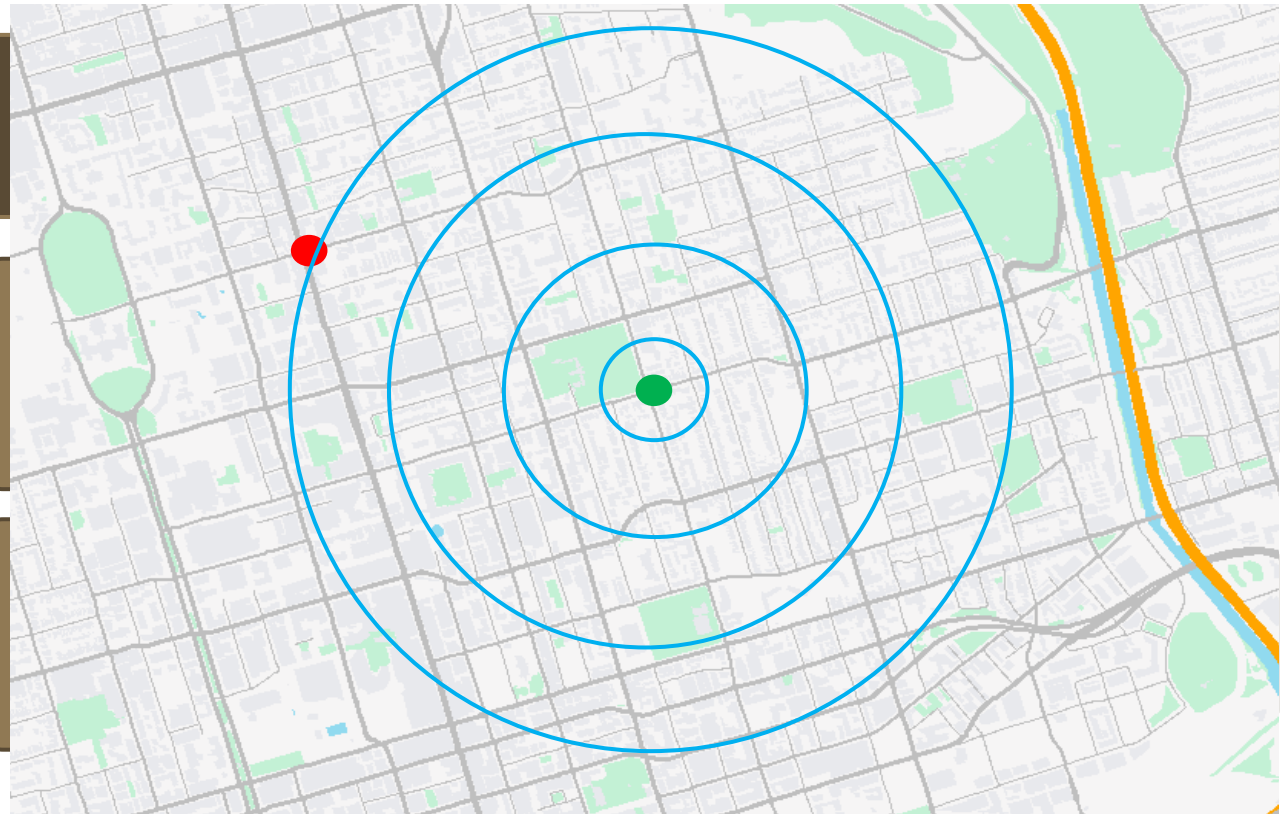


The evolution of our pathfinding algorithms **drives faster and more efficient** route calculations

Breadth-First Search

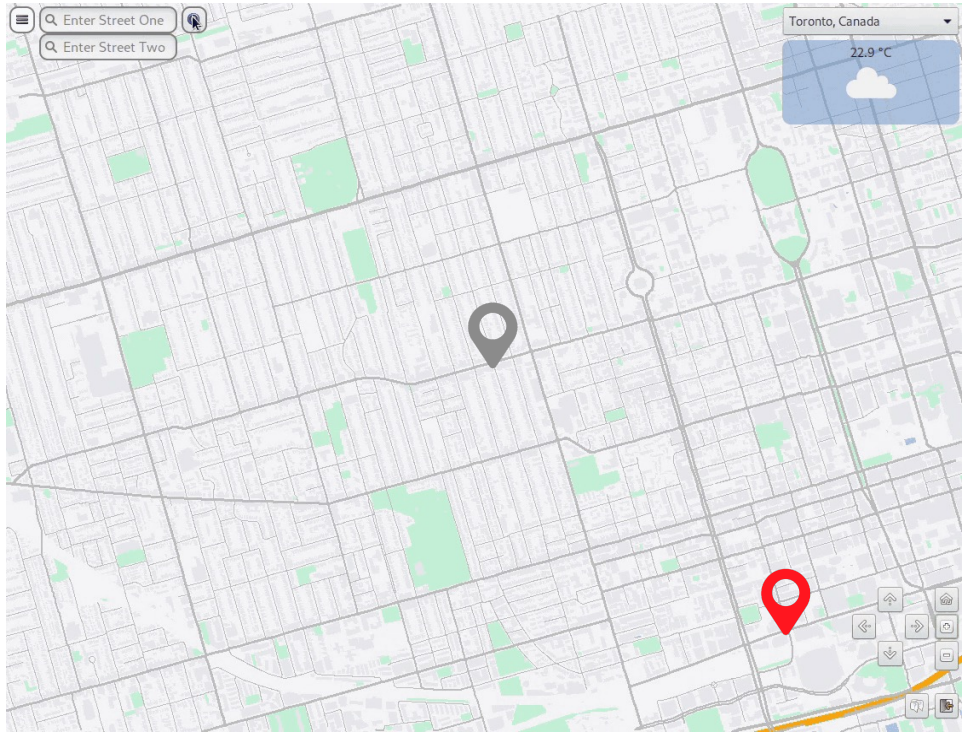
Explores nearby intersections gradually

Does not factor in travel time



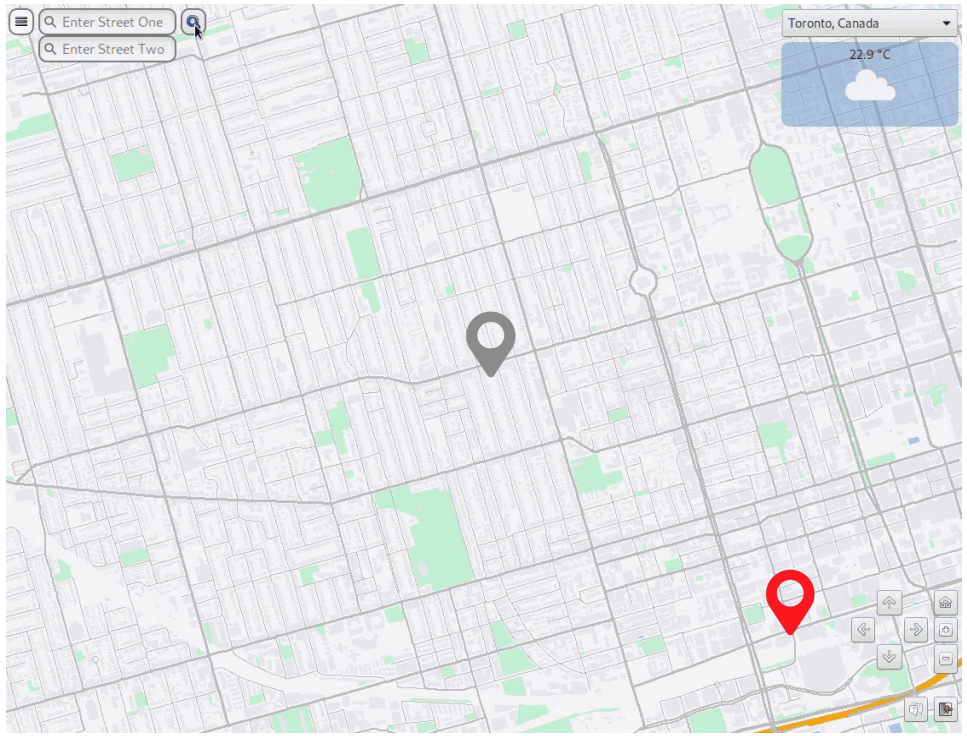
The evolution of our pathfinding algorithms **drives faster and more efficient** route calculations

Dijkstra's Algorithm



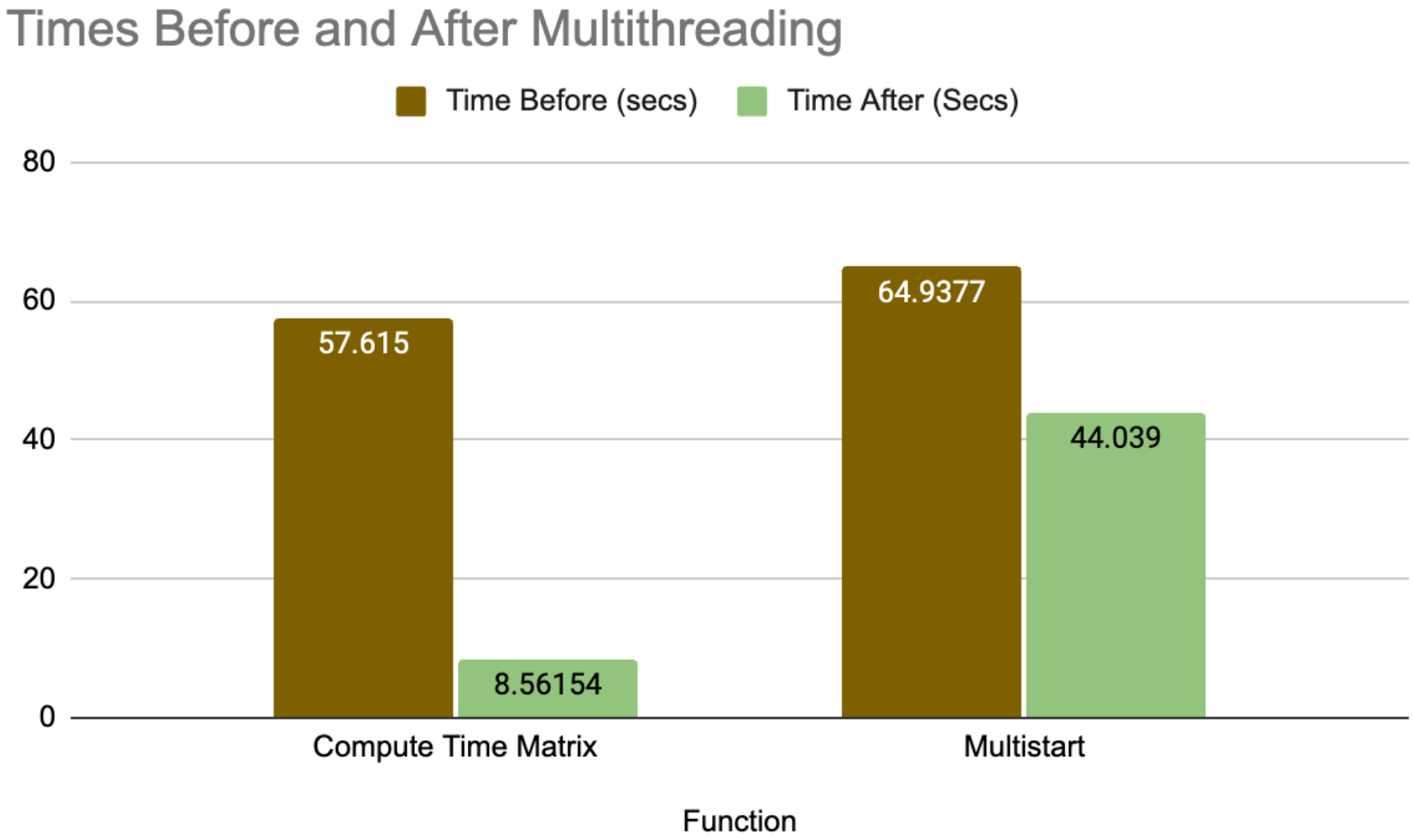
~0.0056s on average

A* Algorithm

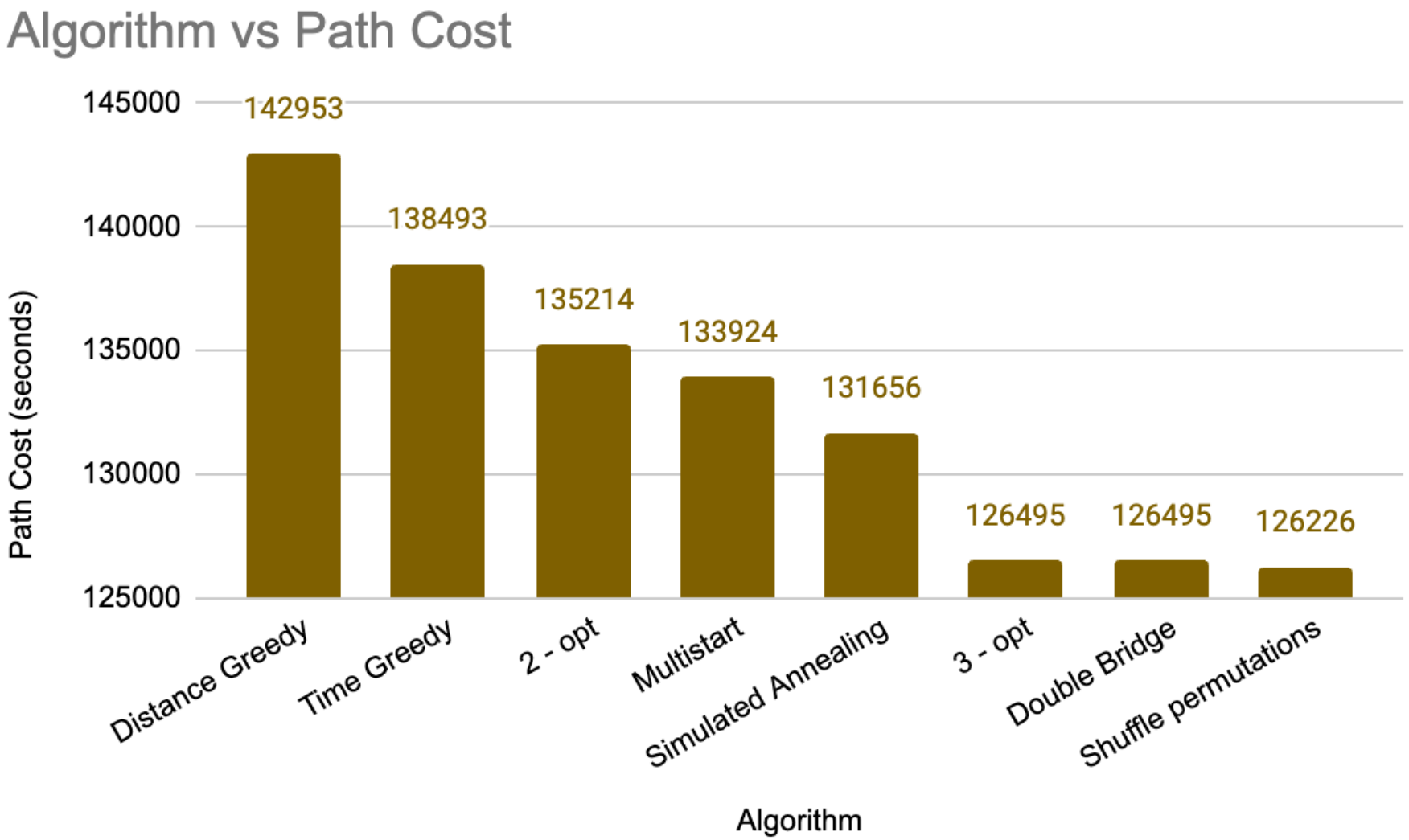


~0.0047s on average

Multithreading minimizes computation times to maximize optimization time

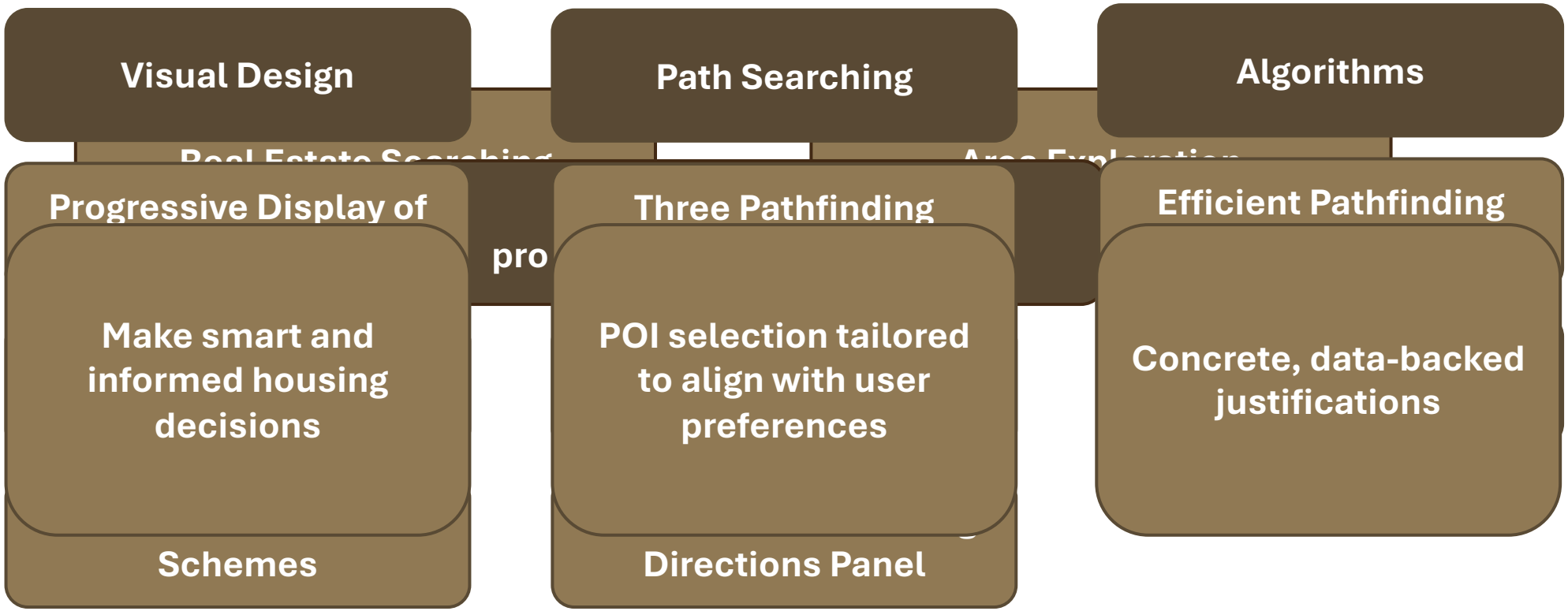


Custom Optimization Functions improved QoR by a total of 46%



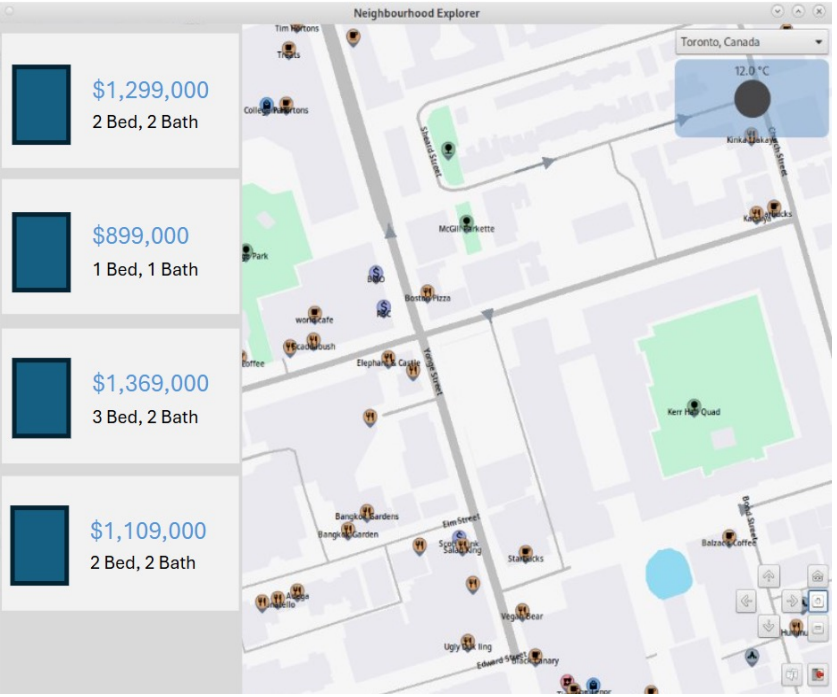
4. Future Vision

Current features of Neighbourhood Explorer enable further development as a **real estate searching** and **area exploration** tool



Transforming how you explore with **integrated features** in **one seamless platform**

Housing Data

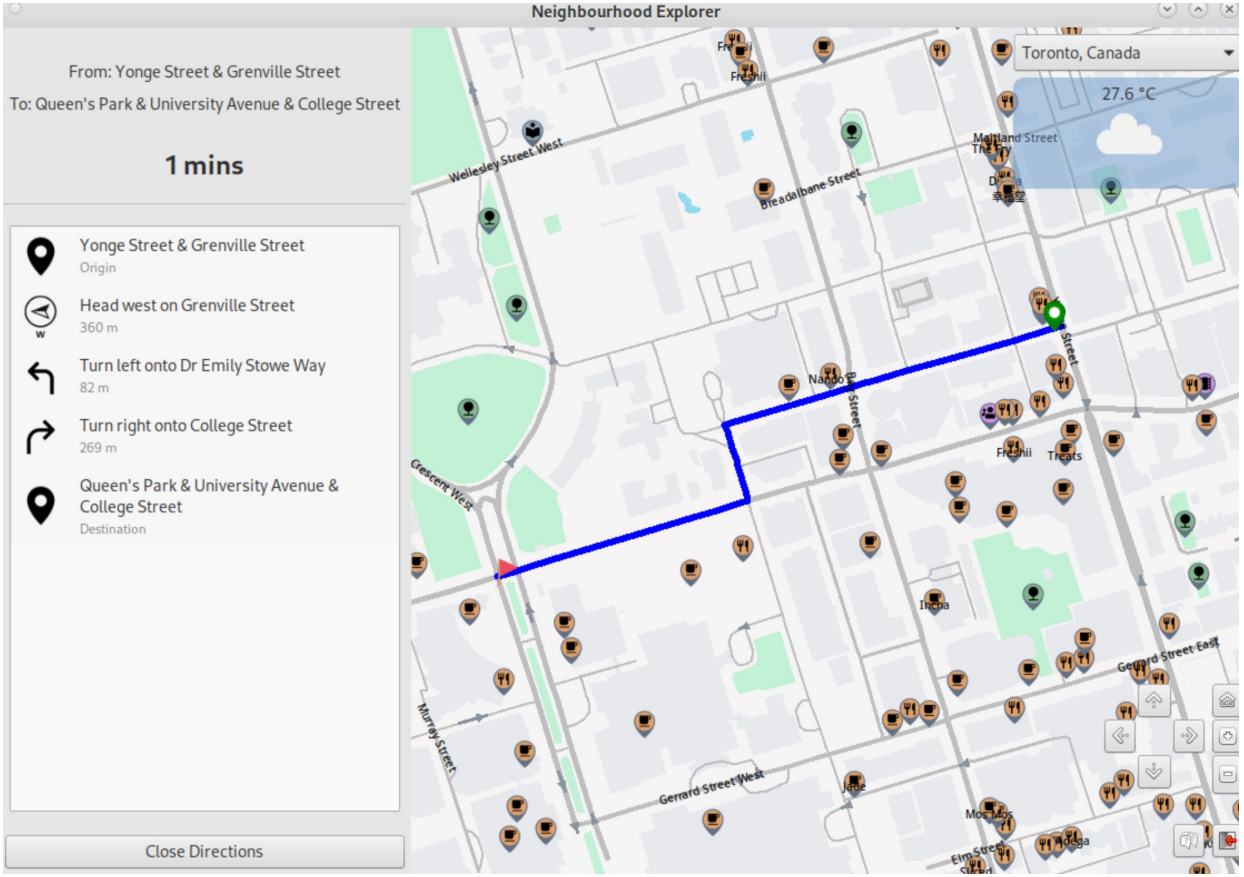


User Personalization

Input personal preferences

User preference-driven searching

Neighbourhood Explorer = Google Maps + Real Estate Agent



References

[1] National Assocation of Realtors, "Profile of Home Buyers and Sellers," National Association of Realtors, Washington, DC, United States of America, 2024. [Online]. Available: https://www.nar.realtor/sites/default/files/2024-11/2024-profile-of-home-buyers-and-sellers-highlights-11-04-2024_2.pdf. [Accessed: Apr. 29, 2025].

[2] Bankrate, "Nearly Half of Current Homeowners Have at Least One Regret Related to Their Purchase, New York, NY, United States of America, May 22, 2024. [Online]. Available: https://www.bankrate.com/f/102997/x/c84a6b9359/homeowner-regrets-survey-press-release.pdf. [Accessed: Apr. 29, 2025].

[3] N. Kong, T. Zhang, and I. Stonebraker, "Evaluation of Web GIS Functionality in Academic Libraries," Libraries Faculty and Staff Scholarship and Research, Jun. 2015. [Online]. Available: https://docs.lib.purdue.edu/cgi/viewcontent.cgi?article=1109&context=lib_fsdocs. [Accessed: Feb. 23, 2025].

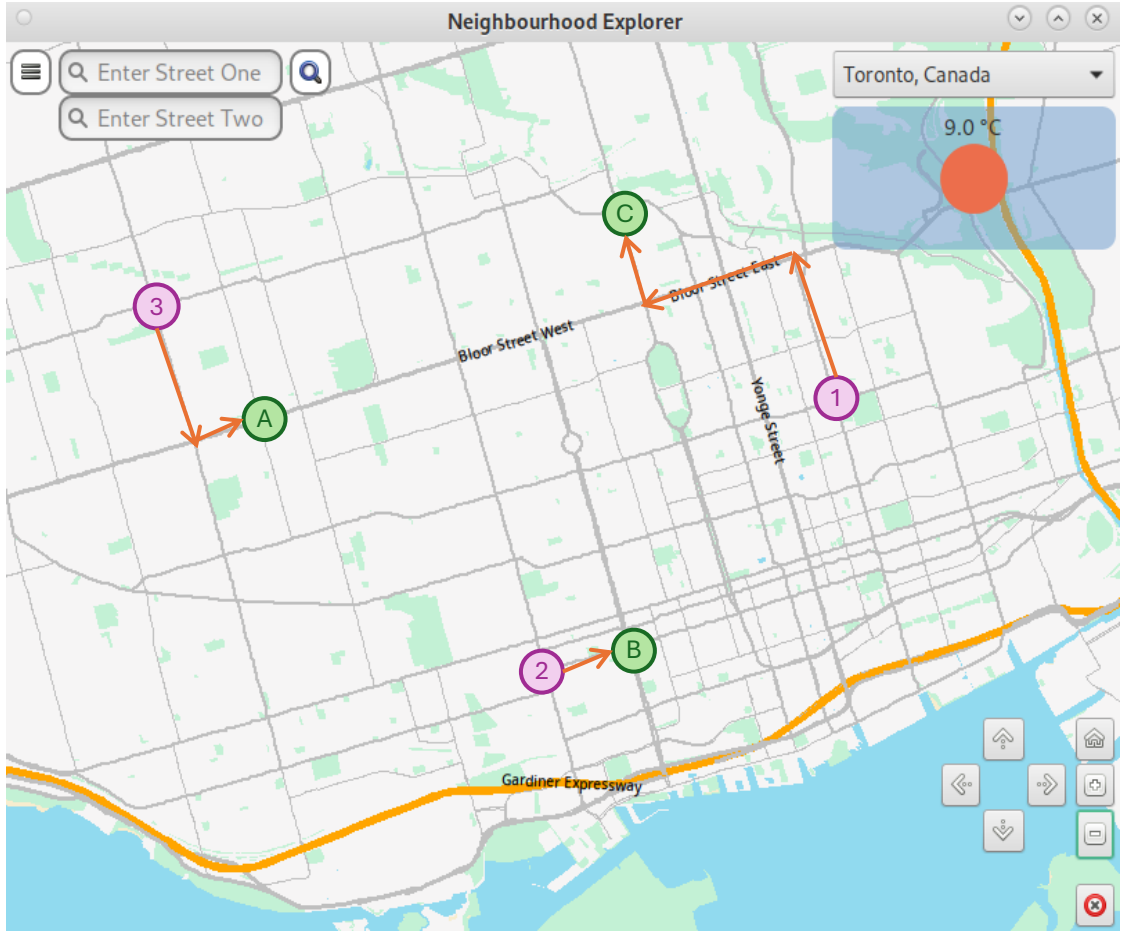
References

[4] Accessibility for Ontarians with Disabilities Act, O Reg. 191/11.

[5] Perkins School for the Blind, "High contrast and low vision," [Online]. Available: https://www.perkins.org/resource/choosing-high-contrast-color-schemes-for-low-vision/. [Accessed: Apr. 28, 2025].

[6] D. Ward, J. Hahn, and K. Feist, "Autocomplete as Research Tool: A Study on Providing Search Suggestions," ITAL, vol. 31, no. 4, pp. 6–19, 2012, doi: 10.6017/ital.v31i4.1930. [Online]. Available: https://www.researchgate.net/publication/284583661_Autocomplete_as_Research_Tool_A_Study_on_Providing_Search_Suggestions. [Accessed: Feb. 19, 2025].

Appendix A: Multi-start



Depot

Delivery Pick Up

- Develop X number of paths starting from the depot with the shortest initial path travel time and run the set of algorithms on each path
- **Purpose:** Starting from multiple locations potentially avoids being stuck in a local minima

Paths from Depot to Closest Pick-Up Point

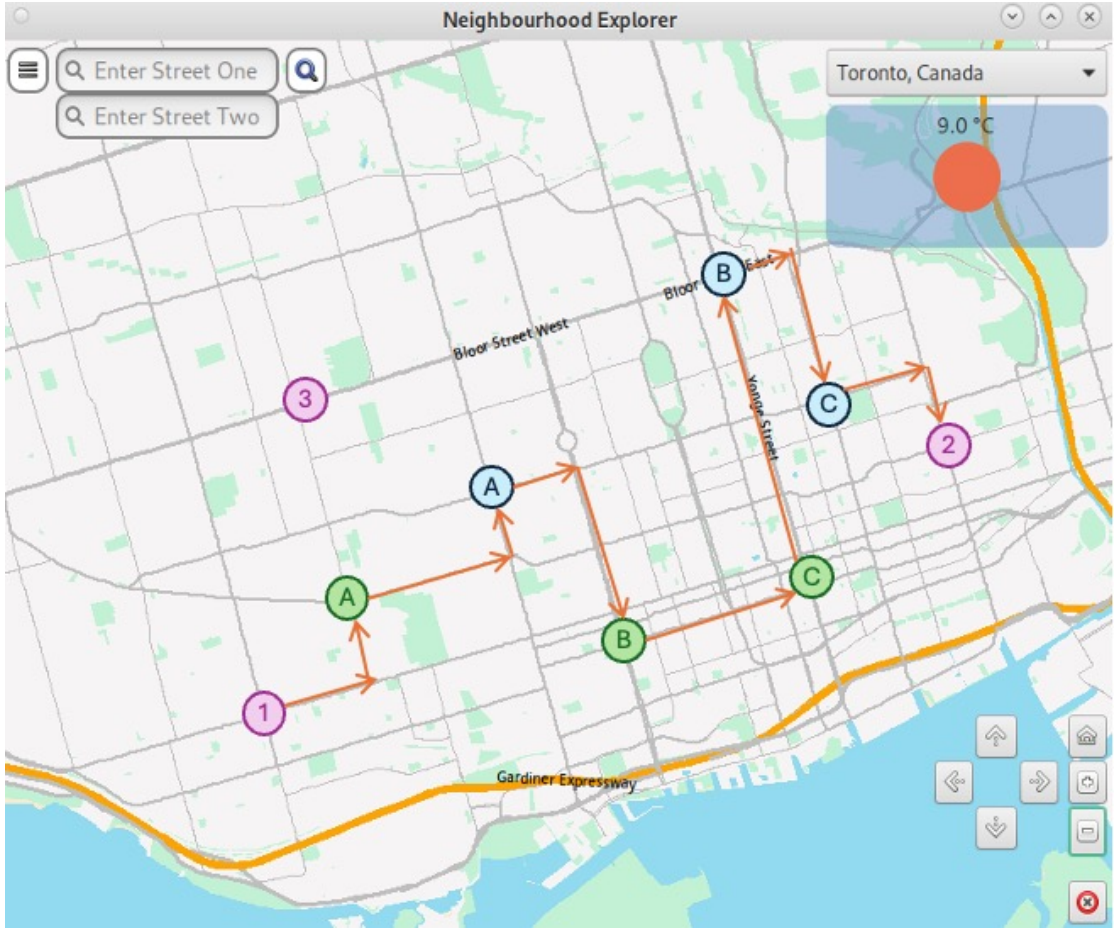
1 -> C | 2 -> B | 3 -> A

Select X Paths with **Shortest** Travel Time

2 -> B | 3 -> A | 1 -> C

Appendix B: Greedy Algorithm

Depot (D)
Delivery Pick Up (PU)
Delivery Drop Off (DO)



Inter-section ID	D1	D2	D3	PU1	PU2	PU3	DO1	DO2	DO3
D1									
D2									
D3									
PU1									
PU2									
PU3									
DO1									
DO2									
DO3									

Time matrix stores path and travel time from intersection in respective **row TO column**

Appendix B: Greedy Algorithm

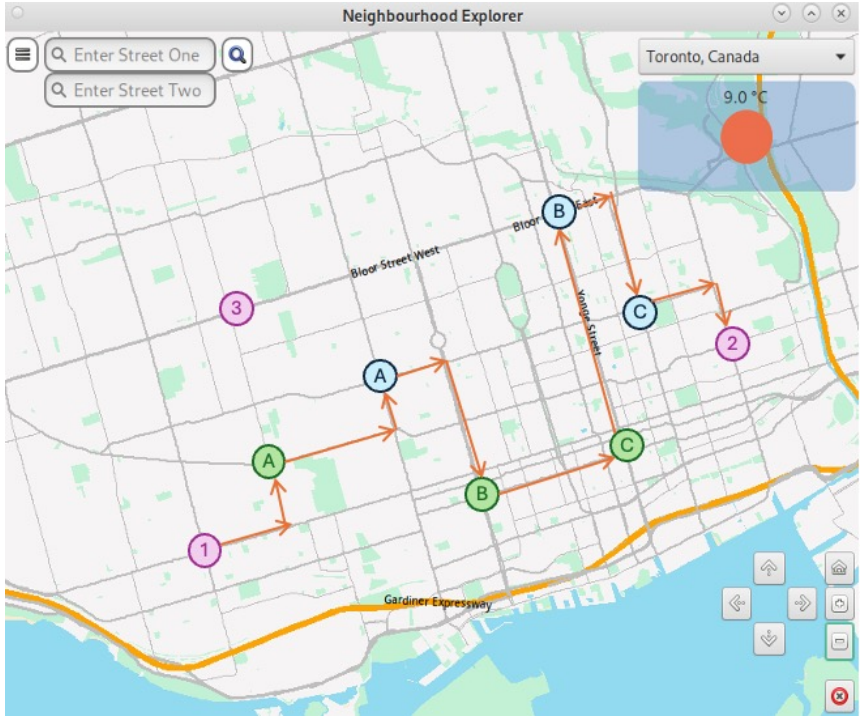
Depot (D)

Delivery Pick Up (PU)

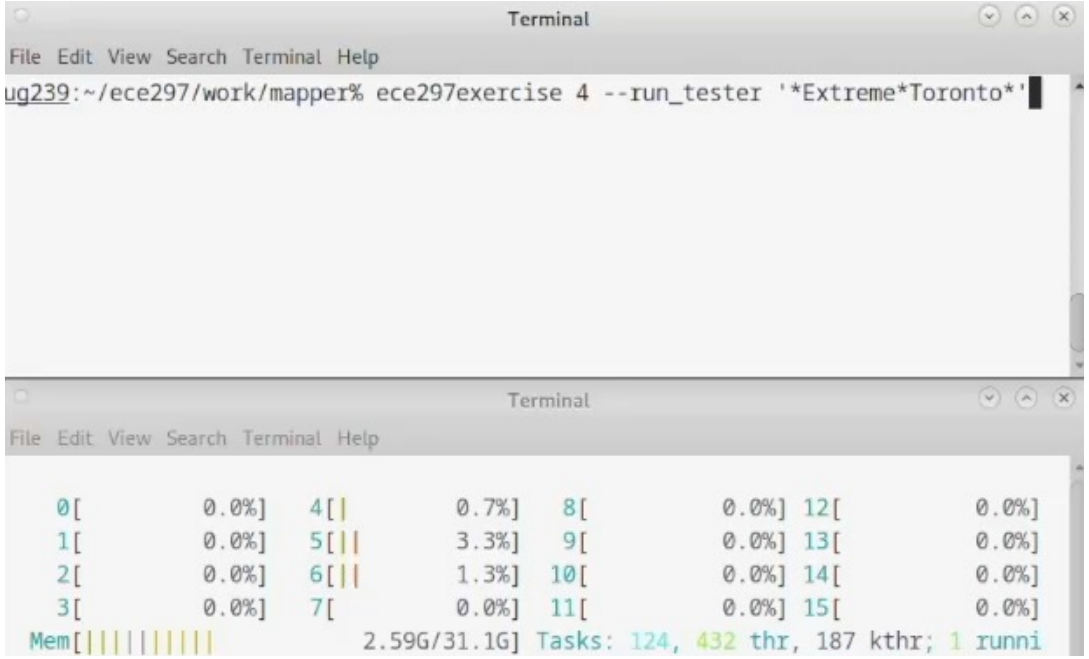
Delivery Drop Off (DO)

1. **Vector of depot objects**: sorted in order of increasing travel time from **depot to first pickup location**
2. Insert first sub-path (depot to first pickup) into travel route
3. While there are still **undelivered packages**
   - Loop through all deliveries: Find **minimum travel time** to **next delivery's pick-up/drop-off location**
   - Ensure all deliveries to be dropped off at that location is picked up (drop off all deliveries at the same time)
4. Upon determining the next location to go to: **add a new sub-path** (current location ID, next location ID, path)
   - Loop through all deliveries, update pick-up and drop-off information for all deliveries at that location
   - Pick-up/Drop-off status managed in a vector of bool pairs
5. After all deliveries: **find nearest depot to last drop-off point,** insert sub-path (last drop-off to depot)

All deliveries at a single location can be **picked up and dropped off simultaneously**



Appendix C: Parallel Processing



Video displays all 16 threads begin/end working simultaneously when parallel implementation of multi-start for all depot begins and ends

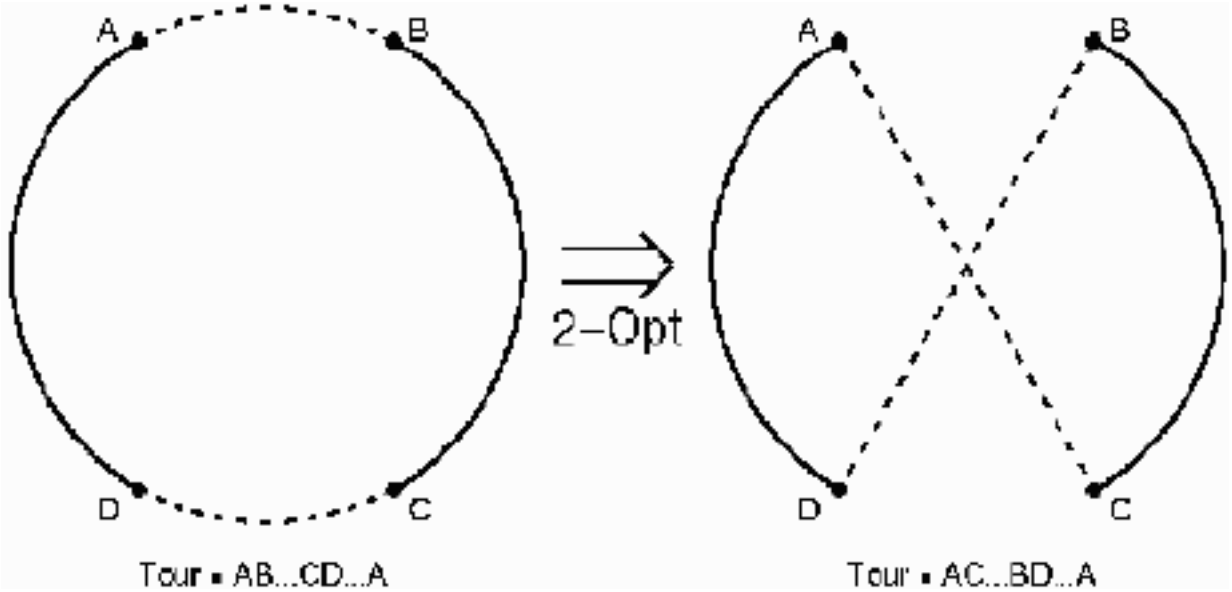
(Visualized using **htop** command)

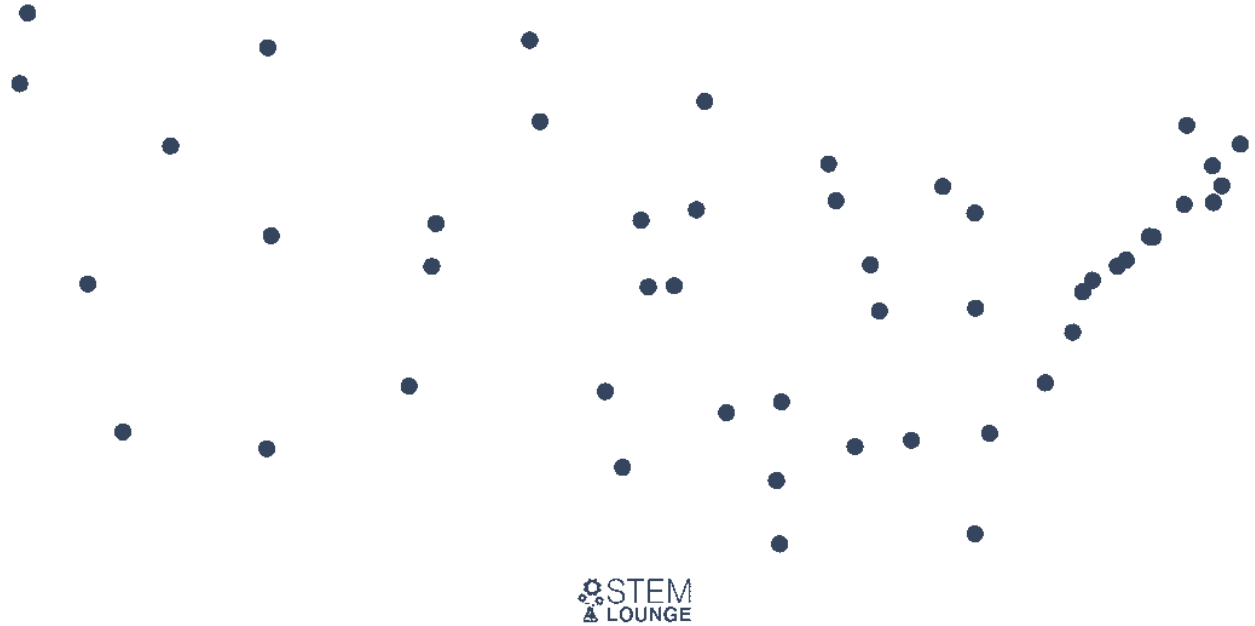
- Dividing tasks into smaller parts and executing them across multiple cores and processors simultaneously to improve performance and efficiency

Tasks performed in parallel in Neighbourhood Explorer:

- Time matrix: use multi-dest Djikstra to compute shortest travel time and path from one source to multiple destinations
- Path finding (Multi-start for many depots)
  - Greedy Algorithm
  - Randomized Two-opt
  - Two-opt
  - Three-opt
  - Simulated Annealing

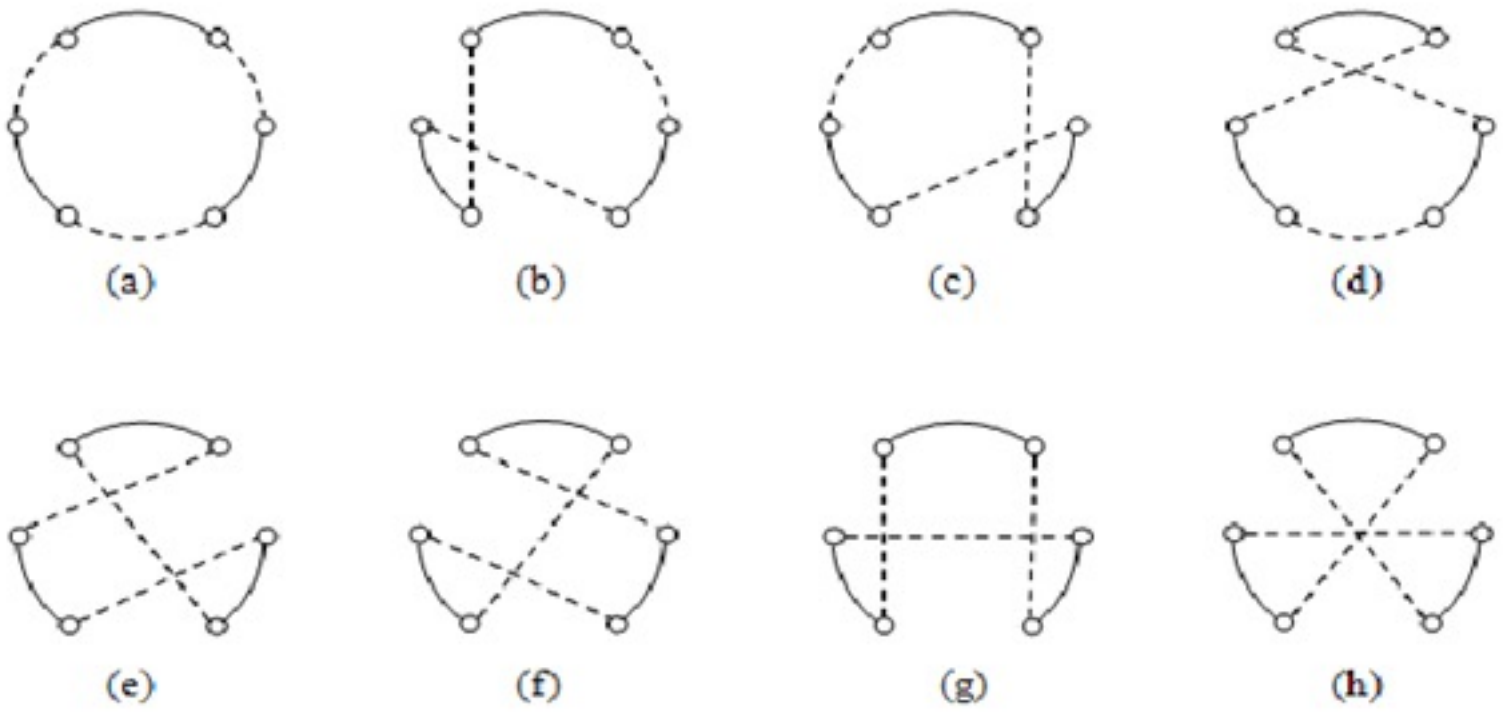
Appendix D: 2- Opt





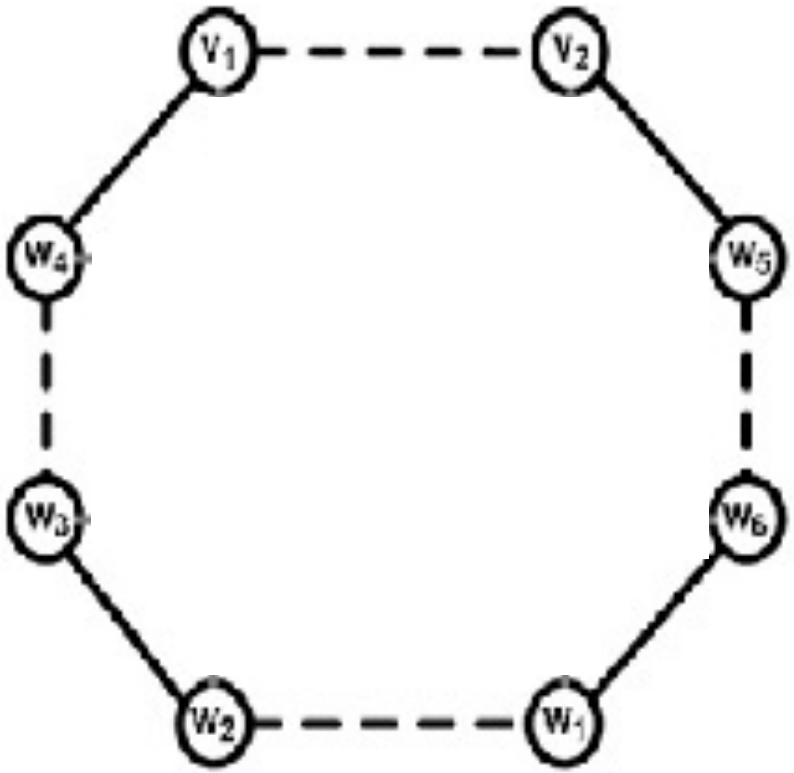
Untangling a messy path by swap two connections!

Appendix E: 3 - Opt

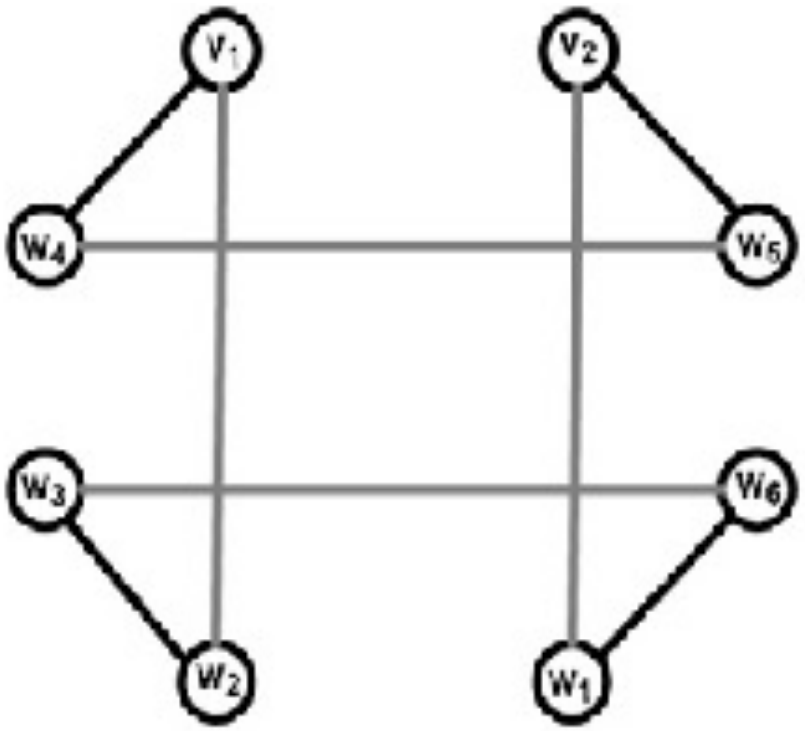


A→B→C→D→E→F (Original)
A→E→D→C→B→F (Case 1)
A→B→C→E→D→F (Case 2)
A→C→B→D→E→F (Case 3)
A→C→B→E→D→F (Case 4)
A→E→D→B→C→F (Case 5)
A→D→E→C→B→F (Case 6)
A→D→E→B→C→F (Case 7)

Appendix F: Double Bridge

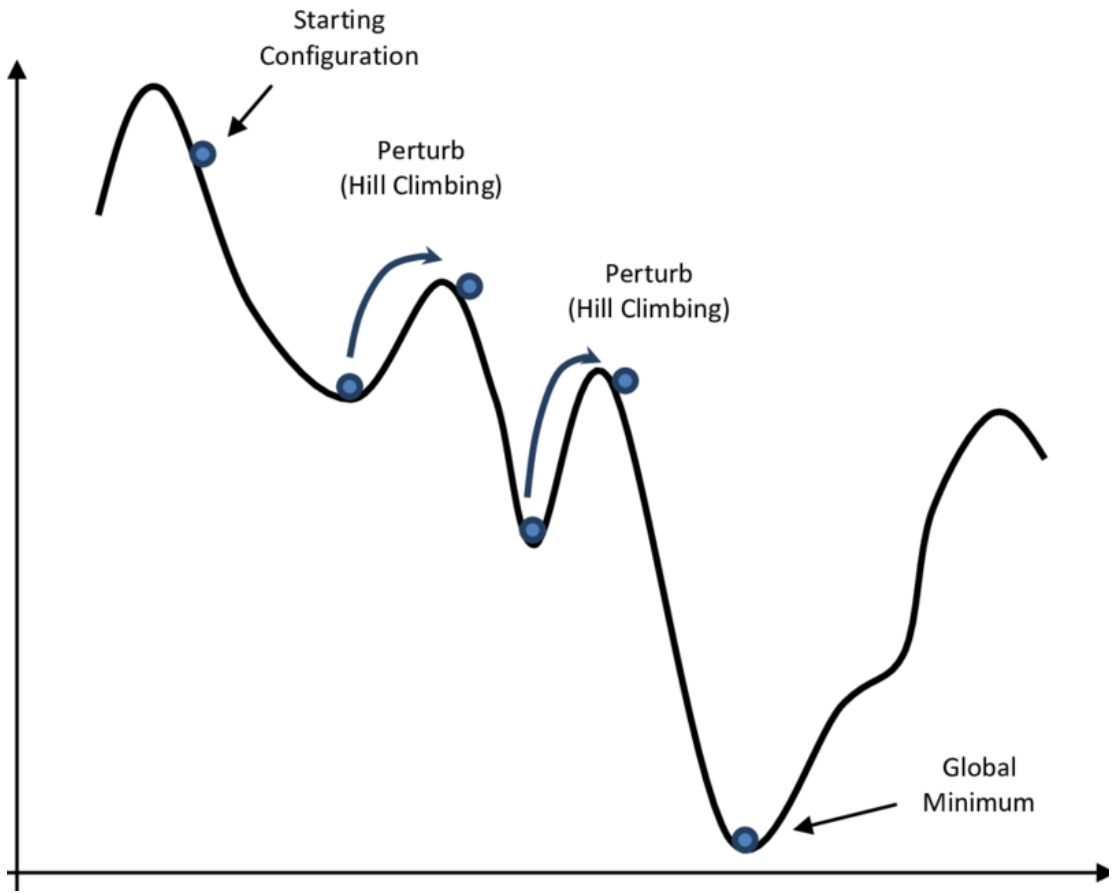


Double-Bridge Move



New Solution

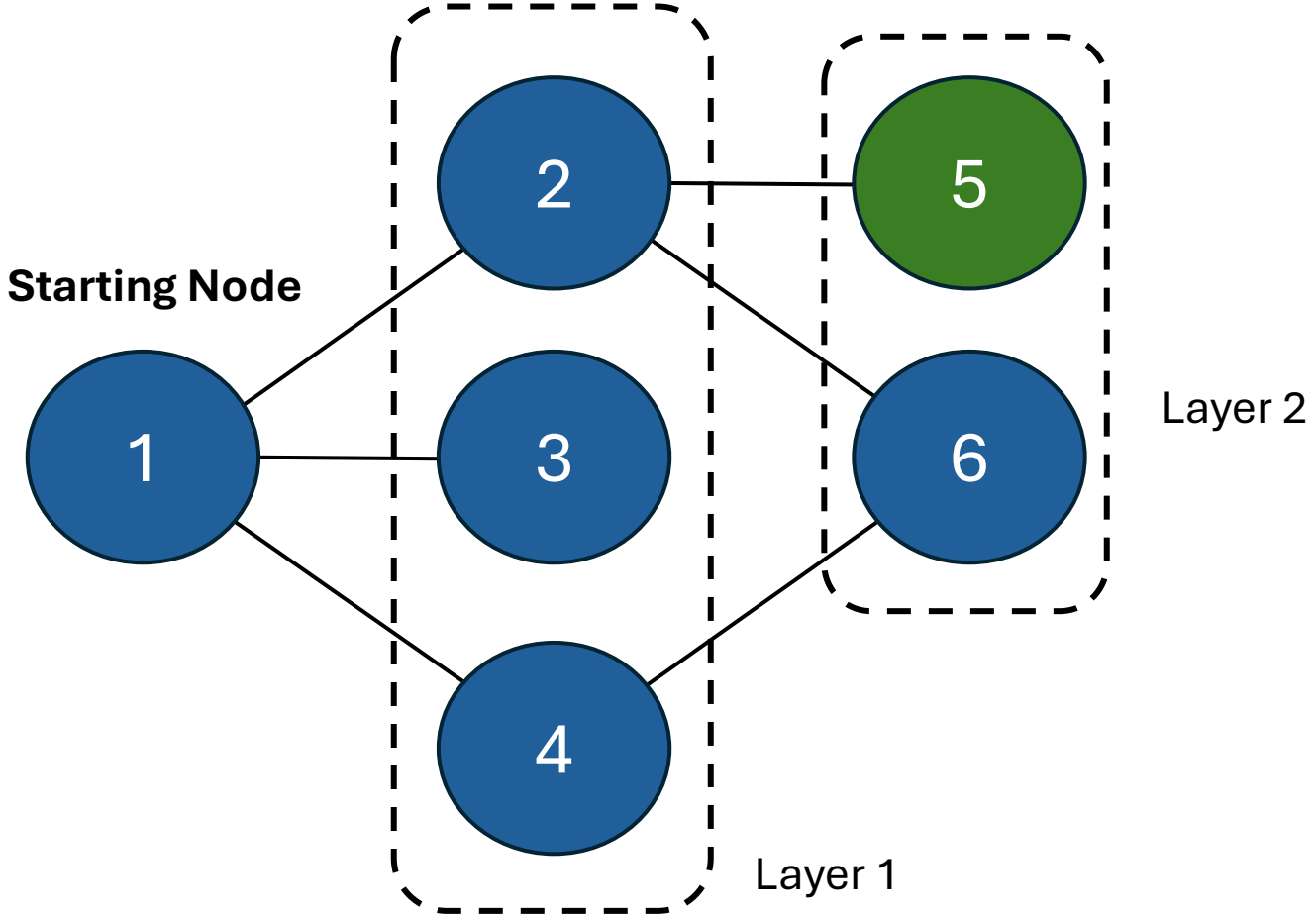
Appendix G: Simulated Annealing



Sample Perturbations:

- Shuffling
- Segment Reversals
- Random Swaps
- 2-opt

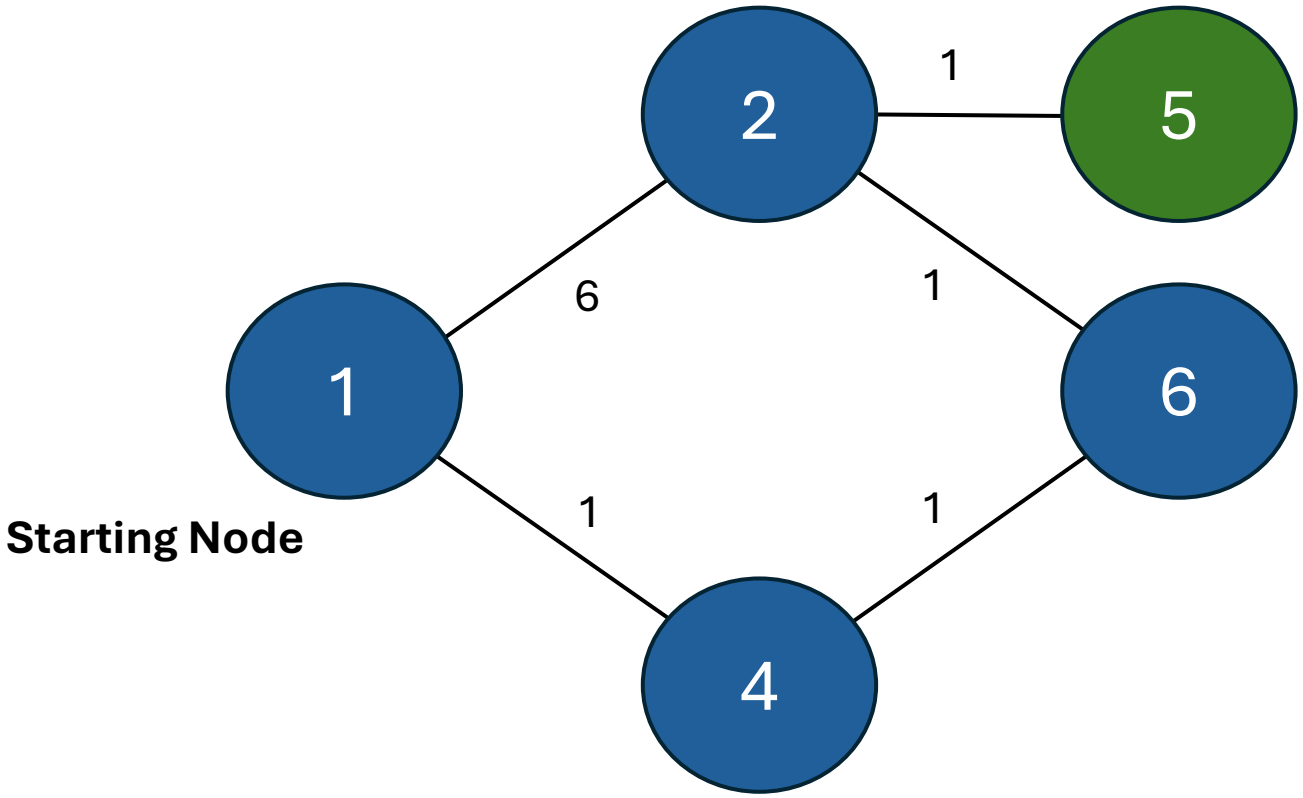
Appendix I: Breadth-First Search



Order of Search:

- 2
- 3
- 4
- 5 – Found Node!

Appendix I: Dijkstra's Algorithm



Order of Search:

- 2
- 4
- 6
- 2
- 5

Appendix J: A* Algorithm

- Similar implementation as Dijkstra's Algorithm
- Uses external information (heuristic) to decide which nodes to traverse through

- **Heuristic Used:** Straight-line distance from current intersection to destination intersection