Path-based algorithms for solving UE

Pramesh Kumar

IIT Delhi

September 21, 2025

FW criticism

- It treats all the O-D pairs equally.
  - different O-D pairs will require different fraction of flow to be shifted to shortest path. However, F-W uses same λ for each O-D pair.
- Target vectors $\mathbf{y}$ are restricted to extreme points of the feasible space.
  - This creates "zig-zag" behavior near the optimal solution.
  - CFW tries to resolve this issue up to certain extent but it still uses fairly restricted target vectors.
- It is unable to erase cycle flows.

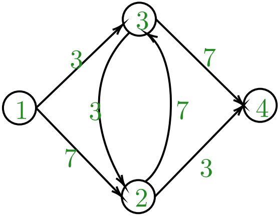

Figure: Persistent cycle in F-W[1]

[1]Figure 6.8 of BLU book

General framework of path-based methods

Path-based method keep track all the used path for each O-D pair, i.e., $\hat{\Pi}^{rs} = \{\pi \in \Pi^{rs} \mid h^{\pi} > 0\}, \forall (r, s) \ \in Z^2$

1. Initialize $\hat{\Pi}^{rs} \leftarrow \phi, \forall (r, s) \in Z^2$
2. Find the shortest path π^*_{rs} between each O-D pair (r, s). If $\pi^*_{rs} \notin \hat{\Pi}^{rs}$, then $\hat{\Pi}^{rs} \leftarrow \hat{\Pi}^{rs} \cup \{\pi^*_{rs}\}$.
3. For each O-D pair, shift travelers among paths so as to get closer to UE.
4. Update travel times and check if the convergence has reached. If converged, stop, otherwise, go to step 2.

Gradient projection method

- In this method, we update the path flows by taking a step in the opposite direction of gradient. However, this can result in infeasible path flows which we resolve by projecting the updated path flows on to the feasible path flow space H.
- We avoid storing all the paths into computer memory and generate sequentially as we need them. Such procedure is called column generation procedure.
- Also, to make the projection easy, we need to make some changes to the Beckmann's function. For this purpose, define basic path $\hat{\pi}_{rs}$ between an O-D pair $(r, s) \in Z^2$ as the path with minimum travel time between $(r, s) \in Z^2$. Define other paths $\pi \neq \hat{\pi}_{rs} : \pi \in \Pi^{rs}$ as non-basic paths.
- The flow conservation constraints can be written as:

$$h^{\hat{\pi}_{rs}} = d^{rs} - \sum_{\pi \in \Pi^{rs} : \pi \neq \hat{\pi}_{rs}} h^{\pi} \tag{1}$$

Gradient projection method

The Beckmann's formulation in terms of path flows is given by:

$$Z^{UE} = \underset{\mathbf{h}}{\text{minimize}} \quad \sum_{(i,j)\in A} \int_0^{\sum_{(r,s)\in Z^2} \sum_{\pi\in\Pi^{rs}} \delta_{ij}^{\pi} h^{\pi}} t_{ij}(x)dx \tag{2a}$$

$$\text{subject to} \quad \sum_{\pi\in\Pi^{rs}} h^{\pi} = d^{rs}, \forall (r,s) \in Z^2 \tag{2b}$$

$$h^{\pi} \geq 0, \forall \pi \in \Pi \tag{2c}$$

Writing the objective function in terms of basic and non-basic path flows,

$$\sum_{(i,j)\in A}\int_0^{\sum_{(r,s)\in Z^2}\sum_{\pi\in\Pi^{rs}}\delta_{ij}^{\pi}h^{\pi}} t_{ij}(x)dx$$

$$=\sum_{(i,j)\in A}\int_0^{\sum_{(r,s)\in Z^2}\left(\delta_{ij}^{\hat{\pi}_{rs}}h^{\hat{\pi}_{rs}}+\sum_{\pi\in\Pi^{rs}:\pi\neq\hat{\pi}_{rs}}\delta_{ij}^{\pi}h^{\pi}\right)} t_{ij}(x)dx$$

$$=\sum_{(i,j)\in A}\int_0^{\sum_{(r,s)\in Z^2}\left(\delta_{ij}^{\hat{\pi}_{rs}}\left(d^{rs}-\sum_{\pi\in\Pi^{rs}:\pi\neq\hat{\pi}_{rs}}h^{\pi}\right)+\sum_{\pi\in\Pi^{rs}:\pi\neq\hat{\pi}_{rs}}\delta_{ij}^{\pi}h^{\pi}\right)} t_{ij}(x)dx$$

$$=\sum_{(i,j)\in A}\int_0^{\sum_{(r,s)\in Z^2}\left(\delta_{ij}^{\hat{\pi}_{rs}}d^{rs}+\sum_{\pi\in\Pi^{rs}:\pi\neq\hat{\pi}_{rs}}(\delta_{ij}^{\pi}-\delta_{ij}^{\hat{\pi}_{rs}})h^{\pi}\right)} t_{ij}(x)dx$$

The Beckmann's formulation (2) can be restated as:

$$Z^{UE}=\underset{\mathbf{h}\succcurlyeq 0}{\text{minimize}}\quad \sum_{(i,j)\in A}\int_0^{\sum_{(r,s)\in Z^2}\left(\delta_{ij}^{\hat{\pi}_{rs}}d^{rs}+\sum_{\pi\in\Pi^{rs}:\pi\neq\hat{\pi}_{rs}}(\delta_{ij}^{\pi}-\delta_{ij}^{\hat{\pi}_{rs}})h^{\pi}\right)} t_{ij}(x)dx \tag{3a}$$

Let $F(\mathbf{h}) = \sum_{(i,j)\in A} \int_0^{\sum_{(r,s)\in Z^2}\left(\delta_{ij}^{\hat{\pi}_{rs}} d^{rs} + \sum_{\pi\in\Pi^{rs}:\pi\neq\hat{\pi}_{rs}} (\delta_{ij}^{\pi} - \delta_{ij}^{\hat{\pi}_{rs}}) h^{\pi}\right)} t_{ij}(x)dx$.

$$\frac{\partial F}{\partial h^{\pi}} = \sum_{(i,j)\in A} (\delta_{ij}^{\pi} - \delta_{ij}^{\hat{\pi}_{rs}}) t_{ij}(x_{ij}) = c^{\pi} - c^{\hat{\pi}_{rs}}, \forall \pi \in \Pi^{rs} : \pi \neq \hat{\pi}_{rs}, \forall (r,s) \in Z^2$$

$$\frac{\partial^2 F}{\partial {h^{\pi}}^2} = \sum_{(i,j)\in A} (\delta_{ij}^{\pi} - \delta_{ij}^{\hat{\pi}_{rs}}) t'_{ij}(x_{ij}) \frac{\partial x_{ij}}{\partial h^{\pi}}$$

$$= \sum_{(i,j)\in A} (\delta_{ij}^{\pi} - \delta_{ij}^{\hat{\pi}_{rs}}) t'_{ij}(x_{ij}) \frac{\partial}{\partial h^{\pi}} \left(\sum_{(r,s)\in Z^2} \left(\delta_{ij}^{\hat{\pi}_{rs}} d^{rs} + \sum_{\pi\in\Pi^{rs}:\pi\neq\hat{\pi}_{rs}} (\delta_{ij}^{\pi} - \delta_{ij}^{\hat{\pi}_{rs}}) h^{\pi} \right) \right)$$

$$= \sum_{(i,j)\in A} (\delta_{ij}^{\pi} - \delta_{ij}^{\hat{\pi}_{rs}})^2 t'_{ij}(x_{ij})$$

$$= \sum_{(i,j)\in \tilde{A}} t'_{ij}(x_{ij}), \forall \pi \in \Pi^{rs} : \pi \neq \hat{\pi}_{rs}, \forall (r,s) \in Z^2$$

where, $\tilde{A} = \{(i,j) \in A : (i,j) \in \pi \cup \hat{\pi}_{rs}, (i,j) \notin \pi \cap \hat{\pi}_{rs}, (i,j)\}$, set of links which are in either π or $\hat{\pi}_{rs}$ but not both.

Gradient projection method

The path flows are updated using the quasi-Newton step size. In any iteration $k+1$,

$$\mathbf{h}^{k+1} = \mathbf{h}^k - t\nabla_{\mathbf{h}^k} F$$

$$\mathbf{h}^{k+1} = \mathbf{h}^k - (\nabla^2_{\mathbf{h}^k} F)^{-1} \nabla_{\mathbf{h}^k} F$$

For a given O-D pair $(r,s) \in Z^2$, for any path $\pi \in \Pi^{rs} : \pi \neq \hat{\pi}_{rs}$,

$$(h^\pi)^{k+1} = (h^\pi)^k - \frac{c^\pi - c^{\hat{\pi}_{rs}}}{\sum\limits_{(i,j)\in\tilde{A}} t'_{ij}(x_{ij})}$$

To project it on non-negative path flow space,

$$\boxed{(h^\pi)^{k+1} = \max\left\{0, (h^\pi)^k - \frac{c^\pi - c^{\hat{\pi}_{rs}}}{\sum\limits_{(i,j)\in\tilde{A}} t'_{ij}(x_{ij})}\right\}}$$

procedure $\textsc{GradProj}(G, \mathbf{t}, \mathbf{d}, \mathsf{tol})$
    Initialize $\hat{\Pi}^{rs} \leftarrow \phi, \forall (r,s) \in Z^2$
    while $\mathsf{gap} > \mathsf{tol}$ **do**
        for $r \in Z$ **do**
            $l^*, pred \leftarrow \textsc{Dijkstra}(G, \mathbf{t}^k, r)$
            for $s \in Z$ **do**
                $\hat{\pi}_{rs} \leftarrow \textsc{TracePreds}(G, pred, s)$
                if $\hat{\pi}_{rs} \notin \hat{\Pi}^{rs}$ **then**
                    $\hat{\Pi}^{rs} \leftarrow \hat{\Pi}^{rs} \cup \{\hat{\pi}_{rs}\}$
                end if
                if $\hat{\Pi}^{rs} == \{\hat{\pi}_{rs}\}$ **then**
                    $h^{\hat{\pi}_{rs}} \leftarrow d^{rs}$
                else
                    for $\pi \in \hat{\Pi}^{rs} \backslash \{\hat{\pi}_{rs}\}$ **do**
$$(h^\pi)^{k+1} \leftarrow \max \left\{ 0, (h^\pi)^k - \frac{c^\pi - c^{\hat{\pi}_{rs}}}{\sum\limits_{(i,j)\in\tilde{A}} t'_{ij}(x_{ij})} \right\}$$
                    end for
                    $h^{\hat{\pi}_{rs}} \leftarrow d^{rs} - \sum_{\pi \in \hat{\Pi}^{rs} \backslash \{\hat{\pi}_{rs}\}} h^\pi$
                end if
            end for
        end for
        Update travel times
        Remove paths which are no longer used for each O-D pair
        Evaluate gap and $k \leftarrow k+1$
    end while
end procedure

There is another path-based method called projected gradient. In this method, we first project the search direction onto feasible space and then take a step towards that feasible direction. I encourage you to read this from the BLU book (Section 6.3.2).

Suggested reading

- BLU book Chapter 6 Section 3
- Jayakrishnan, R., Wei T. Tsai, Joseph N. Prashker, and Subodh Rajadhyaksha. A Faster Path-Based Algorithm for Traffic Assignment. University of California Transportation Center, 1994.

Thank you!