**Guideline to implementing DTALite & NeXTA in practice**

Jiangtao (Tony) Liu ([jliu215@asu.edu](mailto:jliu215@asu.edu))

Xuesong Zhou (xzhou74@asu.edu)

**Table of Contents:**

[1. Introduction to Dynamic Traffic Assignment Model 2](#_Toc389753314)

[2. Demand Side 3](#_Toc389753315)

[A: Departure time profile 4](#_Toc389753316)

[B: Lane capacity 5](#_Toc389753317)

[C: Traffic flow model 7](#_Toc389753318)

[3. Supply Side 8](#_Toc389753319)

[D: Jam density Kjam 8](#_Toc389753320)

[E: Merge/Diverge 9](#_Toc389753321)

[4. Travel Behavior 10](#_Toc389753322)

[F: Route choice model 10](#_Toc389753323)

[5. Network and Model Error Checking 12](#_Toc389753324)

[G: Basic information checking 12](#_Toc389753325)

[H: Sensor count/speed 13](#_Toc389753326)

[6. Time-dependent OD Matrix Estimation 13](#_Toc389753327)

[I: ODME 13](#_Toc389753328)

[7. Validation on the Calibrated Model 14](#_Toc389753329)

Dynamic traffic assignment is an increasingly concerned tool for traffic analysis and evaluation. Its implementation attracts practitioners’ great interest meanwhile facing several big challenges. This document is aimed at providing the procedure of applying DTA method into practical projects with approaches for those challenges. The specific tools are DTALite (a Dynamic Traffic Assignment engine) and NeXTA (a visualization tool).

# **1. Introduction to Dynamic Traffic Assignment Model**

The process of dynamic traffic assignment program can be described according to the following Figure 1.

The basic data input for a **dynamic network loading program** (**f**) includes **time-dependent** **origin-destination demand** (**a**) and the **traffic network** (**b**) with road capacity constraints on links and nodes. However, it is a challenge to obtain accurate dynamic demand and real-world network data required for practical application.

Through calculating the **dynamic travel time** (**c**) of all links (free-flow travel time is used at first iteration), the **route choice model** (**d**) embeds a standard **time-dependent least-cost path algorithm** (**e**) to generate paths for all agents. The combination between least cost and the behavior of traveler’s choice is also still one challenging issue.

The core **dynamic network loading** program (**f**) loads previously generated agents on the traffic network for the entire planning horizon, which produces **link-based traffic states** (**h**) that describes time-varying traffic states at the link level using **traffic flow model** (**g**) and updates the **dynamic travel time database** (**c**).

At the following step within an iterative assignment process, the **route choice module** (**d**) will again re-compute the route selection for each agent using **updated travel times** (**c**), for another iteration of **dynamic network loading** (**f**), until the model is converged or reaches the maximum number of iterations.



Figure 1 Procedure Of Dynamic Traffic Assignment

# **2. Demand Side**

The DTA model usually relies on the estimates from Regional Travel Demand Model that can provide peak-period demand used for static traffic assignment.

However, the peak-period demand is a fixed value and cannot satisfy the requirements for DTA implementation, 24 hour simulation for all day to capture the time-dependent congestion, so a temporal component have to be considered to obtain the dynamic demand.

In NeXTA, the temporal component is a “time-slicing ratio” that divides the total demands into individual 15-minute interval.

## **A: Departure time profile**

In NeXTA, the input of “time-slicing ratio” is in **input\_demand\_meta\_data.csv** file, and it defines **loading period** and the **departure time** for all vehicles. Table 1 is a departure time profile example:

Table 1 One Example of Departure Time Profile

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **File\_name** | **Format\_ type** | **Start\_time** | **End\_time** | **14:00** | **14:15** | **14:30** | **14:45** |
| **Demand\_Data\\SOV\_14\_15.csv** | matrix | 840 | 900 | 0.2 | 0.2 | 0.3 | 0.3 |
| **Demand\_Data\\HOV\_14\_15.csv** | matrix | 840 | 900 | 0.1 | 0.3 | 0.3 | 0.3 |
| **Demand\_Data\\HPCE\_14\_15.csv** | matrix | 840 | 900 | 0.2 | 0.2 | 0.3 | 0.3 |
| **Demand\_Data\\MPCE\_14\_15.csv** | matrix | 840 | 900 | 0.2 | 0.2 | 0.3 | 0.3 |

Project manager can estimate the **time-slicing ratio** according to his/her project experience or observed counts, the result of which should be consistent with existing traffic data (count, speed, ect.) as much as possible.

What needs to be done for time-dependent OD demand in NeXTA can be summarized in table 2.

Table 2 Summary for time-dependent OD demand in NeXTA

|  |  |  |  |
| --- | --- | --- | --- |
|  | **How to prepare** | **Input data** | **Output data** |
| **Demand** | Static Regional Travel Demand | Input\_demand.csv | Time-dependent traffic demand |
| **Departure time** | Experience or observed time-dependent traffic count | Input\_demand\_meta\_data.csv |

## **B: Lane capacity**

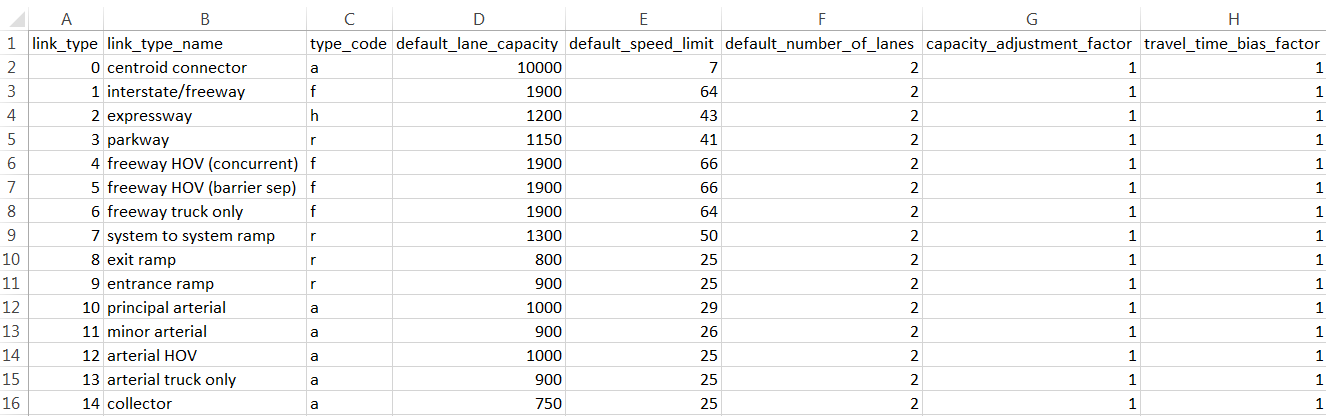
The network for DTA model is usually from **Regional Travel Demand Model**.

Through the settings in **import\_GIS\_settings.csv** file, those various GIS network shape files from Regional Travel Demand Model can be imported into NeXTA and read by DTALite.

However, the lane capacity from macroscopic travel model is **relative lower** than its real value, because for static traffic assignment, reduced capacity can make up those lost travel time, to some extent, like the real stop time at intersections.

Therefore, it needs to adjust the lane capacity for different link types.

In NeXTA, there is one column called “capacity\_adjustment\_factor” in **input\_link\_type.csv.** Project manager can estimate the factor value based on observed traffic data or his/her experience for different link type. Table 3 is the link type table for Atlanta network.

Table 3 capacity adjustment factor in input\_link\_type.csv

## 

## **C: Traffic flow model**

In DTALite, four traffic flow models are applied for traffic state calculation.

Table 4 Application of traffic flow models in DTALite

|  |  |  |  |
| --- | --- | --- | --- |
| **Traffic flow model** | **Lane capacity** | **Kjam** | **Shock wave propagation through Kjam and backward wave *w*** |
| **BPR** | Considered through volume/delay funciton, but allow  V/C >1 | No | No |
| **Point queue** | Yes(its inflow capacity is infinite, and its outflow capacity is lane capacity ) | No | No |
| **Spatial queue** | Yes(its inflow capacity is infinite, and its outflow capacity is lane capacity and storage capacity of next lane) | Yes | No |
| **Newell’s simplified KW model** | Yes  (its inflow capacity is equal to its outflow capacity under free-flow traffic state for each lane) | Yes | Yes, applied on freeway only |

For detailed illustration about link capacity considered by different traffic flow models, please refer to the DTALite white paper (<https://docs.google.com/file/d/0B7B_ItZxmow6TkNxRVA3Nk1kMVk/edit>).

# **3. Supply Side**

## **D: Jam density Kjam**

Since jam density is a key parameter for spatial queue model and Newell’s KW model, it needs to calibrate this value for the base traffic flow model.

The process is called traffic flow model sensitivity analysis based on various jam density values. Through building the connection between simulated results (MOE) with different jam density values, users can compare the simulated data with observed data.

Finally, obtain the ideal jam density to reduce the difference between simulated MOE and observed MOE.

Figure 2 is an example of traffic flow model sensitivity analysis.

Figure 2 Sensitivity analysis of traffic flow model

Based on the objective MOE, calibrate jam density to match simulated data with observed data.

## **E: Merge/Diverge**

Usually the congestion location is the ideal place to calibrate traffic flow model. Where vehicles merge/diverge belongs to such place, so it is necessary to notice this topic.

For **on-ramp** location: when the demand of mainline link and ramp link is more than the inflow capacity of downstream link, as shown in Figure 3. It needs to calibrate the distribution of the available inflow capacity for further traffic analysis.

For **off-ramp** location: as the mesoscopic DNL model moves agents/vehicles with OD and path information, the proportions of vehicles moving out of a link to individual outgoing links are in fact determined directly by the paths (downstream node sequences) associated with vehicles at this link, as shown in Figure 4, so it becomes one part of OD demand matrix calibration.

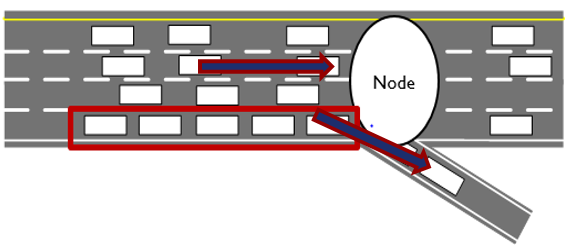
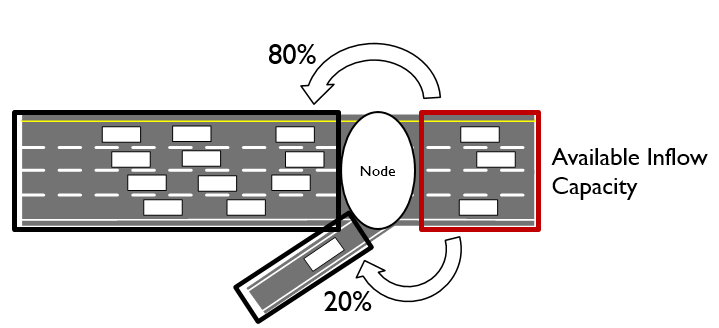


Figure 3 Illustration of merge Figure 4 Illustration of diverge

# **4. Travel Behavior**

## **F: Route choice model**

In DTALite, the route choice is based on **generalized travel cost**, so the parameters regarding the travel cost should be calibrated.

Generalized travel cost = generalized travel time + toll /VOT+ operation cost/VOT

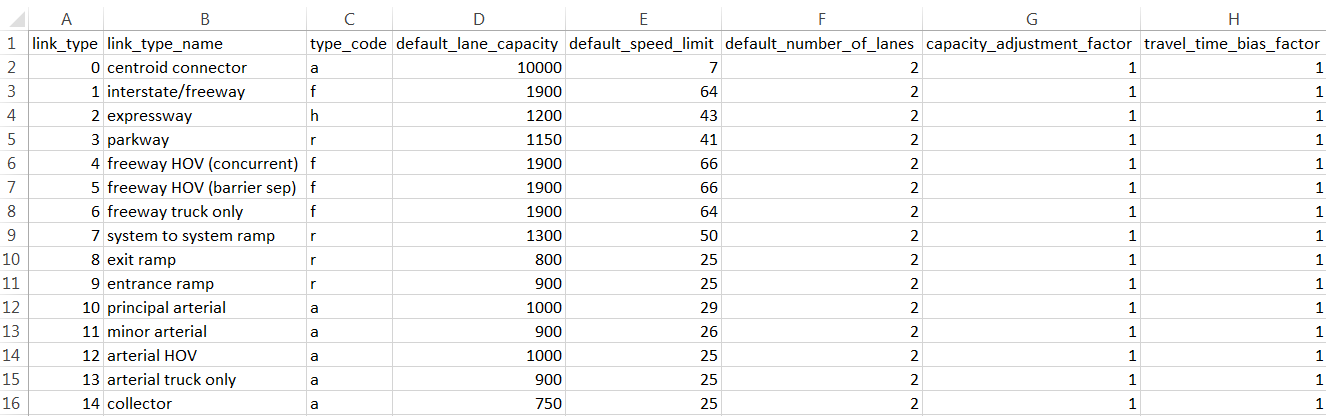
Generalized travel time = arterial travel time + coeff \* freeway travel time

Operation cost = operation cost of distance unit \* distance

Here, the **coefficient** is based on the link reliability. Generally, travelers would like to choose freeway compared with arterial links in their trips, so we can define it as 0.881.

The reference for this **coefficient** is *Network Knowledge and Route Choice* (table 5-13, M. Ramming, Dissertation at MIT).

In NeXTA, there is one column called “travel\_time\_bias\_factor” in input\_link\_type.csv. This factor is indeed the coefficient of choice preference for different link type. The following is the link type table for Atlanta network.

Table 5 Illustration of travel time bias factor in input\_link\_type.csv

In addition to coefficient for different link types, the parameters of generalized travel cost also include **Value of Time** (or distribution of value of time for different demand type) and **operation unit cost**.

In NeXTA, the value of time is defined **in input\_VOT.csv**, and the operation unit cost is defined in **input\_pricing\_type.csv**.

Based on the given toll values in **Scenario\_Link\_Based\_Toll.csv**, the parameters, such as, VOT and operation unit cost for different demand type, can be calibrated so as to match with observed data.

# **5. Network and Model Error Checking**

## **G: Basic information checking**

In NeXTA, users can check “File loading status” for those basic information in Figure 5.

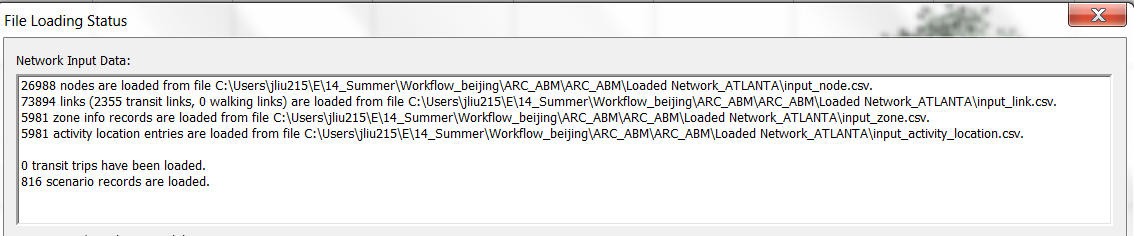


Figure 5 File loading status

Also, users can check input\_node.csv, input\_link.csv, input\_zone.csv, input\_activity\_location.csv, output\_summary.csv (one-shot simulation) to check those information in case there are some basic errors.

## **H: Sensor count/speed**

After trying to run several iterations for simulation, users can utilize sensor data (count/speed/occupancy) to compare it with the simulated results.

Usually, users can choose some specific main corridors/paths to check whether or not there are obvious errors. If so, it is necessary to recheck those basic information and the base model.

# 

# **6. Time-dependent OD Matrix Estimation**

## **I: ODME**

Through the above iterative calibration and checking on factors from A to H, the difference between observed data and simulated data maybe still cannot be ideally solved, so one approach to obtaining the balance between observed data and prior seed matrix is designed, that it, OD matrix estimation.

In DTALite, the detailed methodology and operation process about OD demand estimation can be found at the link <https://docs.google.com/document/d/1hlOgTN4C8zEVzdEMp0VBIb1DteqJoD4_ewct_qbfnnU/edit>.

# **7. Validation on the Calibrated Model**

After the above calibration, what needs to do is to validate the calibrated model.

The method is to apply those unused data into this calibrated model and compare the new simulated data with observed data.

A good match result can enhance the confidence of this calibrated model; whereas, a bad result means that a recalibration may need to be scheduled.