|  |
| --- |
| If you encounter a message showing "mfc120.dll is missing from computer:, please download Visual C++ redistribution package for 2013<http://www.microsoft.com/en-us/download/details.aspx?id=40784> as this package is required for our OpenMP-based parallel computing. |

**Module 1: Introduction to NEXTA/DTALite: (10AM-10:30 AM)**

* Two software applications: NEXTA as GUI and data hub; DTALite as DTA simulation engine
* 32\_bit vs. 64\_bit: 32\_bit for GIS shape file importing and legacy support; 64\_bit for large network: (e.g. NCSU network, 1M vehicles, 5-10 AM, 4 CPU cores, 9GB RAM, 1 hour CPU time for 20 iterations)
* Data files are in CSV format: with geometric fields (for importing from and exporting to GIS, Google Fusion Tables)
* Project folder: \*.tnp file as a reference for other data files:
  + One network per project folder
  + Prefix of input, output files
  + How to manage multiple projects (load multiple projects)



File structure:

* Different layers: different files: node, link, zone, activity locations
* Many model attribute files: node control type, link type, demand type and vehicle type
* Time representation: 24 hour (for demand and sensor data), day number= iteration number, work zone has day attributes (for modelling day-to-day learning)
* Demand meta database file: “Dynamic demand data manager”, read multiple demand files, in different format: column, matrix, agent file, DYNASMART file, different demand loading periods, additional departure time profile
* Scenario setting file: traffic flow model, traffic assignment model, scenario number for multiple scenario runs
* Scenario files for advanced modelling features: work zone, incident, tolling, VMS files
* Sensor data file: for model validation and calibration, different time period
* Output files: simulation summary, network MOE, link MOE, trajectory file

Subfolders under internal release folder

* Documentation (for data structure, users guide, QEM tool)
* Default data folder (default data attribute files )
* Sample data sets (real-world test networks)
* Importing sample data sets (GIS files, Excel, Synchro)
* Test data sets (simple networks for testing traffic flow models and other key modelling features)

Internet Resources:

* 1. Google Code (for hosting source code, latest release, bug reporting) : <https://code.google.com/p/nexta/>
* Training website (for learning material and user forum)
  1. <https://sites.google.com/site/nextadtalitetraining/>
* [www.learning-transportation.org](http://www.learning-transportation.org/) (for hosting general learning material about network modelling)

**Module 2: Working through visualization features in NEXTA (West Jordan Network) (10:30AM-11AM)**

1. Basic GUI features
   1. Turn on and off GIS layers; Move around, select node and links; Toolbars for editing networks
   2. Open project folder (CSV file format)
2. View/Edit data files in NEXTA’s “project” menu
   1. Node/link/zone/activity location
   2. Demand meta database
   3. Scenario files
3. Integration with assignment model
   1. Traffic flow model; Assignment method
4. Advanced visualization functions of NEXTA
   1. 24-hour Time control/Clock bar
   2. Volume (bandwidth), density, speed
   3. Animation and queue: (turn off node layer and bandwidth)
5. Sensor-related display
   1. Turn on sensor layer to see sensor data/locations
   2. Activate sensor data table through right click
   3. Validation Plot; and zoom to the link
6. Path-related display
   1. Manually select a path: travel time over the time
   2. Import path file
      1. Simulated vs. observed travel time series
      2. Contours of density, speed and V/C
   3. Export path summary file to Excel file to do a column chart
7. Vehicle and Summary Charts
   1. Examining route choice decisions
   2. X axis and Y axis: 1 hour and average travel time
   3. Export all summary statistics to Excel
8. Data exporting to Google Earth /GIS package
   1. 2D KML, 3D KML, GIS shape files
   2. Google Earth visualization
   3. Zone level display*: adjust height/color*

**Module 2.2: Importing network and demand data from a regional planning model (11:30AM-12:00PM)**

*Learning Goals:*

1. *Understand how to export GIS shapefiles from CUBE*
2. *Understand how to prepare importing configuration*

*Step 1*: Open and prepare VISUM network, export GIS shape files

* Open the provided Maryland Statewide model network files in Cube
* Change network to WGS 84 coordinate system (Menu > Network > Network parameters, under Spatial reference system)
* Check the available data: link types, links, nodes, zones, matrices...
* Export GIS shape files through Menu > File > Export > Shapefile
  + Give the base file name (i.e. 35S\_shapefile) and export nodes, links, zones, zone centroids and connectors
  + The corresponding files will be written to the destination folder
* Export demand matrix through \_\_\_\_\_\_

*Step 2:* Prepare the necessary CSV files

* Copy import\_GIS\_settings, input\_demand, input\_demand\_meta\_data, input\_link\_type and input\_node\_control\_type CSV files from any sample data set to the destination folder
* Prepare the import\_GIS\_settings file for the exported network for node, link, zone, centroid and connector shapefiles, as well as default settings
  + Hint: use any GIS software (such as Q GIS) to read the key values for each layer
* Prepare input\_link\_type and input\_node\_control\_type files
  + Hint: use List link types and List nodes in VISUM to read corresponding values
* Set input\_demand\_meta\_data to read the corresponding demand matrix, or copy the demand matrix into the input\_demand CSV file

*Step 3:* Import the network into NeXTA

* Menu > File > Import > GIS Planning Data Set
* Save the network as a \*.tnp file
* Check imported network (coordinate system, link, node, demand...)
* Run simulation and perform analysis

**Module 3: Introduction to DTA modelling principles (20 min)**



Typical Simulation-based Dynamic Traffic Assignment Modelling Framework

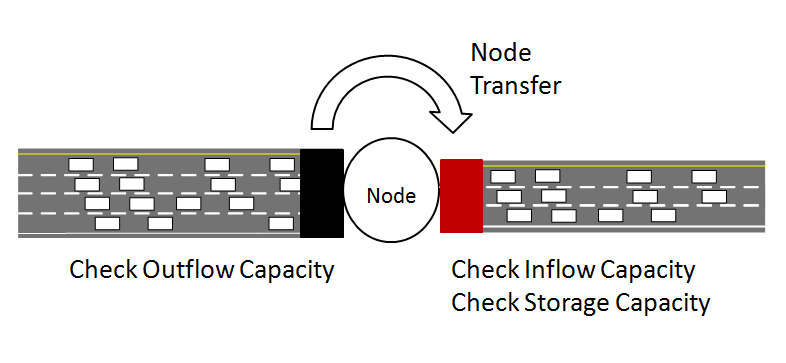
|  |  |  |
| --- | --- | --- |
|  | Key modelling component | Notes |
| D | Dynamic demand, dynamic capacity |  |
| T | Traffic flow models, link model, node model, bottlenecks (lane drop, merge and diverge, signalized intersections) |  |
| A | Equilibrium assignment and gap functions, Day-to-day learning; agent-based routing |  |
| Lite | Light-weight modelling features:   1. Computational efficiency (parallel computing for both traffic flow model and agent-based routing) 2. Signal representation (link-based, and movement-based effective green time) 3. Traffic flow model on freeway |  |



Fundamental diagram for Newell’s simplified kinematic wave model



Link traversal step: outflow, inflow, and storage capacity constraints



Node transfer: Move vehicles between links, subject to capacity constraints



Available Inflow Capacity

80%

20%

Inflow capacity allocation in DTALite using lane-based proportional model



Detailed capacity allocation at merge nodes



Diverge nodes: Inflow constraint relaxation to handle first-in-first-out (FIFO) at off-ramp bottlenecks

**Module 4: Hands-on with 3-Corridor Network:**

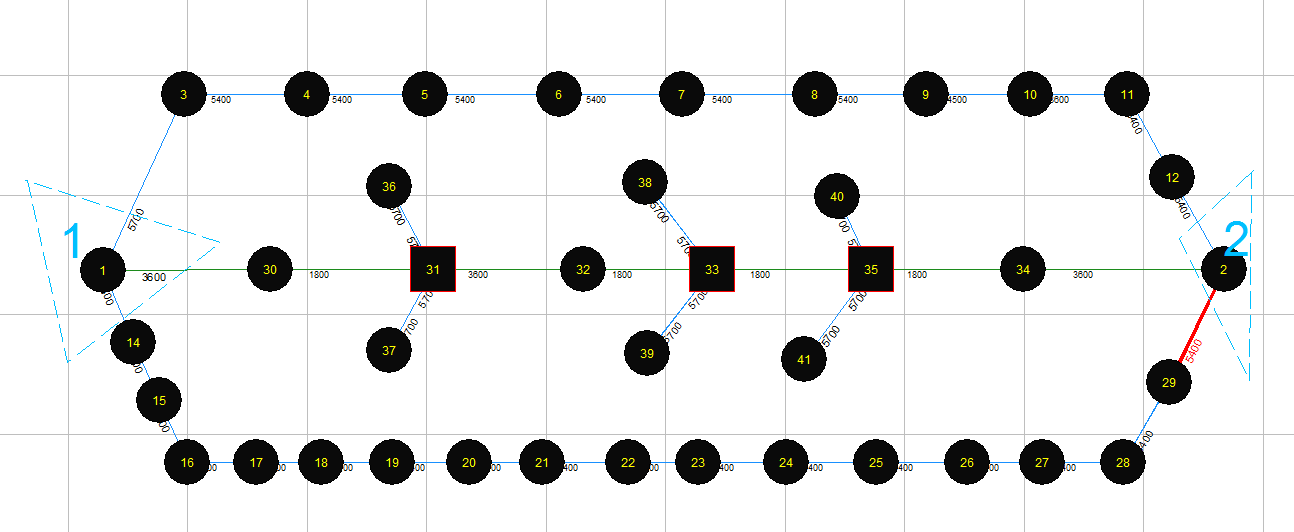
Learning objectives:

I: how to identify bottlenecks and model congestion propagation?

II: how to quantify dynamic traffic equilibrium? Gap functions, and how many iterations to achieve traffic equilibrium; different route choice behaviour at different travel times

III: how to evaluate road tolling scenarios?

1. Introduction: 2 hours of demand, bottlenecks with capacity of 3600 vehicles per link per hour on first freeway corridor



2. Demand = bottleneck capacity: demand multiplier = 1, 1 iteration

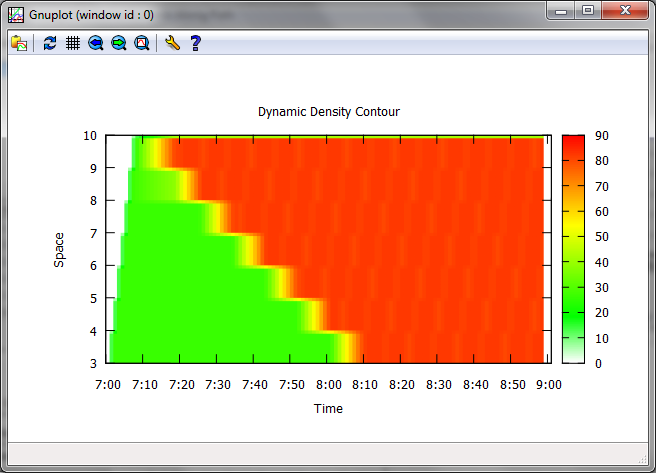
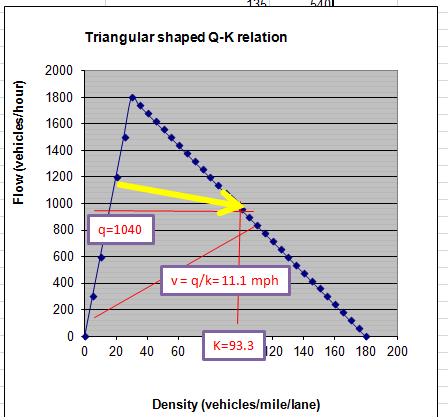
Density increase on bottleneck; Speed = free flow speed; V/C =1 on bottleneck

3. Demand multiplier = 1.2: hourly demand = 3600\*1.2 = 4320

1. slightly higher than capacity of 3600 (on downstream bottleneck)
2. slightly lower than capacity of 4500 (on upstream bottleneck)
3. bottleneck on upstream bottleneck; speed = free-flow speed on downstream bottleneck; V/C = 1 on downstream bottleneck

4. Demand multiplier = 1.3: hourly demand = 3600\*1.3 = 4680

* 1. 4680 > 4500 > 3600 (on two bottlenecks)
  2. Severe queue spillback on the loading link
  3. Queue spillback speed

Link speed = 11.4, link density = 93.3

Shockwave speed = 6.73 mph,

Propagation time per mile = 8.9 min;

Propagation time for 6 miles (from node 3 to node 9) = 8.9\*6= 53.4 min

Observation: 8:05 AM – 7:15 AM = 50 min;

Link speed = 13 mph: density close to 90 vehicles per mile per lane

5. 20 iterations; demand level = 1.3

User equilibrium; Relative gap function; After 5 iterations

6. 40 iterations; demand level = 1.3

Check 15-min gap function

1. 40 iterations: demand level = 1.5

Three paths are used; including the third path with FFTT = 17 min

After 8AM, third path is used.

Average travel time on three paths through vehicle path analysis

1. Add $0.50 toll on link 3->4

VOT = $10 per hour, additional equivalent travel time = $0.5/$10\* 60 min /hour= 3 min

VOT = $20 per hour, additional equivalent travel time = $0.5/$20\* 60 min /hour= 1.5 min

VOT = $30 per hour, additional equivalent travel time = $0.5/$30\* 60 min /hour= 1 min

Check VOT distribution and travel time through Summary Chart

|  |  |  |  |
| --- | --- | --- | --- |
| Category | WITHOUT TOLL: Avg Travel Time (min) | WITH TOLL: Avg Travel Time (min) | Avg Toll Cost ($) |
| $0-$10 | 16.03 | 16.21 | 0 |
| $10-$20 | 16.01 | 14.06 | 0.34 |
| $20-$30 | 15.97 | 13.23 | 0.48 |
| $30-$40 | 16.01 | 13.17 | 0.48 |
| $40-$50 | 15.81 | 13 | 0.5 |
| $60-$70 | 15.82 | 12.99 | 0.5 |

Check where the low-income travelers are diverted to, through vehicle path dialog

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Path No | Count | Percentage | Travel Time (min) | Distance (mile) | Speed (mph) | Toll Cost($) |
| 1 | 7294 | 67.5 | 15.8 | 11 | 41.8 | 0 |
| 2 | 3211 | 29.7 | 16.4 | 8 | 29.3 | 0 |
| 3 | 295 | 2.7 | 17.1 | 17 | 59.8 | 0 |
| Path No | Count | Percentage | Travel Time (min) | Distance (mile) | Speed (mph) | Toll Cost($) |
| 1 | 7203 | 66.7 | 13.1 | 11 | 50.5 | 0.5 |
| 2 | 3297 | 30.5 | 16.1 | 8 | 29.8 | 0 |
| 3 | 300 | 2.8 | 17.1 | 17 | 59.8 | 0 |

9. Add HOV toll by modifying input\_demand\_meta\_data.csv

Total toll revenue: 3601 (regular toll) vs. 3327 (HOV toll)