

# Evaluation of Ohio DOT (ODOT) Speed & Capacity Calculators and Network Support Tools

Following findings/comments were provided by:

**Hui Zhao, P.E.**

**FDOT District 4 - PL&EM**

954-777-4635

1. ODOT tool runs successfully in Cube 6.41. It will not run in Cube 6.11. Citilabs programmed ODOT's long standing procedures into Cube.
2. One of our consultants has used ODOT's methods before, and it appears that their procedures are usually very logical and complete.
3. ODOT tool has programmed several network error diagnostics. This appears to be a good idea. But we should tailor them to fit the FDOT method. This would include FT2, Area\_Type, and number of lanes out of range, and other missing or illogical values. We also should do the same for missing and out of range node attributes.
4. ODOT tool calculates attributes associated with network geometry. We generally don't need this, but we might need some of these things if we adopt any ODOT idea. An example of where we do such a thing is the HCM2010 method we added to SERPM7 that decreases freeway speed as function of interchange density (merge). So we calculate the number of ramps with a fixed distance from every freeway link.
5. ODOT and FDOT have taken very different approaches to calculating speeds and capacities. We may identify things that might be used to improve our methods. To a large extent, ODOT tool uses junction modeling and applies speed and capacity limitations at junctions (intersections). If we think this would be a better approach than DTA and Avenue, we might want to at least test ODOTs procedure, or talk with Greg Giaimo of ODOT to see how he feels about it now.
6. For the capacity calculators, ODOT tool builds up the capacity and applies adjustment factors. It also looks at turn lanes, green time, cycle lengths, and cross streets. FDOT uses lookup tables from the Q/LOS Manual, and applies adjustment factors. We don't directly consider turn lanes and cross streets and percent trucks, as these things are accounted for by the road class and other variables. Also, we calculate capacity in passenger cars and then use a PCE factor when we assign trucks, so we don't try to change the capacity to account for trucks. It might be useful to look more closely at intersection data and make the capacities for sensitive to these things.
7. ODOT tool starts their speed calculations with lookup tables at junction level. Posted Speed is an input attribute, but we didn't find that it was used in the free flow speed calculation.
8. ODOT tool has a turn penalty file write. We used to have MKTURNS which is a Fortran program that adds small penalties to the turn penalty file to discourage multiple turns when, for example running between diagonal corners of a grid street pattern.

Following are the findings/comments sent by:

**Daniel J. Beaty, AICP**

Chief Transportation Planner

Planning

Direct (850) 536-8523 Cell (850) 661-2472

**HNTB CORPORATION**

1276 Metropolitan Boulevard, Suite 304 Tallahassee, FL 32312 | [www.hntb.com](http://www.hntb.com)

I have reviewed the ODOT Network Utility Calculators as well as Hui's comments made above. I agree with Hui's comments and offer the following general comments:

1. Overall this is a good set of procedures. They could all be useful for us in Florida except for the fact that as it stands right now most models only have some of the required input variables and the names are not consistent from model to model.
  - a. The use of the Unified Statewide Base Map as the input network could make this process much easier.
2. If we choose to implement all of the calculators, we should consider doing so with some such as the Junction File Writer and Turn Penalty File as "Optional" to give Districts and MPOs a choice while still providing a valuable tool for Geometry, Capacity and Speed calculating/checking.
3. The fact that this will not run in CUBE 6.1 or lower may be an issue but I wouldn't consider it a big issue.

These are my step specific comments (step 1 is a loop for time periods):

1. Step 2 – Network Preparation: I believe that this step can and should be modified to work with our models. This will mean that if a developer wants to use this they will either have to conform to standard variable naming conventions or they will have to modify the scripts for their specific model. I prefer the former.
2. Step 3 – Geometry: This step is quite useful but will require extensive updating for our current approach
3. Step 4 – Capacity Calculator: We should look into using the ODOT approach to calculating capacity IF we are going to implement junction data in our models. If not, we can delete this step all together as we have the lookup table process in our current models.
4. Step 5 – Speed Calculator: We currently use 2 different approaches to initial or "free flow" speeds. The use of a look up table or the posted speed limit. There has been a shift recently towards the use of posted speeds. If this is a continuing trend then this speed calculator is of no real use to us. However, I do believe that a test model should be validated using the HCM based approach in the ODOT calculator to see if it yields a better validation and then decide if we should look to implement it.
5. Step 6 – Turn Penalty File: We could implement this but this really goes hand in hand with the junction information so if we don't incorporate junction data this step is not useful.
6. Step 7 – Junction File Writer: Again, useful if we implement junction information only.
7. Step 8 – Final Outputs: We should use this and modify for our needs based on which other steps we decide to implement.