A Temporal Domain Decomposition Algorithmic Scheme for Efficient Mega-Scale Dynamic Traffic Assignment – An Experience with Southern California Associations of Government (SCAG) DTA Model

Yi-Chang Chiu
Department of Civil Engineering and Engineering Mechanics, University of Arizona, 1209 E 2nd St. Tucson, AZ, USA
chiu@email.arizona.edu
520-626-8462

Eric J. Nava
Department of Civil Engineering and Engineering Mechanics, University of Arizona, 1209 E 2nd St. Tucson, AZ, USA
ejnava@email.arizona.edu
520-626-3367

Hsi-Hwa Hu, Ph.D.
Transportation Modeler
Research, Analysis and Information Services Division
Southern California Association of Governments
818 W. Seventh Street, 12th Floor
Los Angeles, CA 90017
hu@scag.ca.gov
213-2361834

Words: 2,113
Tables and Figures: 4
Total: 3,113
Submitted to the 4th Conference on Innovations in Travel Modeling (ITM)

1 Corresponding Author: Telephone: 520-626-8462; Email: chiu@email.arizona.edu
1 INTRODUCTION

Dynamic Traffic Assignment (DTA) methodology is ever-evolving, growing both theoretically and practically since the influential research of Merchant and Nemhauser (1978a; 1978b). Historically, DTA research has branched into two modeling approaches that can be generally categorized as analytical or simulation-based methods. A simulation-based dynamic traffic assignment (SBDTA) model typically employs a traffic simulator as the network loading method to capture complex demand/supply interactions, whereas analytical models apply mathematical formulations such as exit/volume-travel time functions for similar purposes.

The common algorithmic framework of a SBDTA model includes the iterative execution of network loading (simulation), the path set update procedure (time-dependent travel time cost of routes for all origin, destination, and departure time triplets), and the path set adjustment (DTA) procedure to update vehicle paths. To determine how close the current solution is to the DUE condition, the evaluation of path assignments, by means of simulation, requires checking a defined convergence criterion. The algorithm terminates if the stopping criterion is met.

With rising interest in SBDTA from practitioners and researchers who recognize the potential capability of the DTA methodology, deployment of SBDTA models for traffic operations and transportation planning have increased. In parallel, research and development of innovative approaches of the SBDTA model have improved, particularly in the realms of DUE solution methodology and traffic simulation. However, computational requirement for large-scale implementation of a DTA model remains to be one of the great challenges. Network size, e.g., links, nodes, and zones, the amount of vehicles being simulated/assigned and the length of the analysis period are the key factors determining the computational requirements of the SBDTA model, thus potentially making assignment procedures intractable.

In this study, the Method of Isochronal Vehicle Assignment (MIVA) is proposed in which the temporal domain of the TDSP and path assignment is decomposed, thereby improving the scalability of the SBDTA model and allowing for simulation and assignment of long-period, large-scale models. The self-tuning adaptive (STA) algorithm is also developed to search for an optimal MIVA specification during run-time. To the best of our knowledge, this paper is the first known effort to enable SBDTA for region-sized networks with diurnal demand modeling period.
2 THE METHOD OF ISOCHRONAL VEHICLE ASSIGNMENT FOR DYNAMIC TRAFFIC ASSIGNMENT

The proposed Method of Isochronal Vehicle Assignment (MIVA) decouples the time domain between simulation and both the path set update and path adjustment procedures (comprised of both the TDSP and assignment solution algorithm) into forward-sliding time periods which allows the memory requirement for both the path set update and path adjustment to be bounded solely on the length of the determined temporal segment instead of the entire analysis period.

2.1 The MIVA Temporal Decomposition Framework

Shown in Figure 1, the MIVA scheme is denoted by two inter-associated time periods: Epoch and Projection Period. The MIVA scheme decouples the temporal domain of the analysis period (also termed simulation period) into sequential segments of equal length called Epochs. For vehicles departing within a single Epoch, the arrival times to their destination are used to estimate the time period, known as the Projection Period, in which the T domain for the TDSP algorithm is defined for path set update for vehicles departing within the current Epoch. At the end of one Epoch, all TDSP and assignment-related memory is de-allocated, and then re-allocated for the next Epoch. The MIVA scheme then slides the path set update and adjustment operations from one Epoch to the next until completing all Epochs. As a result, the memory usage during the entire path set update and adjustment operation is only a function of the Epoch length.
Figure 1: The MIVA computational scheme implemented within the SBDTA algorithmic framework.

The Epoch is the partitioned period that acts as the temporal segment for the path adjustment procedure, meaning the TDSP and assignment procedure is bounded solely by the length of the Epoch. An Epoch consists of multiple assignment intervals (interchangeably termed as departure intervals as assignment is performed for vehicle departing at the same departure interval). An aggregation interval pertains to the time interval in which traffic data, i.e., time-dependent link travel times and intersection delays, are averaged to be the input for the TDSP.

The Projection Period, $P(e^S)$ is defined as the set of assignment intervals for each Epoch $e^S$. Let $P(e^S) = \{\tau^{(s-1)b+1}, \tau^{(s-1)b+2}, ..., \tau^{sb}, \tau^{sb+1}, ..., \tau^{sb+y}\}$, $\forall e^S \in E$ be the set of assignment intervals contained in the Project Period for Epoch $e^S$. By definition, the start of the Projection Period is synchronized with each associated Epoch. However, the Projection Period is extended beyond the end of the Epoch by $\{\tau^{sb+1}, ..., \tau^{sb+y}\}$. This temporal extension is to
allow the TDSP to solve for the later arrival times of those vehicles departing toward the end of the Epoch.

3 NUMERICAL ANALYSIS OF MIVA COMPUTATIONAL PERFORMANCE

The MIVA scheme was implemented in a SBDTA model DynusT (Chiu, Nava et al. 2011; Nava and Chiu 2012). Three networks, as shown in Figure 2, were used for the initial evaluation of the computational performance of the MIVA scheme. Note that all networks for this batch of performance runs were done on a shared-memory computer with a 2.80 GHz Intel Pentium D Dual-Core Processor with 4GB of RAM. Comparisons for each network were benchmarked by simulating each network without the MIVA scheme implemented (termed full-scale). The goal of the comparison was to investigate the computational memory efficiency that can be achieved by the MIVA scheme. There are numerous choices of an Epoch set $E$ size since $e^s$ can be of various equal number of assignment intervals; therefore, $n$ (number of Epochs in Epoch set $E$) can be of different sizes. For each network, different Epoch sets were used to test the MIVA scheme, and described below for each network.

(a) Fort Worth, TX Network

(b) Guam Network
3.1 Peak Memory Usage

The testing results show that for the Fort Worth network, the full scale case required 90.0 megabyte (MB) peak memory usage. For a 2-Epoch case, the peak memory usage dropped from 82.0 to 77.0 MB. As expected, lower percentile $\varphi$ leads to a shorter Projection Period length and less memory usage. For the 3-Epoch case, the overall peak memory usage continued to reduce to the range of 78 to 73.6 MB. The lowest memory usage was observed for the 1st Epoch in the $\varphi = 0.7$ and 5-Epoch case. Comparing the highest and the lowest memory usage, the maximum memory savings was 24.2%.

The Guam network, the peak memory for the full-scale case was 70.2 MB. The memory usage continued to drop as low as 50.9 MB with the increasing number of Epochs and lower percentile value. The maximum memory savings was about 27.0%. Consistent memory savings was also observed for the Minneapolis dataset in which the maximum peak memory usage occurred at the full-scale scenario with 598.6 MB and the lowest memory usage was 430.6 MB in the 6-Epoch case. The maximum memory savings was about 28.2%.

From the results, it becomes apparent that memory saving always increases with increasing number of Epochs. This is a desirable computational property as a model user can always adjust the number of Epoch according to the memory limitation of the intended computing environment. This accomplished the primarily goal of designing a memory-scalable SBDTA solution algorithm.

3.2 Run-Time Performance

For the Fort Worth network, it is noted that the run-time for the full-scale case is 590 seconds for 100 iterations. For all Epoch settings, the run-time remains lower than 590 seconds for all ratios setting except $\varphi = 1.0$. For the Guam network at 50 iterations, the 2-Epoch case with $\varphi = 0.7$ achieved 1390 second, which was lower than the full-scale case; however, at worst, the run-time for the 6-Epoch case with $\varphi = 1.0$ was found at 2,679 seconds, a 91% increase from the full-scale case. In the Minneapolis, MN network at 50 iterations, the memory usage for the full-scale case was 50,813 seconds. Most of the cases were worse than the full-scale case, except the 5-Epoch case with $\varphi = 0.8$ with 49,413 seconds, a 2.7% savings. The worse case,
being the 5-Epoch case with $\varphi = 0.9$, was 10.9% higher than the full-scale case with 56,354 seconds.

4 EXPERIENCE WITH SCAG NETWORK

The Southern California Associations of Governments (SCAG) planning region encompasses over 20k roadway center line miles. The planning network consists of more than 31k nodes, 82K links, and 4109 TAZs. This is arguably the largest planning network in the U.S. Performing a DTA analysis on the entire region over the entire 24-hour period has strategic importance. First, SCAG is well known as one of the most congested metropolitan area in the country as reported by TTI mobility study. Having a dynamic network model to properly capture the traffic dynamic and congestion spatial and temporal pattern is the key to all congestion alleviation strategy formulation. Many policy questions and traffic management strategies can no longer be answered by traditional static assignment based network models. A step toward improving this limitation is the activity-based model (ABM) under development for SCAG. With this ABM model approaching completion it is of great need to have a functional DTA model that is able to seamlessly integrate with the ABM. This means that the DTA model needs to operate for the entire region over the daily period. An operational regional DTA model also serves as the robust and more accurate foundational model for any sub-area analysis often requested by corridor analysis. Because a simulation-based DTA model properly could account for capability constraints and congestion outside the sub-area. The arriving demand at the sub-area boundary would be much more realistic than what would be produced by a static assignment model.

Building such a mega-size DTA model is unprecedented. The proposed innovative computational scheme makes such an attempt feasible. The research team will first convert the entire SCAG model into DynusT format and the research team will apply the proposed MIVA approach to perform 24-hour simulation assignment for the entire SCAG regional model. Figure 3 shows the progress of network conversion from existing planning model to DynusT format. The initial testing of the SCAG network show 10GB peak memory usage in simulation and 3GB peak memory usage during assignment for three-hour simulation and assignment. This means that the overall peak memory usage is 10GB.

Further work is underway to test the 24-hr simulation/assignment. It is anticipated that the memory usage won’t increase drastically from this initial test run as the peak memory required by simulation is only related to number of vehicles exist in the network in a given time.
and the peak memory for assignment is decoupled from the length of analysis period and is capped at about 6GB. Further run time requirement results will be reported at the conference.

Figure 3: DynusT SCAG Network Shown in DynuStudio

(a) Vehicles Animation
(b) Cumulative Volumes

Figure 4: DynuST Vehicle Simulation (a) and Cumulative Volumes (b) of SCAG Model shown in DynuStudio

5 IMPACT FOR THE SCIENCE AND PRACTICE OF TRAVEL MODELING

With ever increasing computational power, it is always argued that DTA run time will no long be an issue in the future. However, the computational issues need to be addressed now as more applications find DTA a much needed tool to supplement existing modeling capabilities. The proposed MIVA scheme has been shown to demonstrate very promising results in small- to mid-size networks, but being able to efficiently operate on any mega-size network is the ultimate challenge and testimony to the practicality of simulation-based DTA model to users. This project will provide unique and valuable insight from the SCAG model experience.

6 REFERENCES