This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

A Monte Carlo Approach to Simulate the Stochastic Demand in a Continuous Dynamic Traffic Network Loading Problem

María Teresa Sánchez-Rico, Ricardo García-Ródenas, and José Luis Espinosa-Aranda

***Abstract*—Dynamic traffic assignment models are mathematical tools used for traffic management and control. These require a dynamic network load (DNL) model, a route choice model, and a mechanism to ensure the relationship between the submodels. The DNL problem aims to find, on a congested network, dynamic traffic volumes and travel times for a given time period. The DNL problem involves a high computational cost; thus, the model becomes intractable in real time and, often, on offline applications. This paper proposes a discrete event algorithm for the continuous DNL problem based on flow discretizations, instead of time discretizations. These discretizations create homogeneous traffic packets according to their route. The algorithm propagates the packets synchronously across the links. The dynamic mechanism used in the network links are based on a generalization of the whole-link travel time model, which divides the links in the *running section* and the *vertical queue section*. The first one is associated with the travel time, and the second one is associated with the capacity. A generalization of the point-queue model is introduced to tackle dynamic link capacities such as signalized intersections. Under certain assumptions, the resulting model satisfies the first in, first out rule, and it is used to obtain a computationally tractable model. It allows stochastic demands to be dealt with a Monte Carlo simulation approach. This scheme is computationally expensive but can be addressed through distributed computing techniques. The method and its implementation by using parallel computing techniques is assessed using the Nguyen–Dupuis and Sioux Falls networks.**

***Index Terms*—Dynamic network loading (DNL), mesoscopic simulation, Monte Carlo simulation, stochastic traffic flows.**

I. Introduction

The complexity of current transportation systems implies the need for analytical tools capable of providing an adequate information system and making predictions in real time. The traffic assignment models describe the behavior of users on a traffic network. The static assignment models assume that, over a time period, demand conditions remain the same, i.e., these models assume steady-state flows. The evolution of static models to dynamic ones is due to several reasons, including the inability to explain flow changes in short time periods and the underestimation of total travel times.

Manuscript received July 16, 2013; revised October 31, 2013; accepted January 31, 2014. This work was supported by Ministerio de Economía y Competitividad (Spain) under project TRA2011-27791-C03-03. The Associate Editor for this paper was V. Punzo.

The authors are with the Departamento de Matemáticas, Universidad de Castilla-La Mancha, 13071 Ciudad Real, Spain (e-mail: teresanchezr@gmail.com; Ricardo.Garcia@uclm.es; JoseL.Espinosa@uclm.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2014.2305473

Dynamic traffic assignment (DTA) models consider that users try to minimize their travel time, continuously updating their chosen routes according to traffic conditions. The DTA has substantially evolved since the seminal work of [1].

In order to be able to create a unified theory of DTA modeling (as occurs in the static case), it is necessary to compare and to establish the relationships between the existing models. In [2], it is considered that a basic theory of DTA models should include at least realistic models of the following: 1) traffic propagation when the paths used are known; this submodel represents the continuous dynamic network loading (CDNL) problem; 2) path choice model when the time-varying link times are known; and 3) equilibrium to reconcile the predictions of items 1) and 2). A specific DTA model can be viewed as a combination of the previous three submodels.

Two loading procedures can be found in the DTA literature.

– *Simulation-based methods.* Generally speaking, loading procedures based on simulation work with individual entities (vehicle or vehicle packet). At each simulation step, the location of each entity within the network is recorded. Such loading procedures provide more flexibility and can, accordingly, deal with more complex traffic phenomena, obtaining more realistic loading results. Nevertheless, these procedures share certain drawbacks as well. For instance, as much more detailed information about both the network and vehicles is required, the resulting loading procedures usually prove to be computationally intensive (often slow and memory consuming). The simulation is called *microscopic* when the movements of the vehicles are individually traced. DynaMIT [3] represents a simulation-based real time system that estimates and predicts the state of a transportation network adopting such an approach. For reviews on microscopic simulation-based models, see [4]–[6].

  The *mesoscopic* simulation assumes that the vehicles are clustered into packets, and the link performances are expressed in an aggregate way (see [7]–[11]). Since its emergence, the point-queue (PQ) model (see [12]) has been widely used as a link model due to its good predictions at low computational cost (see [13]). Nowadays, more realistic approaches of queue (see [14]) have been developed to give a more detailed representation of congestion effects.

  Currently, the mesoscopic CDNL model has been extended and applied to multimodal flow and pedestrian flow, in ordinary and emergency conditions (see [15]–[17]).

– *Macroscopic approaches.* Loading procedures based on analytical models. These approaches work with macroscopic traffic flow variables that describe the average behavior of vehicles within the network. The variables are time dependent and describe states such as inflow/outflow rate, volumes, and travel times for each link in the network. Thus, these procedures are comparatively simple to implement and efficient to compute, even for large road networks, but are often believed to be less accurate in capturing complex traffic phenomena than simulation-based loading procedures. A review of analytical CDNL models is presented in [18].

All aforementioned loading models are deterministic because they capture average traffic states but they do not deal with the stochastic nature of the demand. The main existing types of stochastic dynamic traffic models are based on time series models (see [19] and [20]) or Bayesian networks (see [21]–[24]).

These papers are focused on estimating travel times using sensors. The problem of travel time *prediction* is different from *estimation*. These methods are able to obtain the travel times by tracking the driving routes on real networks or on traffic models. In the context of travel time predictions, the references for stochastic load models are scarce.

The parameter calibration of traffic models has a bilevel structure (see [25]–[30]), in which the lower level is given by the traffic model, and the upper level is focused on the estimation problem. Calibration and prediction under uncertainty requires massive evaluation of these models, which makes the computational aspect a major barrier. Moreover, recent studies have shown that CDNL problem consumes the larger part of the running time in the solution algorithm (see [31]). As the size of a real road network could be large, DTA solution algorithms require efficient loading procedures.

Discrete event algorithms are widely used in transportation research, for example, in railway research, for detecting the conflicts produced in a given schedule [32] or for solving the vehicle rescheduling problem in cases of emergency [33].

This paper discusses the efficient resolution of the mesoscopic CDNL model based on a discrete event simulation. The adopted mesoscopic CDNL model is a broader version of the whole-link model of [14]. This link model aims to find a tradeoff between computational tractability and congestion representation.

The main contributions of this paper are the following.

– A discrete event algorithm proposed to solve the mesoscopic CDNL model. The main characteristic is high computational performance. In this scheme, unlike with the traditional time discretization models, at each time step, only one packet is updated (event).

– Distributed computing approaches can be adopted to address the computational burden for solving the stochastic CDNL model (see [34] and [35]). In this paper, parallel computing techniques are used to address the demand uncertainty by implementing a Monte Carlo simulation approach.

– The adopted whole-link travel time model allows time-dependent reduction of the network capacity to be tackled. The resulting CDNL model presents a first in, first out (FIFO) behavior, which is used to accelerate the procedure.

The proposed discrete event algorithm requires reduced computational costs, which allows its implementation in real time. This enables the development of efficient tools capable of evaluating travel times in real time. This challenge is central to route recommendation problems in intelligent transportation systems. From an offline point of view, this computational advantage enables the consideration of stochastic demands, which allows operational planning of traffic network incorporating demand uncertainty. The CDNL model with stochastic demand obtains the probability distribution of the travel times. The reliability-based traffic assignment models [36], [37] require these inputs within the route choice model.

This paper is structured as follows. In Section II, the basic notation and the formulation of the model are described. In Section III, the proposed solution procedure and its computing components for the network simulation are introduced. Numerical experiments are carried out in Section IV, and finally, some conclusions are drawn in Section V.

II. CDNL Model

A CDNL problem consists of, given the path flows and link performance functions, determining time-dependent network flow conditions such as link travel times, link counts, link inflows, and link outflows. An accurate way of utilizing a DNL approach is to formulate the DNL problem as a continuous-time system of nonlinear equations expressing link dynamics, flow conservation, flow propagation, and boundary conditions. Here, we describe the proposed CDNL model.

A. Nomenclature

The general notations of the variables and indices are given below.

Sets and index

$\mathcal{N}$	Set of nodes.
$\mathcal{P}$	Set of paths.
$\mathcal{O}$	Set of origins.
$\mathcal{A}$	Set of links of the traffic network.
$\mathcal{W}$	Set of origin–destination (O–D) pairs.
i	Packet.
a	Link.
p	Path.
w	O–D pair.
t	Time.
$\mathcal{A}_p$	Set of links of path p.
$\mathcal{P}_\omega$	Set of paths for the O–D pair ω.
θ	Random variable that defines the parameters for the demand.
$(0, T)$	Simulation period.
$(0, T_\infty)$	Time interval from the instant when flows enter the network to the last instant when all flows leave the network.

Functions

$h_p(t)$	Inflow rate at the origin of path p at time t.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

$u_a^p(t)$	Entrance flow (inflow) to link a from path p at time t.
$v_a^p(t)$	Exit flow (outflow) to link a from path p at time t.
$V_a^p(t)$	Cumulative exit flow (outflow) to link a from path p at time t.
$x_a^p(t)$	Number of vehicles on link a from path p at time t.
$\tau_a(t)$	Exit time from link a associated with entering at time t.
$u_a(t)$, $v_a(t)$, $x_a(t)$	As previous definitions given for paths but for aggregate flows.
$u_a^r(t)$, $v_a^r(t)$, $x_a^r(t)$, $\tau_a^r(t)$	As previous definitions given for paths but for the running section.
$u_a^q(t)$, $v_a^q(t)$, $x_a^q(t)$, $\tau_a^q(t)$	As previous definitions given by paths but for the vertical queue section.
$d_\omega(t,\theta)$	Intensity of demand at time t.
α_a	Travel time in the running section of link a.
$\beta_a(t)$	Capacity of link a at time t.
$\mathbb{P}(p,t)$	Choice probability associated to path p at time t.

Algorithm

$\mathcal{T}$	Simulation clock.
$\mathcal{Q}_a$	Set of packets that are currently in the running section of link a.
ϕ_a	Last instant in which a packet has left link a.
$\widehat{a}_i$	Current link for packet i.
$\widetilde{a}_i$	Next link for packet i.
τ_i	Instant when packet i leaves its current link.
n_i	Ordinal number of the current link within the followed path p.

B. *Formulation of the Basic Problem*

Consider a strongly connected traffic network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, where $\mathcal{N}$ and $\mathcal{A}$ denote the sets of nodes and directed links, respectively. Set $\mathcal{N}$ contains two types of nodes, which represent the intersections and the so-called *centroids*, which are dummy nodes representing generation and attracting zones for the traffic demand, i.e., access/egress nodes of trips. The links are classified into *centroid connectors*, which link the centroids with the intersections, and *normal links*, which represent roads. An example of this modeling is given in Fig. 1, which represents the Nguyen–Dupuis network topology. The centroid connectors are represented as discontinuous lines, and the road links are represented as solid lines. The centroids are represented as triangles, and the nodes are represented as circles.

The CDNL model can be formulated with the following system of equations:

- *Flow conservation* equations

$$x_a^p(t) = x_a^p(0) + \int_0^t \left[u_a^p(\xi) - v_a^p(\xi)\right] d\xi \quad \forall p \in \mathcal{P}, \forall a \in \mathcal{A}_p \quad (1)$$

$$u_{a'}^p(t) = h_p(t) \quad \forall p \in \mathcal{P} \quad (2)$$

where a' is the first link (centroid connector) of path p.

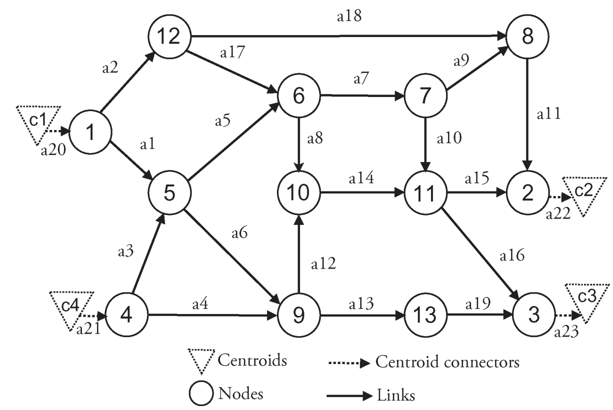

Fig. 1. Nguyen–Dupuis network showing the type of links and nodes.

The flow conservation equation between two consecutive links can be expressed as

$$v_p^a(t) = u_p^{a^-}(t) \quad \forall p \in \mathcal{P}, \forall a \in \mathcal{A}_p \quad (3)$$

where a and a^- are two consecutive links of path p.

- *Flow propagation* equations describe the flow progression over time as

$$V_a^p(t) = \int_{I_a(t)} u_a^p(\xi) d\xi \quad \forall p \in \mathcal{P}, \forall a \in \mathcal{A}_p \quad (4)$$

where

$$I_a(t) = \{\xi : \tau_a(\xi) \leq t\} \quad (5)$$

and $\tau_a(t)$ is the exit time from link a for a vehicle entering at time t.

- *Boundary conditions.* For simplicity, they are assumed to be empty at $t = 0$, i.e.,

$$v_a^p(0) = 0 \quad x_a^p(0) = 0 \quad u_a^p(0) = 0, \quad \forall p \in \mathcal{P}, \forall a \in \mathcal{A}_p. \quad (6)$$

Equations (1)–(3) model the conservation principles with regard to hydrodynamic analogy. Equations (4) and (5) describe the flow progression over time. The flow entering link a at time t exits the link at $\tau_a(t)$, and set $I_a(t)$ describes the elapsed time when the flow leaves link a by time t. Therefore, the relationships between aggregated and disaggregated variables are calculated as follows:

$$x_a(t) = \sum_p \delta_{ap} x_a^p(t) \quad (7)$$

$$v_a(t) = \sum_p \delta_{ap} v_a^p(t) \quad (8)$$

$$u_a(t) = \sum_p \delta_{ap} u_a^p(t) \quad (9)$$

where δ_{ap} is the the link–path incidence matrix, with $\delta_{ap} = 1$ if link a is on path p and $\delta_{ap} = 0$ otherwise. In the preceding expressions, $x_a(t)$ is the link volume, i.e., the number of vehicles on a link at time t.

C. *Adopted Link Dynamic Model: A Generalized PQ Model*

The CDNL model described above is completely defined by specifying how to calculate travel times on the links.

The link traversal time can be computed by using whole travel time (WTT) models. These models assume the existence of instantaneous changes in the propagation of density within the links of a network and reflect the behavior of the vehicles in a macroscopic way. A generic macroscopic whole-link model for a given link a is formed by the following equations:

- *Flow conservation* equation

$$x_a(t) = x_a(0) + \int_0^t [u_a(\xi) - v_a(\xi)]\, d\xi \tag{10}$$

- *Flow* behavior (one of)

$$\tau_a(t) = f\left(\Gamma, \Omega(t)\right);\ \text{delay-function model} \tag{11}$$

or

$$v_a(t) = g(\Gamma, \Omega(t));\ \text{exit-flow function model} \tag{12}$$

- *Flow propagation*

$$\int_t^{\tau_a(t)} v_a(s) ds = x_a(t) \tag{13}$$

where Γ is a vector of parameters reflecting the physical characteristics of a specific link such as the free-flow travel time and bottleneck capacity, and $\Omega(t)$ is a vector of the current link state variables such as $(x_a(t), u_a(t), v_a(t))$.

Reference [38] indicated that only two special cases of WTT models depict transportation-like phenomena. Each model denotes a link with no spatial dimension containing a PQ or a link with constant travel time and no queueing. Road segments exhibiting both phenomena must be represented by two links in series. This is how the basic PQ model operates. The PQ model has been implemented in several DTA studies [39]–[42].

In this paper, a generalized PQ model is proposed and can be described as a concatenation of two link models: the so-called *running section*, which models the travel time in the link, and the *vertical queue section*, which models the link capacity restrictions. A delay-function model is used in the running section, whereas in the vertical queue section, an exit-flow function model is employed.

Other methods [8] require the resolution of fixed-point problems and the maximization of the problem of flow optimization in each node as a part of its analytical rules or interpolation method [43]–[46]. This could indicate *a priori* that the proposed approach is less time consuming than those developed in the literature and could be applicable to large-scale networks.

We assume that a link $a \in \mathcal{A}$ is modeled by a single segment with two sections, as shown in Fig. 2. The following relationships describe the variables of each section with the whole-link variables:

$$x_a(t) = x_a^r(t) + x_a^q(t) \tag{14}$$
$$\tau_a(t) = \tau_a^q\left((\tau_a^r(t)\right) \tag{15}$$
$$u_a(t) = u_a^r(t) \tag{16}$$
$$v_a(t) = v_a^q(t) \tag{17}$$
$$v_a^r(t) = u_a^q(t) \tag{18}$$

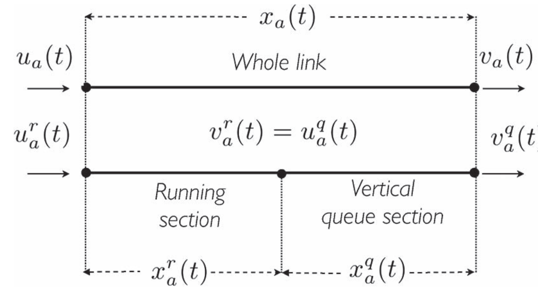

Fig. 2. Functional scheme of a link.

where the superscripts relate the meaning of the variable to the appropriate section.

1) *Running section.* This represents the traversal time of a vehicle in the whole link. In this paper, we assume a delay-function model. The most notable examples (static assignment traffic problem) are the so-called *Bureau of Public Roads* (BPR) *functions* (see [47]), i.e.,

$$f_a\left(x_a^r\right) = \alpha_a + \gamma_a \left(\frac{x_a^r}{\beta_a}\right)^{n_a} \tag{19}$$

where x_a^r is the link volume in the running section, α_a is the travel time in the free-flow link, β_a is the link capacity, and γ_a and n_a are parameters that should be calibrated (see [25] and [48]).

The function f_a defines the exit time as

$$\tau_a^r(t) = t + f_a\left(x_a^r(t)\right) \quad \forall\, t \in [0, T_\infty] \tag{20}$$

where T_∞ denotes the latest instant when all flow leaves the network, and $f_a(x_a^r(t))$ is the traversal time of link a at instant t.

2) *Vertical queue section.* This section of the link models the capacity restrictions related to the flow that should be reached for each link $a \in \mathcal{A}$ and for each instant t. The first constraint of the model is that the outflow is lower than the link capacity for all instants. Thus

$$v_a^q(t) \le \beta_a(t) \quad \forall\, t \in [0, T_\infty] \tag{21}$$

where $\beta_a(t)$ is a function that dynamically defines link capacity.

The users in the queue satisfy the FIFO rule. This implies that a user is served at the first instant after all earlier users in the queue have been served. The exit time from the queue of a user who arrives at instant t is denoted by $\tau_a^q(t)$ and satisfies the following equation:

$$x_a^q(t) = \int_t^{\tau_a^q(t)} v_a^q(\xi) d\xi. \tag{22}$$

It is worth noting that the proposed model generalizes both the delay-function model and the exit-flow function model. For example, the PQ model is obtained when $\gamma_a = 0$ and $\beta_a(t) = \beta_a$ are used. Moreover, if $\beta_a(t) = \infty$, it leads to the delay-function model.

Theorem 2.1—The Proposed Model is FIFO Rule Consistent: The model implied by expressions (14)–(22), together with the FIFO assumption of the running sections for all link a, guarantees satisfaction of the FIFO rule.

Proof: If all links in a path satisfy the FIFO condition, then the users that are on the same path also satisfy the FIFO condition. Note that, for a generic link a, the FIFO condition holds.

Let $t_1 \leq t_2$, where $t_1, t_2 \in [0, T_\infty]$. If and only if $\tau_a(t_1) \leq \tau_a(t_2)$. Using (15), the preceding relationship leads to

$$\tau_a^q\left(\tau_a^r(t_1)\right) \leq \tau_a^q\left(\tau_a^r(t_2)\right).$$

It is assumed that the FIFO condition holds in the running section, thus

$$\tau_a^r(t_1) \leq \tau_a^r(t_2).$$

Let $t_1^r = \tau_a^r(t_1)$ and $t_2^r = \tau_a^r(t_2)$. Using the relationship (10) and the nonnegativity of the inflow rate function $u_a^q(t)$, we obtain

$$\begin{aligned} x_a^q\left(t_2^r\right) &= x_a^q\left(t_1^r\right) + \int_{t_1^r}^{t_2^r} \left[u_a^q(\xi) - v_a^q(\xi)\right] d\xi \\ &\geq x_a^q\left(t_1^r\right) - \int_{t_1^r}^{t_2^r} v_a^q(\xi) d\xi. \end{aligned} \tag{23}$$

Using (22) and the linearity of the integral, we obtain

$$\begin{aligned} x_a^q\left(t_1^r\right) = \int_{t_1^r}^{\tau_a^q(t_1^r)} v_a^q(\xi) d\xi &= \int_{t_1^r}^{t_2^r} v_a^q(\xi) d\xi \\ &+ \int_{t_2^r}^{\tau_a^q(t_2^r)} v_a^q(\xi) d\xi + \int_{\tau_a^q(t_2^r)}^{\tau_a^q(t_1^r)} v_a^q(\xi) d\xi. \end{aligned} \tag{24}$$

Equation (24) can be rewritten as

$$\int_{\tau_a^q(t_2^r)}^{\tau_a^q(t_1^r)} v_a^q(\xi) d\xi = x_a^q\left(t_1^r\right) - x_a^q\left(t_2^r\right) - \int_{t_1^r}^{t_2^r} v_a^q(\xi) d\xi. \tag{25}$$

Equation (23) shows that the right side of (25) is negative. As $v_a^q(\xi) \geq 0$, we obtain $\tau_a^q(t_2^r) \geq \tau_a^q(t_1^r)$. □

Nevertheless, finding proper delay functions is challenging. For example, two decades after the work of [49], the linear delay function is still the only delay function of the form $f_a(x_a^r(t))$ that satisfies the FIFO condition under all inflow profiles. Additionally, recent studies have indicated that, probably, there is no FIFO-consistent delay function other than the linear one. In [50], it is shown that, for BPR type, the linear functions only satisfy the FIFO condition and, moreover, that piecewise linear functions do not satisfy the FIFO condition. Taking this into account, the linear delay-function model (i.e., $n_a = 1$ in BPR function) is assumed. This approach is presented in [51].

One way of improving the predictions of the proposed model increasing the computational cost would be that of, instead of applying the proposed travel-time model to the whole link, dividing the link into segments, applying the model (suitably adjusted) sequentially to these segments. Reference [52] proved that, as the discretization is refined, the solution converges to the solution of the Lighthill–Whitham–Richards model. In addition, the queue section can make close approximations of inhomogeneous links by treating the links as homogeneous, using the mean value as a capacity parameter.

D. Stochastic Demand

The traffic network, barring any incidents, is accurately known, whereas demand has a strong stochastic component to be considered. For this reason, we assume that the traffic demand intensity $d_\omega(t, \theta)$ for the O–D pair w at instant t depends on a random variable θ, which reflects the change in demand from day to day.

The flow intensity in path p is given by

$$h_p(t, \theta) = \mathbb{P}(p, t) d_\omega(t, \theta), \quad p \in \mathcal{P}_\omega \tag{26}$$

where $\mathbb{P}(p, t)$ is the probability that a user of pair ω chooses path $p \in \mathcal{P}_\omega$ at instant t. Note that $\sum_{p \in \mathcal{P}_\omega} \mathbb{P}(p, t) = 1$. In this paper, we assume that the function $\Pi(t) := (\ldots, \mathbb{P}(p, t), \ldots)$ and $D(t, \theta) := (\ldots, d_\omega(t, \theta), \ldots)$ are exogenous to the CDNL model and are known. The fact that $D(t, \theta)$ depends on a random variable means that flows and travel times on the paths of the network, which are denoted by $C(\Pi(t), \theta)$, are also considered random variables defined as the solution of the CDNL model. The aim in this paper is to efficiently compute the vector of random travel times $C(\Pi(t), \theta)$. It does not imply that the route choice model $\Pi(t)$ is independent of the level of congestion but is rather considered outside of the CDNL model. This matter is analyzed below.

DTA models integrate a traffic demand model, a supply model (the CDNL model), and a demand/supply interaction model. The demand model represents the decisions made by users on the traffic network, such as the choice of whether the journey is made, the destination, the starting time, or the path selected. Discrete choice demand models [53]–[55] can be used to explicitly define $\Pi(t)$ as a function of the vector of random travel times C on the paths of the network. Thus

$$\Pi(t) = G(t, C). \tag{27}$$

The whole DTA model is defined by finding a fixed point $\Pi(t)$ such that

$$\Pi(t) = G\left(t, C\left(\Pi(t), \theta\right)\right). \tag{28}$$

Equation (28) shows that the route choice model $\Pi(t)$ depends on the level of congestion C. The demand models will allow the extension of the CDNL model to the context of multimodal networks, for example, [56] used a nested logit distribution as a demand model, which explicitly takes into account the choice of mode of transport by users and transfer node among modal networks.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

III. Proposed Solution Procedure

The procedure proposed for solving the CDNL model is a hybrid of an analytical approach and a simulation-based approach. A discretization scheme is applied to analytical formulation of the CDNL model; and the analytical rules considering flow conservation, flow propagation, capacity, and boundary conditions lead to a discrete-packet approach. Mesosimulation is employed in [8], [10], [11], [57], and [58]. The main difference between the proposed approach and the previous work is that, instead of employing temporal discretization $[t, t+\Delta t]$, it is the flow Δx that is discretized.

The simulation algorithm discretizes the demand and obtains the so-called traffic *packets* traveling through the network. A packet is the set of vehicles leaving a centroid connector in a time interval (variable for each packet) and following the same path p. For simplicity, we assume that all packets have the same number of vehicles Δx (size of the packet).

In Section III-A, the rules that describe the link dynamic are summarized. In Section III-B, the mechanisms to synchronize all packets are described.

A. Discretization of the Link Dynamic Model

The basis of the mesosimulation procedure is the *packet*; the state of each packet i is defined by the pair of features (τ_i, n_i), where τ_i is the departure time for the current link, and n_i is the ordinal number of the current link in its path. In the beginning of the simulation, all packets are in the first link; this is $n_i = 1$, which represents the centroid connectors. Now, the calculation of the first value of τ_i is described.

Consider two types of link on the network: the centroid connectors and the road links. The former are used to make the demand load, and the latter are responsible for demand propagation within the network.

1) Dynamic of Centroid Connectors: We have two types of mesoscopic load computing components in centroid connectors: deterministic load, which assumes that the number of people who want to travel at every moment and the chosen path are known, and stochastic load, in which the existence of uncertainty in both choices is assumed.

a) Deterministic load in centroid connectors: Suppose as starting point the set of flow paths $\{h_p(t)\}_{p\in\mathcal{P}}$. In this case, θ is a known parameter. The associated packets are analyzed for each path p.

The path–link flow conservation equation (2) allows an inductive relationship to be obtained, which determines the load of packet i in the traffic network as a function of the previous packet $(i-1)$. We assume that the $(i-1)$th packet has been loaded at the instant τ_{i-1} of entry and that τ_{i-1} is less than T (the last time in the simulation). The next entry time for packet i satisfies the equation

$$\Delta x = \int_{\tau_{i-1}}^{\tau_i} h_p(\xi) d\xi. \tag{29}$$

If the solution holds $\tau_i > T$, the packet would leave the temporal simulation horizon and would be considered null, proceeding to load demand in the following path $p = p + 1$ by solving

$$\Delta x = \int_{0}^{\tau_i} h_p(\xi) d\xi. \tag{30}$$

b) Stochastic load in centroid connectors: The stochastic load is analogous to the deterministic case since the path associated with packet i is randomly chosen according to the probabilities $\mathbb{P}(p, \tau_i)$ with $p \in \mathcal{P}_w$.

In order to define more formally the numerical scheme employed, suppose that a realization of the random variable θ has been performed, and to do this, the demand ω has been analyzed, and the packet i has been generated. The following equation is solved:

$$\Delta x = \int_{\tau_{i-1}}^{\tau_i} d_\omega(\xi, \theta) d\xi \tag{31}$$

if $\tau_i > T$, a null packet is generated, and the following demand $\omega = \omega + 1$ is analyzed by solving

$$\Delta x = \int_{0}^{\tau_i} d_\omega(\xi, \theta) d\xi. \tag{32}$$

Then, path p followed by packet i should be randomly generated through the Monte Carlo simulation using the probabilities $\mathbb{P}(p, \tau_i)$ with $p \in \mathcal{P}_w$.

2) Dynamic of Road Links: Assume that packet i arrives at instant $\mathcal{T}$ at link a. The new value τ_i, in which the packet leaves the link, is computed as follows. It is necessary to consider the time to traverse the running section. Thus

$$\tau_i^r = \tau_a^r(\mathcal{T}) = \mathcal{T} + \alpha_a + \gamma_a \cdot x_a^r(\mathcal{T}). \tag{33}$$

As the model satisfies the FIFO condition, the instant in which the packet before i enters the vertical queue should be prior to $\mathcal{T}$, and therefore, it is assumed that the entry time is known and denoted by ϕ_a.

Equation (21) implies

$$\int_{t_1}^{t_2} v_a^q(\xi) d\xi \leq \int_{t_1}^{t_2} \beta_a(\xi) d\xi \quad \forall t_1, t_2 \in [0, T_\infty]. \tag{34}$$

The last equation can be discretized. Let ϕ_a be the last instant in which a packet leaves link a, then τ_i is the next instant in which packet i leaves the vertical queue according to the flow capacity restrictions. Equation (34) leads to

$$\Delta x = \int_{\max\{\tau_i^r, \phi_a\}}^{\tau_i} v_a^q(\xi) d\xi \leq \int_{\max\{\tau_i^r, \phi_a\}}^{\tau_i} \beta_a(\xi) d\xi. \tag{35}$$

The value τ_i is the least instant for which the previous relationship holds and thus should satisfy

$$\Delta x = \int_{\phi_a'}^{\tau_i} \beta_a(\xi) d\xi \tag{36}$$

where

$$\phi'_a = \max\{\tau_i^r, \phi_a\}. \tag{37}$$

It is worth noting that, in (35), the value of τ_i^r is previously computed using (33).

In the case of constant capacity $\beta_a(t) = \beta_a$ along the simulation period, (36) leads to

$$\tau_i = \phi'_a + \frac{\Delta x}{\beta_a}. \tag{38}$$

In the Appendix, the case in which $\beta_a(t)$ is a square wave has been solved. This type of capacity function allows for modeling signal-controlled intersections.

B. Discrete Event Algorithm for Moving Packets Within Traffic Network

The discretized CDNL model can be interpreted as a queueing network in which each link is an individual queue. The customers (packets) follow an itinerary defined by the path initially chosen. Each packet is served at each link using the dynamic mechanism previously described. When a packet is served at a link, it can join another link and queue to be served or leave the traffic network.

We consider the following function:

$$a := a(i, n) \tag{39}$$

which returns the link a that would be visited by the packet i in its nth queue. This function defines the customer routing in the queueing network.

We will denote an *event* as the situation in which a packet leaves its current link in a given time. The proposed algorithm processes all events in time order. Let $\mathcal{T}$ be the current instant of the simulation clock; the algorithm calculates the next event to be processed as the next packet that leaves the set of individual queues. This event is associated with a packet i' and through the function

$$\widehat{a} = a(i', n_{i'}) \tag{40}$$

where $n_{i'}$ is the number of links visited by packet i', and $\widehat{a}$ is the current link. The link queue $\widehat{a}$ is defined by the set $\mathcal{Q}_{\widehat{a}}$ of all packets that are currently on the link $\widehat{a}$. This means that, when i' leaves the queue, it should be updated. Thus, we can calculate the next link to visit as follows:

$$\widetilde{a} = a(i', 1 + n_{i'}) \tag{41}$$

and the queue $\mathcal{Q}_{\widetilde{a}}$ is updated.

Table I summarizes the discrete event simulation algorithm.

TABLE I
DISCRETE EVENT ALGORITHM

0. ***(Initialization).*** Let $[0, T]$ be the simulation period and Δx be the packet size. Initialize the simulation clock $\mathcal{T} = 0$ let $\phi_a = 0$ for all links $a \in \mathcal{A}$.
1. ***(Packet generation)***. Load the packets deterministically or stochastically in the centroid connectors during the simulation period. Locate the packet i in the first link of its path (centroid connector) $n_i = 1$. By using equations (29)–(32) compute the exit times τ_i. Update all queues $\mathcal{Q}_a$ where a is a centroid connector. Otherwise $\mathcal{Q}_a = \{\emptyset\}$ and $\Gamma_a := +\infty$. Compute the next event for the queue a by:
   $\Gamma_a := \text{minimize}_{i \in \mathcal{Q}_a}\{\tau_i\}$
2. ***(Processing of events***). While $\mathcal{T} \le T$
   2.1 ***(Update clock***). Let
   $\widehat{a} := \arg\ \text{minimize}_{a \in A}\{\Gamma_a\}$
   be the link where the next event is produced (exit of packet from link $\widehat{a}$) for analysis.
   Update clock $\mathcal{T} = \Gamma_{\widehat{a}}$.
   Let i' be the packet associated with the event being analysed in the link $\widehat{a}$, that is,
   $i' := \arg\ \text{minimize}_{i \in \mathcal{Q}_{\widehat{a}}}\{\tau_i\}$.
   and go to Step 2.3. If *false*
   2.2 ***(Simulation of queue*** $\mathcal{Q}_{\widetilde{a}}$*).*
   2.2.1 **If** the packet i' has finished its trip calculate the exit time $t_{i'} = \mathcal{T}$
   Take $\phi_{\widehat{a}} = \mathcal{T}$.
   Update queue $\mathcal{Q}_{\widehat{a}} = \mathcal{Q}_{\widehat{a}} - \{i'\}$.
   Update its next event.
   If $\mathcal{Q}_{\hat{a}} \neq \{\emptyset\}$ then compute
   $$\tau_i^r := \min_{j \in \mathcal{Q}_{\widehat{a}}}\{\tau_j^r\}.$$
   Using $\phi_{\widehat{a}}, \tau_i^r$ and equations (36)–(37) compute τ_i. Let $\Gamma_{\widehat{a}} = \tau_i$.
   Otherwise if $\mathcal{Q}_{\hat{a}} = \{\emptyset\}$ set $\Gamma_{\widehat{a}} = +\infty$.
   2.2.2 **Otherwise** let $n_{i'} = n_{i'} + 1$ and $\widetilde{a} = a(i', n_{i'})$. Compute
   $x_{\widetilde{a}}^r = \left|\{j \in \mathcal{Q}_{\widetilde{a}} : \tau_j^r \le \mathcal{T}\}\right| \cdot \Delta x$
   where $|\cdot|$ is the cardinal of a set. Compute
   $\tau_{i'}^r = \alpha_{\widetilde{a}} + \gamma_{\widetilde{a}} \cdot x_{\widetilde{a}}^r$.
   If $\mathcal{Q}_{\widetilde{a}} = \{\emptyset\}$ then compute $\tau_{i'}$ by using $\phi_{\widetilde{a}}, \tau_{i'}^r$ and equations (36)–(37); and let $\Gamma_{\widetilde{a}} = \tau_{i'}$.
   $\mathcal{Q}_{\widetilde{a}} = \mathcal{Q}_{\widetilde{a}} \cup \{i'\}$.

End

C. Monte Carlo Simulation

The Monte Carlo method is very flexible and allows any random variables θ to be addressed. Assume that $d_\omega(t, \theta)$ depends on a random variable θ and that the variables of interest are obtained by using the discrete event algorithm (travel times, flows, etc.) for each realization of the random variable. This process allows a random sample for any size of the variables of interest to be generated.

The main drawback of the Monte Carlo approach is that it is a very time-consuming task. Nevertheless, the possibility of using distributed computing strategies makes it a useful alternative.

IV. NUMERICAL EXPERIMENTS

Here, the proposed method is illustrated and evaluated. The proposed discrete event algorithm has been implemented in MATLAB.

The Nguyen–Dupuis [59] and the Sioux Falls networks have been used to test the model.

The Nguyen–Dupuis network consists of 13 nodes and four O–D pairs, as shown in Fig. 1. The running section is defined by the linear cost functions (33) with parameters $\gamma_a = 0$ and α_a included in Table II. In the queue section (36) and (37), the

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE II
PARAMETERS OF THE NGUYEN–DUPUIS NETWORK EXAMPLE

Link a	α_a (hours)	β_a (veh./hour)	Link a	α_a (hours)	β_a (veh./hour)
$(1,5)$	$7/60$	2500	$(1,12)$	$9/60$	2500
$(4,5)$	$9/60$	2500	$(4,9)$	$12/60$	2500
$(5,6)$	$3/60$	2500	$(5,9)$	$9/60$	2500
$(6,7)$	$5/60$	2500	$(6,10)$	$13/60$	2500
$(7,8)$	$5/60$	2500	$(7,11)$	$9/60$	2500
$(8,2)$	$9/60$	2500	$(9,10)$	$10/60$	2500
$(9,13)$	$9/60$	2500	$(10,11)$	$6/60$	2500
$(10,12)$	$9/60$	2500	$(11,3)$	$8/60$	2500
$(12,6)$	$7/60$	2500	$(12,8)$	$14/60$	2500
$(13,3)$	$11/60$	2500	$(13,1)$	0	50000
$(16,4)$	0	50000	$(2,14)$	0	50000
$(3,15)$	0	50000			

TABLE III
PARAMETERS OF THE O–D DEMAND INTENSITY FUNCTIONS

Pair	θ_w	μ_1	μ_2	σ_1	σ_2
w_1	4000	8	15	2	2
w_2	8000	9.5	15.5	1.5	1.5
w_3	6000	7.5	14.5	2	2
w_4	2000	8.5	16	1.5	1.5

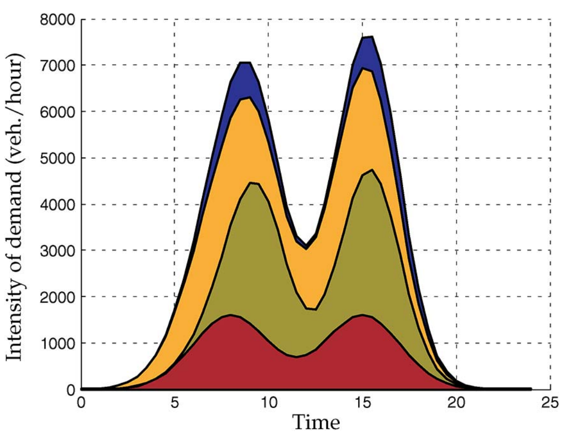

Fig. 3. O–D demand intensity functions for the Nguyen–Dupuis network.

capacity is considered constant $\beta_a(t) = \beta_a$ over time. These parameters are also shown in Table II.

The demand intensity is given by

$$d_\omega(t,\theta) := \sum_{i=1}^{2} \frac{\theta_\omega}{\sigma_i^w \sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-\mu_i^w}{\sigma_i^w}\right)^2} \tag{42}$$

where the values of the parameters are given in Table III.

In the experiments, the daily demand intensity presents two peak hours (around 8:00 and 15:00). This behavior is modeled as the sum of two Gaussian functions. The parameter μ_i is associated with the peak hour i, and the parameter σ_i defines its amplitude. The amount of users for the pair ω is proportional to θ_ω.

Fig. 3 shows the demand intensity for the four O–D pairs considered in the Nguyen–Dupuis network.

The Sioux Falls network consists of 76 nodes, 600 paths, and 528 O–D pairs; in this paper, we describe the Nguyen–Dupuis network example, and in order to save space, the Sioux Falls network data used are available at http://bit.ly/18KVDfM.

TABLE IV
TEST OF COMPUTATIONAL BURDEN FOR SIMULATION OF NGUYEN–DUPUIS AND SIOUX FALLS NETWORK EXAMPLES

	Nguyen-Dupuis	
Packet size (Δx)	Number of packets	Simulation time (seconds)
10	6807	2,9
5	13617	6,15
1	68093	68,64
	Sioux Falls	
Packet size (Δx)	Number of packets	Simulation time (seconds)
10	12221	1,89
5	25032	4,56
1	126196	43,51

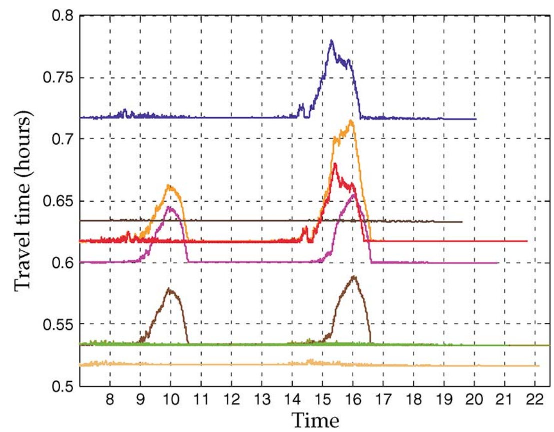

Fig. 4. Nguyen–Dupuis network: travel times on the different paths of the illustrative example.

A. Computational Burden

The test problem has been solved on an Intel core 2 Quad Q9550 (2.83 GHz) CPU with 4 GB of random access memory with three different sizes of packet for both networks over a simulation period ranging from 00:00 to 23:00 h. Simulation results are shown in Table IV.

The computational cost depends on the number of packets analyzed on the network. Taking into account that the reduction in the packet size causes an increase in the simulation time, a nonlinear relationship between the simulation time and the packet size is evident. The size of the network influences computational cost linearly. The results show a high performance for the algorithm mainly due to the FIFO behavior of the model, which means not having to reorder queues once the packet has changed from one queue to another.

It is worth noting that despite thousands of packets being analyzed, the computational cost is on the order of seconds.

B. Example of Deterministic Demand

This section shows the simulation results with the Nguyen–Dupuis network. The goal of this section is to test the internal consistency of the model.

Fig. 4 shows the travel time for all the paths in the network. Each color corresponds to a specific path. It is possible to see that travel times in each path increase at different moments; it shows also an increase in congestion between 09:30–10:00 and 15:00–16:00; therefore, greater travel times are obtained during peak hours. Note that the maximum intensity of the demand is

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

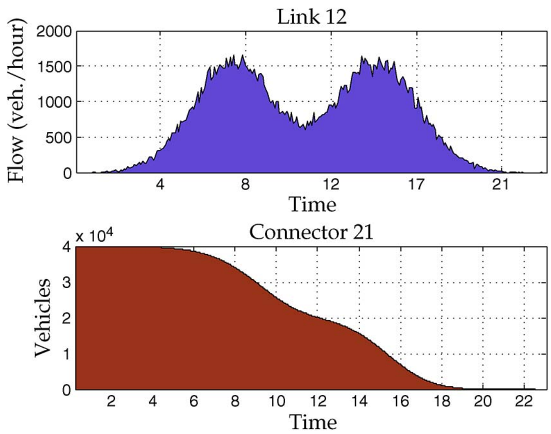

Fig. 5. Nguyen–Dupuis network: dynamically varying flows over time in two links.

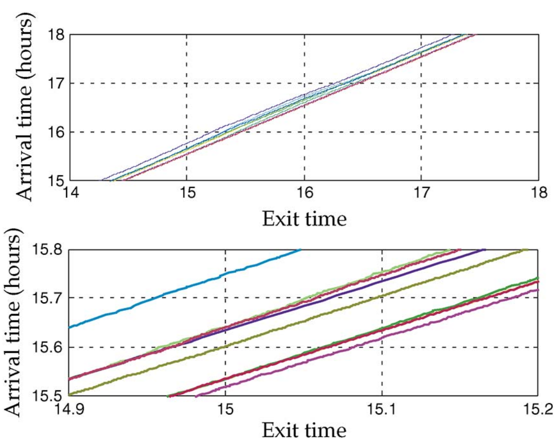

Fig. 6. Nguyen–Dupuis network: arrival times for different paths versus departure time.

reached at times 9:00 and 15:30, but the maximum congestion is delayed half hour.

It is important to note that the network behavior is similar to the PQ model applied to each link separately. The reason is that travel times are affected by variations in demand only if the network capacity is exceeded, i.e., the network experiences congestion.

A second output of the model shows the link loads as a function of the time. Fig. 5 shows the link flows and how they vary dynamically over time, resulting in a zigzag effect as a consequence of the discretization of the problem (packet size). Note that the number of packets in connector link 21 decreases during the simulation.

Fig. 6 shows the arrival times for the different paths, depending on the instant at which the trip is initiated. We can observe a monotonically increasing behavior in all paths, which ensures the FIFO condition and is one of the main contributions of this model.

C. Example of Stochastic Demand

Here, we have considered that θ, which represents the total demand, is a multivariate normal variable $N(\bar{\theta}, \Sigma)$, i.e., with the following probability density function:

$$f(\theta) = (2\pi)^{-n/2}|\Sigma|^{-1/2} \exp\left\{-1/2(\theta - \bar{\theta})^T \Sigma^{-1}(\theta - \bar{\theta})\right\}$$

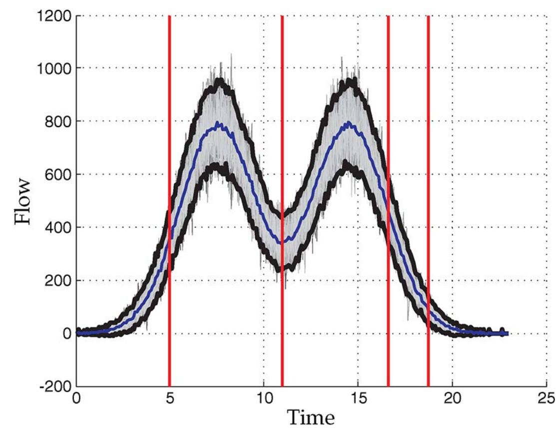

Fig. 7. Results of Monte Carlo simulation for the flow at link $a = 3$.

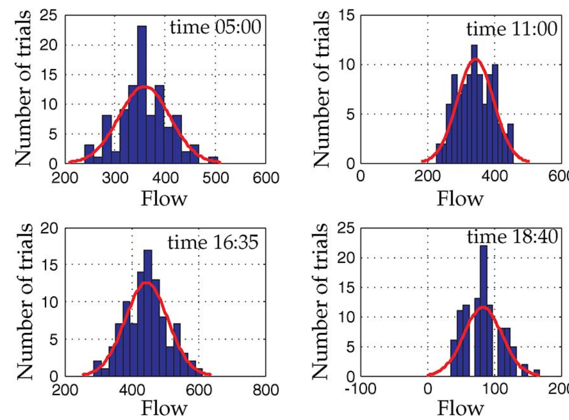

Fig. 8. Empirical distribution of the flow at link $a = 3$ for four different instants.

where $\bar{\theta}$ is the $n = 4$-dimensional vector of means given in Table III. Σ is the following variance–covariance matrix:

$$\Sigma := \begin{pmatrix} 10\,000 & 2500 & 0 & 0 \\ 2500 & 10\,000 & 0 & 0 \\ 0 & 0 & 10\,000 & 2500 \\ 0 & 0 & 2500 & 10\,000 \end{pmatrix}.$$

$|\Sigma|$ is the determinant of Σ, and θ^T denotes the transpose of θ.

In order to illustrate the stochastic demand model, we have employed the Monte Carlo simulation method to obtain a random sample of incoming traffic demand in the Nguyen–Dupuis network. This sample of size 100 has been randomly generated and simulated by the discrete event algorithm given in Table I. Fig. 7 shows the dynamic evolution of the flows in the link $a = 3$. Gray lines represent individual simulations, and black lines represent the mean plus ± 1.96 times the sample standard deviation. Note that the output of the model provides not only mean values (blue line) of predictions but the corresponding variabilities as well. In fact, they provide empirical probability density functions. To illustrate this fact, four temporal instants are considered and are depicted by the red vertical lines in Fig. 7. In addition, a histogram of the data and an adjustment of a normal probability density function are made. The results are shown in Fig. 8. It is shown to be a good fit for the normal distribution with different means and variances in each instant.

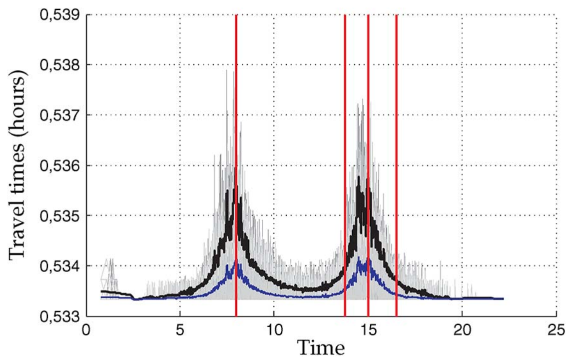

Fig. 9. Results of Monte Carlo simulation for travel times on path $p = 1$.

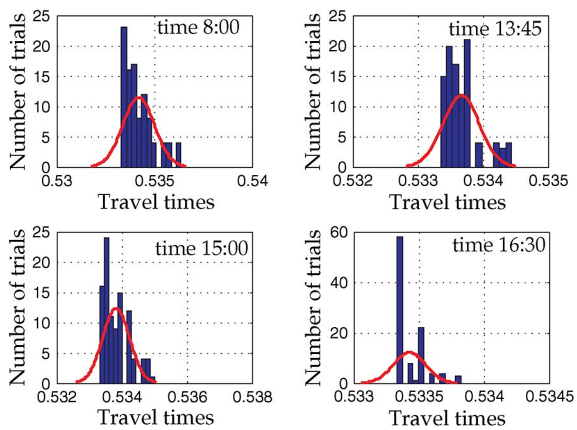

Fig. 10. Empirical distribution of the travel times at path $p = 1$ for four different instants.

Fig. 9 shows the travel times in path 1 corresponding to links (20, 2, 18, 11, 22, 30) for 100 simulations; the blue line represents mean values, and the vertical lines indicate temporal instants as in the previous example. The figure shows that travel times are bounded below by the free-flow travel time. Gray lines represent individual simulations, and the black line represent the mean plus 1.96 times the sample standard deviation.

Fig. 10 shows the variation of the travel times on the time instants (red lines) through the 100 simulations. In this case, it is possible to observe a poor fit to a normal distribution. Note that this variability in travel times can be used as input for route and departure time choice models, which represent risk-averse behavior.

D. Distributed Computing

Finally, a case study focused on the assessment of the distributed computing of the Monte Carlo simulation has been carried out. In this experiment, a set of 100 random simulations was executed in a parallel architecture with 12 cores (AMD Dual Opteron 6 Core 4226 2.7 GHz), comparing the performance of the nonparallel version with respect to the executions using from 2 to 12 cores. Fig. 11 depicts the speedup ratio (CPU time with one core/CPU time with n cores) of each simulation for each network. It can be seen that the two graphs have similar growth until they use six or more cores. Therefore, the maximum speedup ratio reached is approximately 7 for the Sioux Falls network and 5 for the Nguyen–Dupuis network. The difference between the behavior of the two graphs is related to the amount of access to memory required for each network. This example highlights the fact that the memory is the bottleneck of the procedure, stating that, for some networks, other distributed computing techniques such as grid computing could be more convenient.

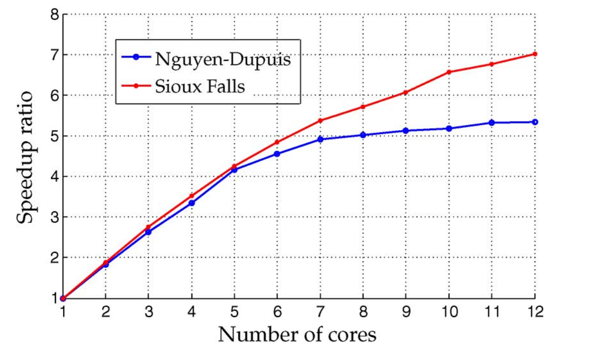

Fig. 11. Performance analysis of parallel computing using from 2 to 12 cores.

V. Conclusion

The main contributions of this paper are given as follows.

1) A discrete event algorithm is proposed for solving a mesoscopic CDNL model. Compared with the proposals presented in the literature, we have opted for discretizations of the packet size Δx instead of time discretizations. The numerical tests show a high performance for the algorithm.
2) The Monte Carlo simulation approach and the distributed computational strategies allow the stochastic CDNL model for general random variables to be tackled. This kind of approach is necessary for risk-averse traffic modeling.
3) The goal of the approaches developed for the CDNL model is a tradeoff between precision and computational cost. Our approach is based on a macroscopic whole-link model, which presents the following characteristics: i) it generalizes both the PQ model and the delay-function model; ii) it is compatible with the FIFO condition in the case of linear delay-function models; and iii) it explicitly reflects time-dependent link capacities. This last property is key in the evaluation of interventions in real time systems. The FIFO condition is used to speed up the discrete event simulation procedure.
4) The simulation code is very versatile; it permits online observations of the state of traffic to be incorporated easily and can be used for working with discontinuous intensities and capacities. An example given in this paper is the use of link capacity by means of square waves. The corresponding formulas have been developed to simulate this case, which is a difficult task for analytical methods. From previous experience, the proposed approach is easily integrable as a component of flow propagation in a DTA model with a wide range of applications, for example, in advanced traveller information systems. Dynamic signal optimization and ramp metering are other possible topics that the model extension described can study, in the areas of capacity management and speed regulation.

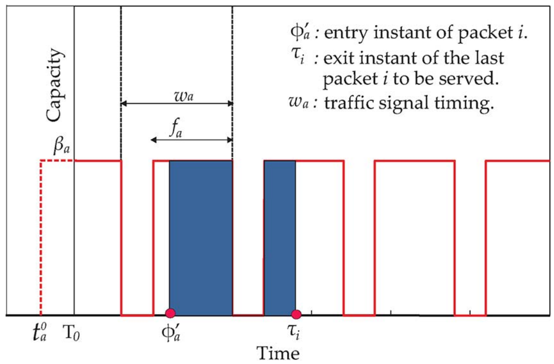

Fig. 12. Link capacity function $\beta_a(t)$ given by a square wave.

Appendix
Dynamic Capacity Given by a Square Wave

Here, we will analyze a link capacity $\beta_a(t)$ given by a square wave, as shown in Fig. 12. We introduce the following notation.

- w_a Wave amplitude associated with the square wave $\beta_a(t)$.
- f_a Period of time in which the link capacity has the maximum value.
- T_0 Initial instant of the simulation.
- t_a^0 Instant lesser than the initial instant T_0 in which the square wave begins its maximum capacity (see Fig. 12).

The solution of (36) yields the value of τ_i and coincides with the moment in which the shaded area below the wave is exactly Δx. This value is given by the expression

$$\tau_i := \begin{cases} \phi'_a + \frac{\Delta x}{\beta_a}, & \text{if } \Delta x \leq (z_a - \phi'_a)\,\beta_a \\ (N_a + M_a + 1)w_a + s_a f_a - t_a^0, & \text{otherwise,} \end{cases} \tag{43}$$

where

$$M_a := \left[\frac{\phi'_a - T_0 + t_a^0}{w_a}\right] \tag{44}$$

$$z_a := T_0 - t_a^0 + M_a w_a + f_a \tag{45}$$

$$N_a := \left[\frac{\Delta x - (z_a - \phi'_a)\beta_a}{\beta_a f_a}\right] \tag{46}$$

$$s_a := \frac{\Delta x - (z_a - \phi'_a)\,\beta_a}{\beta_a f_a} - N_a \tag{47}$$

and $[\cdot]$ is the integer part of a number. Note that, in this paper, it is assumed that the initial instant of the simulation is $T_0 = 0$.

References

[1] D. Merchant and G. L. Nemhauser, "A model and an algorithm for the dynamic traffic assignment problems," *Transp. Sci.*, vol. 12, no. 3, pp. 1028–1031, Aug. 1978.

[2] C. F. Daganzo, "The cell transmission model I: A dynamic representation of highway traffic consistent with the hydrodynamic theory," *Transp. Res. B, Methodol.*, vol. 28, no. 4, pp. 269–288, Aug. 1995.

[3] M. Ben-Akiva, M. Bierlaire, H. Koutsopoulos, and R. Mishalani, "DynaMIT: A simulation-based system for traffic prediction," presented at the DACCORD Short-Term Forecasting Workshop, Delft, The Netherlands, 2006.

[4] S. Hoogendoorn and P. Bovy, "State-of-the-art of vehicular traffic flow modelling," *Proc. Inst. Mech. Eng. I, J. Syst. Control Eng.*, vol. 215, no. 4, pp. 283–303, Jun. 2001.

[5] S. Panwai and H. Dia, "Comparative evaluation of microscopic car-following behavior," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 3, pp. 314–325, Sep. 2005.

[6] E. Brockfeld and P. Wagner, "Validating microscopic traffic flow models," in *Proc. IEEE ITSC*, 2006, pp. 1604–1608.

[7] H. Celikoglu, "Reconstructing freeway travel times with a simplified network flow model alternating the adopted fundamental diagram," *Eur. J. Oper. Res.*, vol. 228, no. 2, pp. 457–466, Jul. 2013.

[8] H. Celikoglu, E. Gedizlioglu, and M. Dell'Orco, "A node-based modeling approach for the continuous dynamic network loading problem," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 165–174, Mar. 2009.

[9] H. Celikoglu and M. Dell'Orco, "A dynamic model for acceleration behaviour description in congested traffic," in *Proc. IEEE ITSC*, 2008, pp. 986–991.

[10] H. Celikoglu and M. Dell'Orco, "Mesoscopic simulation of a dynamic link loading process," *Transp. Res. C, Emerging Technol.*, vol. 15, no. 5, pp. 329–344, Oct. 2007.

[11] M. Dell'Orco, "A dynamic network loading model for mesosimulation in transportation systems," *Eur. J. Oper. Res.*, vol. 175, no. 3, pp. 1447–1454, Dec. 2006.

[12] W. S. Vickrey, "Congestion theory and transport investment," *Amer. Econ. Rev.*, vol. 59, no. 2, pp. 251–260, May 1969.

[13] X. Nie and H. M. Zhang, "A comparative study of some macroscopic link models used in dynamic traffic assignment," *Netw. Spatial Econ.*, vol. 5, no. 1, pp. 89–115, Mar. 2005.

[14] V. Adamo, V. Astarita, M. Florian, M. Mahut, and J. H. Wu, "Modelling the spill-back of congestion in link based dynamic network loading models: A simulation model with application," in *Proc. 14th Int. Symp. Transp. Traffic Theory*, A. Ceder, Ed., 1999, pp. 555–573.

[15] M. Di Gangi, "Modeling evacuation of a transport system: Application of a multimodal mesoscopic dynamic traffic assignment model," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1157–1166, Dec. 2011.

[16] M. Gangi and P. Velonà, "Multimodal mesoscopic approach in modeling pedestrian evacuation," *Transp. Res. Rec.*, no. 2090, pp. 51–58, 2009.

[17] M. Di Gangi, P. Velonà, and A. Catanzariti, "Safety of users in road evacuation: Some enhancement in modelling pedestrian evacuation of a building," in *Proc. WIT Trans. Built Environ.*, 2008, vol. 101, pp. 739–749.

[18] T. Friesz, T. Kim, C. Kwon, and M. Rigdon, "Approximate network loading and dual-time-scale dynamic user equilibrium," *Transp. Res. B, Methodol.*, vol. 45, no. 1, pp. 176–207, Jan. 2011.

[19] S. Lee and D. Fambro, "Application of subset autoregressive integrated moving average model for short-term freeway traffic volume forecasting," *Transp. Res. Rec.*, no. 1678, pp. 179–188, 1999.

[20] M.-C. Tan, S. Wong, J.-M. Xu, Z.-R. Guan, and P. Zhang, "An aggregation approach to short-term traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 60–69, Jan. 2009.

[21] C. Zhang, S. Sun, and G. Yu, "A Bayesian network approach to time series forecasting of short-term traffic flows," in *Proc. 7th IEEE Conf. Intell. Transp. Syst.*, 2004, pp. 216–221.

[22] S. Sun, C. Zhang, and G. Yu, "A Bayesian network approach to traffic flow forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 124–132, Mar. 2006.

[23] E. Castillo, J. Menéndez, and S. Sánchez-Cambronero, "Predicting traffic flow using Bayesian networks," *Transp. Res. B, Methodol.*, vol. 42, no. 5, pp. 482–509, Jun. 2008.

[24] E. Castillo, M. Nogal, J. Menéndez, S. Sánchez-Cambronero, and P. Jiménez, "Stochastic demand dynamic traffic models using generalized beta-Gaussian Bayesian networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 565–581, Jun. 2012.

[25] R. García-Ródenas and D. Verastegui-Rayo, "Adjustment of the link travel-time functions in traffic equilibrium assignment models," *Transportmetrica*, vol. 9, no. 9, pp. 798–824, Oct. 2013. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/18128602.2012.669415

[26] M. Montanino, B. Ciuffo, and V. Punzo, "Calibration of microscopic traffic flow models against time-series data," in *Proc. 15th IEEE ITSC*, Sep. 16–19, 2012, pp. 108–114.

[27] C. Antoniou, R. Balakrishna, H. Koutsopoulos, and M. Ben-Akiva, "Calibration methods for simulation-based dynamic traffic assignment systems," *Int. J. Model. Simul.*, vol. 31, no. 3, pp. 227–233, 2011.

[28] B. Ciuffo, V. Punzo, and V. Torrieri, "Comparison of simulation-based and model-based calibrations of traffic-flow microsimulation models," *Transp. Res. Rec.*, no. 2088, pp. 36–44, 2008.

[29] R. Balakrishna, C. Antoniou, M. Ben-Akiva, H. Koutsopoulos, and Y. Wen, "Calibration of microscopic traffic simulation models: Methods and application," *Transp. Res. Rec.*, no. 1999, pp. 198–207, 1999.

[30] J. Wu, M. Brackstone, and M. McDonald, "The validation of a microscopic simulation model: A methodological case study," *Transp. Res. C, Emerging Technol.*, vol. 11, no. 6, pp. 463–479, Dec. 2003.
[31] Y. Nie, H. Zhang, and D.-H. Lee, "Models and algorithms for the traffic assignment problem with link capacity constraints," *Transp. Res. B, Methodol.*, vol. 38, no. 4, pp. 285–312, May 2004.
[32] J. L. Espinosa-Aranda and R. García-Ródenas, "A discrete event-based simulation model for real-time traffic management in railways," *J. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 94–107, Apr. 2012.
[33] M. Almodóvar and R. García-Ródenas, "On-line reschedule optimization for passenger railways in case of emergencies," *Comput. Oper. Res.*, vol. 40, no. 3, pp. 725–736, Mar. 2013.
[34] Z. Liu and Q. Meng, "Distributed computing approaches for large-scale probit-based stochastic user equilibrium problems," *J. Adv. Transp.*, vol. 47, no. 6, pp. 553–571, Oct. 2013.
[35] E. Huang, C. Antoniou, J. Lopes, Y. Wen, and M. Ben-Akiva, "Accelerated on-line calibration of dynamic traffic assignment using distributed stochastic gradient approximation," in *Proc. IEEE ITSC*, 2010, pp. 1166–1171.
[36] H. Shao, W. H. K. Lam, and M. L. Tam, "A reliability-based stochastic traffic assignment model for network with multiple user classes under uncertainty in demand," *Netw. Spatial Econ.*, vol. 6, no. 3/4, pp. 173–204, Sep. 2006.
[37] H. K. Lo, X. W. Luo, and B. W. Y. Siu, "Degradable transport network: Travel time budget of travelers with heterogeneous risk aversion," *Transp. Res. B, Methodol.*, vol. 40, no. 9, pp. 792–806, Nov. 2006.
[38] C. F. Daganzo, "Properties of link travel time functions under dynamic loads," *Transp. Res. B, Methodol.*, vol. 29, no. 2, pp. 95–98, Apr. 1995.
[39] O. Drissi-Kaietouni and A. Hameda-Benchekroun, "Dynamic traffic assignment model and a solution algorithm," *Transp. Sci.*, vol. 26, no. 2, pp. 119–128, May 1992.
[40] D. Gangi, M. Astarita, and V. Adamo, "A dynamic network loading model for simulation of queue effects and while-trip re-routing," in *Proc. 24th Eur. Transp. Forum PTRC*, Uxbridge, U.K., Sep. 2–6, 1996, pp. 1–11.
[41] M. Kuwahara and T. Akamatsu, "Decomposition of the reactive dynamic assignments with queues for a many-to-many origin–destination pattern," *Transp. Res. B, Methodol.*, vol. 31, no. 1, pp. 1–10, Feb. 1997.
[42] J. Li, O. Fujiwara, and S. Kawakami, "A reactive dynamic user equilibrium model in network with queues," *Transp. Res. B, Methodol.*, vol. 34, no. 8, pp. 605–624, Nov. 2000.
[43] E. Castillo, J. M. Menéndez, M. Nogal, P. Jiménez, and S. Sánchez-Cambronero, "A FIFO rule consistent model for the continuous dynamic network loading problem," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 264–283, Mar. 2012.
[44] J. H. Wu, Y. Chen, and M. Florian, "The continuous dynamic network loading problem: A mathematical formulation and solution method," *Transp. Res. B, Methodol.*, vol. 32, no. 3, pp. 173–187, Apr. 1998.
[45] Y. W. Xu, J. H. Wu, M. Florian, P. Marcotte, and D. L. Zhu, "Advances in the continuous dynamic network loading problem," *Transp. Sci.*, vol. 33, no. 4, pp. 341–353, Apr. 1999.
[46] J. M. Rubio-Ardanaz, J. H. Wu, and M. Florian, "Two improved numerical algorithms for the continuous dynamic network loading problem," *Transp. Res. B, Methodol.*, vol. 37, no. 2, pp. 171–190, Feb. 2003.
[47] *Highway Capacity Manual (HCM)*, TRB, National Research Council, Washington, DC, USA, 1965.
[48] F. Russo and A. Vitetta, "Reverse assignment: Calibrating link cost functions and updating demand from traffic counts and time measurements," *Inverse Prob. Sci. Eng.*, vol. 19, no. 7, pp. 921–950, Oct. 2005.
[49] T. Friesz, D. Bernstein, T. E. Smith, R. L. Tobin, and B. W. Wei, "A variational inequality formulation of the dynamic network user equilibrium problem," *Oper. Res.*, vol. 41, no. 1, pp. 179–191, Jan./Feb. 1993.
[50] R. García-Ródenas, M. L. López-García, A. Niño-Arbelaez, and D. Verastegui-Rayo, "A continuous whole-link travel time model with occupancy constraint," *Eur. J. Oper. Res.*, vol. 175, no. 3, pp. 1455–1471, Dec. 2006.
[51] V. Astarita, "A continuous time link model for dynamic network loading based on travel time function," in *Proc. 13th Int. Symp. Transp. Traffic Theory*, Lyon, France, 1996, pp. 79–102.
[52] M. Carey and Y. Ge, "Convergence of a discretised travel-time model," *Transp. Sci.*, vol. 39, no. 1, pp. 25–38, Feb. 2005.
[53] D. McFadden, "Conditional logit analysis of qualitative choice behavior," in *Frontiers in Econometrics*, P. Zarembka, Ed. New York, NY, USA: Academic, 1974, pp. 105–142.
[54] M. Ben-Akiva and S. Lerman, *Discrete Choice Analysis: Theory and Applications to Travel Demand*. Cambridge, MA, USA: MIT Press, 1995.
[55] E. Cascetta, *Transportation Systems Engineering: Theory and Methods*. Norwell, MA, USA: Kluwer, 2001.
[56] R. García and A. Marín, "Network equilibrium with combined modes: Models and solution algorithms," *Transp. Res. B, Methodol.*, vol. 39, no. 3, pp. 223–254, Mar. 2005.
[57] D. R. Leonard, P. Gower, and N. B. Taylor, "Contram: Structure of the model," TRL, Wokingham, U.K., TRRL Res. Rep., 1989, vol. 1.
[58] R. Jayakrishnan, H. S. Mahmassani, and T. Y. Hu, "An evaluation tool for advanced information and management systems in urban networks," *Transp. Res. C, Emerging Technol.*, vol. 2, no. 3, pp. 129–147, Sep. 1994.
[59] S. Nguyen and C. Dupuis, "An efficient method for computing traffic equilibria in networks with asymmetric transportation costs," *Transp. Sci.*, vol. 18, no. 2, pp. 185–202, May 1984.

María Teresa Sánchez-Rico received the B.Sc. degree in computer science from Instituto Tecnológico de Morelia, Morelia, México, in 2008, and the M.Sc. degree from the University of Castilla-La Mancha, Ciudad Real, Spain, in 2012. She is currently working toward the Ph.D. degree in the Models and Algorithms for Transportation group at the University of Castilla-La Mancha.

Her research interests include operational research applied to transportation systems and artificial intelligence.

Ricardo García-Ródenas received the B.Sc. degree in mathematics from the University of Valencia, Valencia, Spain, in 1991 and the Ph.D. degree from Universidad Politécnica de Madrid, Madrid, Spain, in 2001.

Since 2008, he has been an Assistant Professor with the Department of Mathematics, University of Castilla-La Mancha, Ciudad Real, Spain. His current research interests include operational research applied to transportation systems and artificial intelligence.

José Luis Espinosa-Aranda received the B.Sc. degree in computer engineering and the M.S. degree in computer science from the University of Castilla-La Mancha (UCLM), Ciudad Real, Spain, in 2009 and 2010, respectively. He is currently working toward the Ph.D. degree in the MAT group at UCLM.

His current research interests include operational research applied to transportation systems and artificial intelligence.