Emmendorfer, L. R; Retamoso, M.;Costa, A. C. R.; Werhli, A. V.

IV MCSUL

Southern Conference on

Computational Modeling

IX ERMAC

Encontro regional de Matemática

Aplicada e Computacional

Universidade Federal do Rio Grande, Rio Grande - RS

Anais do IV Simpósio de Modelagem Computacional do Sul – 3MCSul /Emmendorfer, L. R.; Retamoso, M.; Costa, A. C. R.; Werhli, A. V. (Org) - Rio Grande: Universidade Federal do Rio Grande 2010.

402p. :il.
ISSN 2179-0671

1. Modelagem Física e Matemática. 2. Modelagem de Fluidos Geofísicos. 3. Fenômenos de Transporte e Termodinâmica4. Computação Científica e Modelagem Física e Matemática. 5. Sistemas Robóticos e Autônomos. I. Emmendorfer, L. R II. Retamoso, M.; III. Costa, A. C. R.;IV. Werhli, A. V.

CDD

Computational Experience with a State-dependent Traffic Assignment Problem

Felipe F. Cardoso, Oriane Magela Neto
Programa de Pós-Graduação em Engenharia Elétrica (PPGEE),
Universidade Federal de Minas Gerais,
31270-901 - Belo Horizonte - MG, Brazil
Email: warriorssjufmg@ufmg.br, oriane@dee.ufmg.br

Frederico R. B. Cruz
Departamento de Estatística,
Universidade Federal de Minas Gerais,
31270-901 - Belo Horizonte - MG, Brazil
Email: fcruz@est.ufmg.br

***Abstract*—In this paper, we examine the System Optimum (SO) problem. The SO formulation is equivalent to a situation in which users cooperate with each other in order to minimize the overall travel cost. Usually, the travel costs are expressed in terms of times and are typically given from classical formulas. In this paper we aim to investigate an $M/G/c/c$ state-dependent queueing network based formula, which is is not convex but S-shaped. As a consequence multiple solutions may be present for the SO, which justifies the use heuristic procedures such as a Differential Evolution algorithm. Computational results are present to show the efficacy and efficiency of the approach.**

I. Introduction

In the classical Wardrop System Optimum (SO) assignment model, the users are assumed to cooperate with each other in order to minimize the overall travel costs [1]. Even though the SO is based on a rather non-realistic behavioral assumption, we argue that its solution may be seen as a result of a well-succeeded control action on the transportation network. In other words, signal timings may be re-optimized and alternative routes may be re-defined in response to an increase in demand causing extra green light timings, as it is known that traffic lights and routing can improve the flow [2], depending on the traffic densities (e.g., Dynamic Routing Information Panel Systems - DRIPS).

One major problem is that the overall travel costs, usually expressed in term of time [3], are based on deterministic travel times on the single links, yet these times are known to be rather variable between trips, within and between days etc. Usually, only the mean travel times are represented in the SO model and an important issue in urban transportation networks is to model congestion, especially during rush hours, when the demand exceeds capacity by far [4]. As seen in Figure 1, which presents results from many empirical studies for North American roads [5]–[9], congestion may be perceived as a decrease in the mean speed when the vehicular density increases, resulting in the well-known speed-flow density curves (see the seminal work by Greenshields [7]).

There have been successful attempts in the literature to model how users select their routes in a congested network (for instance see Helbing et al. [10], and references therein). Under the assumption that drivers have perfect knowledge about travel costs, they will choose the best route according to Wardrop's first principle, which is equivalent to the mixed-strategy Nash equilibrium of n-players, non-cooperative game [11]. This leads to the Deterministic User Equilibrium (UE), another classical traffic assignment model [1]. In the equilibrium, routes carrying a positive flow will have equal travel costs. Unfortunately the conclusion is that, by doing so, scarce resources (street and road capacity) are used in an inefficient way [10]. For this reason, the focus here will be only in the SO assignment model.

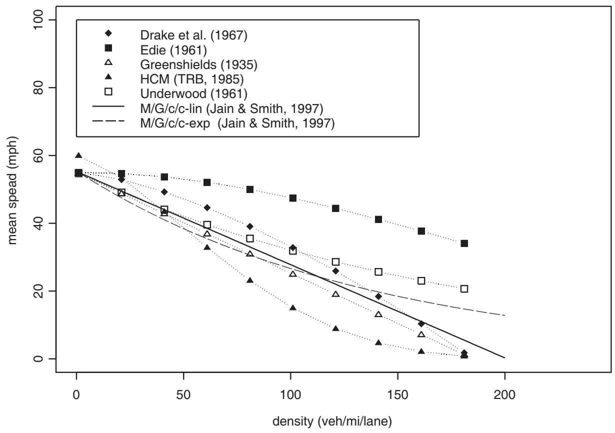

Fig. 1. Empirical Distributions for Vehicular Traffic Flows [5]–[9] and $M/G/c/c$ State-dependent Models [12]

In this paper, we discuss an extension of the SO model by applying the state-dependent $M/G/c/c$ queueing network model in order to estimate the travel times, usually the main factor for route selection [3]. The $M/G/c/c$ state-dependent model has been used in the past [12]. In fact, Cruz et al. [13] have shown recently that the $M/G/c/c$ state-dependent model is also quite effective for modeling the travel time on single links, in comparison to well-established formulas, because of its capability of representing traffic congestion, as it is seen in Figure 1. Now, we provide detailed and explicit description of the state-dependent $M/G/c/c$ queueing network model.

A single road link may be seen as c parallel servers to its occupants and c is also the total number of users allowed in a system. Consequently, there is no buffer or waiting space. Second, based on the empirical results presented in Figure 1, the service time for the occupants depends on the number of users currently in the system. As a consequence, an $M/G/c/c$ state-dependent queueing model seems to be a reasonable tool to describe a single link [12]. The limiting probabilities for the random number of entities N in an $M/G/c/c$ queueing model, $p_n \equiv \Pr[N = n]$, will be:

$$p_n = \left\{ \frac{\left[\lambda E[T_1]\right]^n}{n! f(n) f(n-1) \cdots f(2) f(1)} \right\} p_0, \qquad (1)$$

where $n = 1, 2, \ldots, c$. The empty system probability, p_0, is given by:

$$p_0^{-1} = 1 + \sum_{i=1}^{c} \left\{ \frac{\left[\lambda E[T_1]\right]^i}{i! f(i) f(i-1) \cdots f(2) f(1)} \right\}, \qquad (2)$$

where λ is the arrival rate and $E[T_1] = l/V_1$ is the expected service time of a lone vehicle in the traffic space of length l, considering that V_1 is the speed of a lone vehicle. The capacity, c, is given by:

$$c = \lfloor klw \rfloor,$$

where l is the length, w is the width (in number of lanes), k is the capacity of the link per length-unit per lane, and $\lfloor x \rfloor$ is the largest integer not superior to x. Additionally, $f(n) = V_n/V_1$ is the service rate, that is, the ratio of the average speed of n users in the link to that of a lone occupant, V_1. In Section III, we will detail how to adjust the model to vehicular traffic and to extend it to *networks* of road links.

Also discussed in this paper is a successful way to solve the SO model with a state-dependent travel time. Because the resulting SO is intrinsically non-convex (due to the shape of $M/G/c/c$ travel time functions), we decided not to use the FrankWolfe based optimization approaches, but selected a flexible heuristic. In this paper, we use a Differential Evolution (DE) based heuristic. The optimization quality of DE is independent of the shape of the objective functions and proves to be an efficient and acceptable solution for similar non-linear and non-convex problems [14].

In Section II the mathematical programming formulation for the classical SO assignment model is presented in detail. Section III describes the algorithm for solving the SO, as well as the performance evaluation algorithm. Section IV focuses on the computational experiments. Finally, Section V summarizes the paper and discusses open questions for future research in the area.

II. Mathematical Programming Formulations

Well-known from the literature, the SO formulation is briefly reviewed as follows. The network notation used is summarized in Table I.

The SO formulation is equivalent to a situation in which users cooperate with each other in order to minimize the total travel costs. According to Wardrop's second principle, the SO model is formulated as follows.

TABLE I
Basic Network Notation

Variable	Description
$\mathcal{N}$	node (index) set
$\mathcal{A}$	arc (index) set
$\mathcal{R}$	set of origin nodes; $\mathcal{R} \subseteq \mathcal{N}$
$\mathcal{S}$	set of destination nodes; $\mathcal{S} \subseteq \mathcal{N}$
$\mathcal{K}_{rs}$	set of paths connecting origin-destination (*O-D*) pair *r-s*; $r \in \mathcal{R}$, $s \in \mathcal{S}$;
x_a	flow on arc a; $\mathbf{x} = (\ldots, x_a, \ldots)$
$c_a(x_a)$	travel time on arc a; $\mathbf{c}(\mathbf{x}) = (\ldots, c_a(x_a), \ldots)$
f_k^{rs}	flow on path k connecting *O-D* pair *r-s*; $\mathbf{f}^{rs} = (\ldots, f_k^{rs}, \ldots)$; $\mathbf{f} = (\ldots, \mathbf{f}^{rs}, \ldots)$
c_k^{rs}	travel time on path k connecting *O-D* pair *r-s*; $\mathbf{c}^{rs} = (\ldots, c_k^{rs}, \ldots)$; $\mathbf{c} = (\ldots, \mathbf{c}^{rs}, \ldots)$
q^{rs}	trip rate between origin r and destination s; $(\mathbf{q})^{rs} = q^{rs}$
$\delta_{a,k}^{rs}$	indicator variable: $\delta_{a,k}^{rs} = \begin{cases} 1, & \text{if link } a \text{ is on path } k \text{ between } O\text{-}D \text{ pair } r\text{-}s, \\ 0, & \text{otherwise;} \end{cases}$ $(\mathbf{\Delta}^{rs})_{a,k} = \delta_{a,k}^{rs}$; $\mathbf{\Delta} = (\ldots, \mathbf{\Delta}^{rs}, \ldots)$

SO:

$$\min z(\mathbf{x}) = \sum_a x_a c_a(x_a),$$

s.t.:

$$\begin{aligned} \sum_k f_k^{rs} &= q^{rs}, && \forall\, r, s, \\ x_a = \sum_r \sum_s \sum_k f_k^{rs} \delta_k^{rs}&, && \forall\, a, \\ f_k^{rs} &\geq 0, && \forall\, k, r, s, \end{aligned}$$

where x_a is the flow on link a, $c_a(x_a)$ is the travel cost on link a, f_k^{rs} is the flow on route k between origin r and destination s, and q^{rs} is the demand between r and s. The complete notation is seen in Table I.

Notice that the travel costs, $c_a(x_a)$, are usually expressed in terms of times [3] and are usually given from classical formulas (BPR-like [15]) that have been constructed over the past 40 years for this purpose of describing the travel time, among some of which are the contributions by many authors (e.g., [16]–[18]). However, as one shall see shortly, none of these formulas is able to represent congestion as the $M/G/c/c$ state-dependent queueing model does.

The optimum solution is reached when the marginal travel costs on each path carrying a positive flow are equal, that is:

$$\begin{aligned} f_k^{rs}(g_k^{rs} - g^{rs*}) &= 0, && \forall\, r, s, \\ g_k^{rs} - g^{rs*} &\geq 0, && \forall\, r, s, \\ \sum_k f_k^{rs} &= q^{rs}, && \forall\, r, s, \\ f_k^{rs} &\geq 0, && \forall\, k, r, s, \end{aligned}$$

in which g_k^{rs} is the marginal cost on route k and g^{rs*} is the optimal marginal cost, both between r and s.

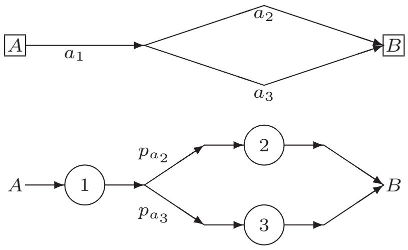

Fig. 2. Three-road Network and Corresponding $M/G/c/c$ Model

III. Algorithms

Researchers have solved the SO and similar models by means of many algorithms. Results have been reported with exact algorithms (dual [19], parallel [20], and Lagrangian based algorithms [21]) and heuristics [4], [13]. As mentioned earlier, we will use the DE because the state-dependent travel time formula is non-convex. Basically, what we have to do is to find routing probabilities for alternative routes to go from A to B (e.g., p_{a_2} and p_{a_3}, as seen in Figure 2), in such a way that the objective function $\sum_a x_a c_a(x_a)$ is minimized. The steps are detailed as follows.

A. Computation of the Travel Times

Single Queues

As described earlier in Section I, an $M/G/c/c$ state-dependent queueing model for a *single* link is straightforward. Considering vehicular related applications and realizing that k represents the jam density parameter (veh/mi-lane), normally k ranges from 185-265 veh/mi-lane. The aim is to derive a congestion model that represents the effect depicted in Figure 1, i.e., in which the service rate depends on the number of users in the system. The following exponential model seems quite reasonable:

$$f(n) = \exp\left[-\left(\frac{n-1}{\beta}\right)^{\gamma}\right],$$

with

$$\gamma = \log\left[\frac{\log(V_a/V_1)}{\log(V_b/V_1)}\right] / \log\left(\frac{a-1}{b-1}\right),$$

and

$$\beta = \frac{a-1}{[\log(V_1/V_a)]^{1/\gamma}} = \frac{b-1}{[\log(V_1/V_b)]^{1/\gamma}}.$$

The values a and b are arbitrary points used to adjust the exponential curve. In vehicular related applications, commonly used values are $a = 20lw$ and $b = 140lw$ corresponding to densities of 20 and 140 veh/mi-lane respectively. Looking at the curves presented in Figure 1, reasonable values for such points are $V_a = 48$ mph and $V_b = 20$ mph.

From (1), important performance measures can be derived:

$$\begin{cases} p_c & = & \Pr[N = c], \\ \theta & = & \lambda(1 - p_c), \\ L & = & E[N] = \sum_{n=1}^{c} n p_n, \\ W & = & E[T] = L/\theta, \end{cases}$$

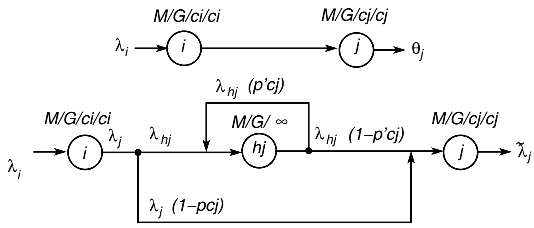

Fig. 3. The Generalized Expansion Method (GEM)

where p_c is the blocking probability, $\theta \equiv x_a$ is the throughput in veh/h, L is the expected number of customers in the link (also known as work-in-process, WIP), and $W \equiv c_a(x_a)$ is the expected service time in hours (here derived from Little's formula).

Networks of Queues - Series Queues

The modeling issue is only half solved because deriving performance measures for *networks* of $M/G/c/c$ state-dependent queues is a task considerably more complex. Indeed, the routing probabilities define the input in each queue. Besides, inter-blocking effects now may be present. The Generalized Expansion Method (GEM) was a method proposed by Kerbache and Smith [22] back in the 1980 decade and has a long tradition in the area. The GEM is a combination of repeated trials and node-by-node decomposition approximation methods, with a key characteristic that an artificial holding node is added preceding each finite queue in the network in order to register blocked customer that attempt to enter the finite node when it is at capacity, see Figure 3, which shows an example of two road links configured in series and presents the corresponding queueing network. By the addition of holding nodes, the queueing network is 'expanded' into an equivalent Jackson network, in which each node can then be decomposed and analyzed separately. Now, we shall describe briefly the GEM for series queues, which consists of the three following stages, performed for each finite node in the original queueing network.

1) Stage 1 - Network Reconfiguration: For each node with finite capacity, an artificial node is added directly preceding it, as shown in Figure 3. Customers that are impeded to move to the forwarding node (because such a node is at capacity) are re-routed to the artificial node. The probability that an arriving customer is blocked by node j equals p_{c_j}. Thus, with probability $(1-p_{c_j})$, it will enter node j, and with probability p_{c_j} it will enter the holding node (h_j). The holding node is modeled as an $M/G/\infty$ queue, so that there will be no waiting to enter this node.

After service at the holding node, the customer will be blocked again with a new probability, p'_{c_j}. With probability $(1-p'_{c_j})$, it will proceed to the following node. Otherwise, it must retrace its path through the feedback loop into artificial node h_j again.

2) Stage 2 - Parameter Estimation: The value of p_{c_j} can be determined from known analytical results. For $M/G/c/c$ state dependent queues, such a value is given directly by (1),

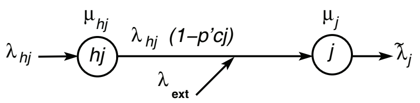

Fig. 4. External Arrival Rate λ_{ext}

i.e., $p_{c_j} = \Pr\{N = c_j\}$. The value of p'_{c_j} is determined from approximation results. After a customer completes its service at holding node h_j, it is forced to return with probability p'_{c_j}, for another immediate service delay. An approximation that uses diffusion techniques states that [23]:

$$p'_{c_j} = \left\{ \frac{\mu_j + \mu_{h_j}}{\mu_{h_j}} - \frac{\lambda\left[(r_2^{c_j} - r_1^{c_j}) - (r_2^{c_j - 1} - r_1^{c_j - 1})\right]}{\mu_{h_j}\left[(r_2^{c_j+1} - r_1^{c_j+1}) - (r_2^{c_j} - r_1^{c_j})\right]} \right\}^{-1}, \quad (3)$$

where r_1 and r_2 are the roots to:

$$\lambda_{\text{ext}} - (\lambda_{\text{ext}} + \mu_{h_j} + \mu_j)x + \mu_{h_j}x^2 = 0. \quad (4)$$

Defined with help of Figure 4, the external arrival rate λ_{ext}, used in (4), is:

$$\left\{ \begin{array}{lcl} \lambda_{\text{ext}} & = & \tilde{\lambda}_j - \lambda_{h_j}(1 - p'_{c_j}), \\ \tilde{\lambda}_j & = & \lambda_j(1 - p_{c_j}), \\ \lambda_{h_j} & = & \lambda_j(p_{c_j}), \\ \lambda_j & = & \lambda_i(1 - p_{c_i}) = \tilde{\lambda}_i. \end{array} \right. \quad (5)$$

Using renewal theory, it can be shown that the service rate of the holding node is (in the exponential case) as follows [24]:

$$\mu_{h_j} = \frac{2\mu_j}{1 + \sigma_j^2 \mu_j^2}, \quad (6)$$

where σ_j^2 is the service time variance. However, since the service rate is state dependent, a reasonable assumption is to consider the worst case:

$$\mu_{h_j} = \mu_j \approx \frac{c_j}{E[T_1]/f(c_j)}, \quad (7)$$

where c_j is the maximum number of servers in parallel and $E[T_1]/f(c_j)$ is the service time for c_j occupants.

3) Stage 3 - Feedback Elimination and Update: A reconfiguration of the holding node is performed to remove the strong dependencies in arrival processes caused by the repeated visits (feedback) to the artificial node. The feedback arc is removed from the holding node by recomputing the service rate at this node as follows:

$$\mu'_h = (1 - p'_{c_j})\mu_{h_j}. \quad (8)$$

Finally, the average service time that a customer spends at node i preceding node j is given by:

$$\tilde{\mu}_i^{-1} = \mu_i^{-1} + p_{c_j}(\mu'_h)^{-1}. \quad (9)$$

Equation (9) represents the final step of the GEM, which ultimate goal is to provide an approximation scheme to update the service rates of upstream nodes that takes into account all blocking after service caused by downstream nodes.

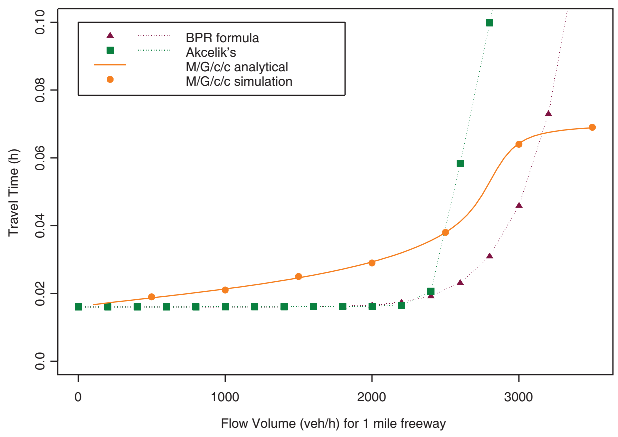

Fig. 5. One-mile Vehicular Traffic Flows [13]

Networks of Queues - Merge and Split Queues

Similarly, the process can be extended to merge and split networks. Details will not be given here but can be found easily in the literature [12]. Important to stress is that we do not physically modify the networks. The holding nodes are only artificial nodes included in the performance evaluation software.

Final Remarks

In conclusion, for a given routing probability vector, $\mathbf{p}$, it should not be hard to estimate the corresponding performance measure, $\sum_a x_a c_a(x_a)$, which is the objective function to be minimized. The problem is that the use of state-dependent travel times complicates the optimization problem. In fact, as seen in Figure 5, typical travel time functions (BPR-like) differ quite much from $M/G/c/c$ state-dependent queueing model functions, specially under heavy traffic. Under low traffic, the queueing approach is close to the classical formula but the $M/G/c/c$ model predicts S-shaped travel-time curves, which represent serious trouble for any optimization algorithm.

B. *Differential Evolution Algorithm*

A DE algorithm will be used to compute the optimal routing probability vector, $\mathbf{p} = (\ldots, p_a, \ldots)$, $\forall a \in \mathcal{A}$. DE algorithms are part of a broader family of Genetic Algorithms (GA). Some of the features of DE algorithms that justify their use for solving SO problems include fastness, robustness, ease of use, and ability to operate on flat surfaces. Additionally, it was found that DE algorithms were the best evolutionary computation method after the study of seven difficult design and control MINLP problems in chemical engineering [14].

A DE algorithm is defined as a parallel direct search method which operates on a population P_G of constant size that is associated with each generation G and consists of NP vectors, or candidate solutions, $\mathbf{X}_{p,G}$, $p = 1, 2, \ldots, NP$. Each vector $\mathbf{X}_{p,G}$ consists of D decision variables $X_{o,p,G}$, $o = 1, 2, \ldots, D$. This is briefly summarized as:

TABLE II

SETTINGS FOR THE THREE-ROAD NETWORK

Route	Length*	Width†	$V_1^\ddagger$	$V_a^\ddagger$	$V_b^\ddagger$	c (veh)	$E[T_1]^\sharp$
a_1	0.80 (0.50)	5	25 (40)	23 (37)	10 (16)	800	0.0320 (115)
a_2	2.50 (1.55)	2	20 (32)	18 (29)	6 (10)	1,000	0.1250 (450)
a_3	1.85 (1.15)	2	20 (32)	18 (29)	6 (10)	740	0.0925 (333)

Remarks: *in miles (km); †in # lanes; ‡in mph (km/h); ♯in h (s);

$$
\begin{aligned}
P_G &= \{\mathbf{X}_{1,G}, \mathbf{X}_{2,G}, \ldots, \mathbf{X}_{p,G}, \ldots, \mathbf{X}_{NP,G}\}, \\
\mathbf{X}_{p,G} &= \{X_{1,p,G}, X_{2,p,G}, \ldots, X_{o,p,G}, \ldots, X_{D,p,G}\}, \\
G &= 1, \ldots, G_{\max}, \\
NP &\geq 4.
\end{aligned}
$$

Each routing probability is then considered as the decision variable, $X_{o,p,G} \equiv p_a$. The DE algorithm is composed by the following steps:

1) Choose a strategy (Price and Storn [25] present 10 different strategies);
2) Initialize the crossover constant CR, the population size NP, the mutation scaling factor F, the coefficient of combination K, and the maximum number of generations $G_{\max}$;
3) Set the initial population $P_{G=0}$;
4) Evaluate the profit of each vector and find the one with the highest profit;
5) Perform mutation and recombination; DE mutates an object vector by adding the weighted difference of randomly sampled pairs of vectors in the current population P_G; the crossover operation creates a trial vector $\mathbf{U}_{p,G+1}$ by selecting elements from the target vector $\mathbf{X}_{p,G}$ and the mutated donor vector $\mathbf{V}_{p,G+1}$; the crossover constant CR controls the probability that a trial vector parameter will come from the mutated vector $\mathbf{V}_{p,G+1}$, instead of from the current vector $\mathbf{X}_{p,G}$, and therefore ranges from 0 to 1;
6) Check lower and upper bounds of the variables;
7) Perform selection;
8) Repeat the evolutionary cycle until $G_{\max}$ is reached.

More details on DE algorithms may be found in literature [25], including mutation schemes, values for the control parameters, other constraint handling methods, and stopping criteria.

IV. COMPUTATIONAL EXPERIMENTS

All algorithms described earlier were coded in C++ and are available from the authors for research and educational purposes. The experiments were conducted on a common PC, under Windows Vista operating system.

The example illustrated in Figure 2 is a three-link network. A and B are connected by link a_1 and two alternative links, a_2 and a_3, where one of the alternative routes is longer (and consequently slower) than the other. The adjustments of the respective $M/G/c/c$ model is presented in Table II. The algorithm was run for different arrival rates (λ). The results obtained may be seen in Table III.

TABLE III

OPTIMAL ASSIGNMENTS FOR THE THREE-ROAD NETWORK

λ	route	assignment	$E[T]^*$
0	a_1-a_2	n/a	0.1570 (565)
	a_1-a_3	n/a	0.1245 (448)
500	a_1-a_2	152	0.1591 (573)
	a_1-a_3	348	0.1287 (463)
1,000	a_1-a_2	370	0.1635 (588)
	a_1-a_3	630	0.1341 (483)
2,000	a_1-a_2	890	0.1791 (645)
	a_1-a_3	1,110	0.1484 (534)
4,000	a_1-a_2	1,496	0.4742 (1,707)
	a_1-a_3	**1,225**	**0.8964 (3,227)**
8,000	a_1-a_2	1,507	0.4750 (1,710)
	a_1-a_3	**1,224**	**0.8970 (3,229)**

Remark: *in hours (in seconds);

When the arrival rate is zero, we have that the expected travel time is the travel time of a lone occupant, which is the sum of the lone occupant expected travel times of the respective links (i.e., $0.1570 = c_{a_1}(0) + c_{a_2}(0) = 0.0320 + 0.1250$ and $0.1245 = c_{a_1}(0) + c_{a_3}(0) = 0.0320 + 0.0925$). From $\lambda = 500$, we observe an increase on the expected travel time caused by the system congestion level. Notice that the expected service times for both routes are never equal meaning that users with knowledge about the travel costs could reduce their own travel time by changing from a slow to a fast route. Such improvement would be impossible from the UE problem optimum solution (see Sheffi [1]) but the problem solved here is the SO model, which seeks the overall minimum travel time and not individual minimum travel times.

Concerning the optimum assignment, we remark that the traffic is mostly directed to the fastest route (i.e., with the lowest expected travel time) and then to the slowest route. This is what it should be expected, which is encouraging. For this network, we also observe that up to the arrival rate of 2,000 users per time unit, all the traffic goes through the network without any blocking (i.e., roughly the sum of the assignments equals the arrival rate). However, from this point on the network seems to have reached its capacity because only a fraction of the additional traffic can successfully go through.

Finally we would like to notice a somewhat unexpected behavior at link a_1-a_3, that is, an increase on its travel time (from 0.8964 to 0.8970) in spite of a reduction of the traffic on it (from 1,225 to 1,224). This is a type of behavior that would be impossible under BPR-like travel time estimation formulas but it is perfectly reasonable under the use of $M/G/c/c$ state-dependent models. In fact, as reported by many researchers, the $M/G/c/c$ state-dependent models induce a throughput-*vs.*-arrival-rate curve that reaches a maximum after which it decreases before it finally stabilizes, as seen in Figure 6.

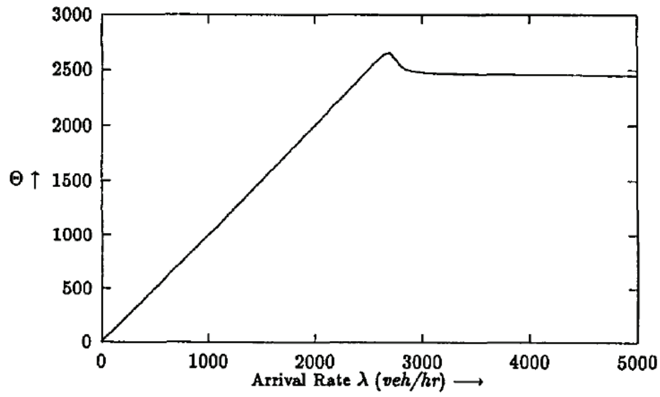

Fig. 6. Exponential Congestion [12]

V. Concluding Remarks

The SO problem was solved under a different stochastic travel time formula based on an $M/G/c/c$ state-dependent queues. This new formula clearly has advantages over the previously used formulas as it is in close agreement with the reality by modeling congestion effects (i.e., the travel time reduces when the link congestion level increases). On the other hand, the resulting travel time function is S-shaped, which brings difficulties to the optimization algorithms as now multiple optima may be present. Computational results attest that DE heuristics may be quite effective in solving the SO problem, as sound solutions were found. Additionally, the solutions seemed to be robust as it was demonstrated by the sensitivity analysis presented here.

Some possible future directions for this research include the analysis of more general networks and the application of the algorithms to modeling pedestrian networks, since that many of the similar features of the travel delay function as shown in this paper apply to pedestrian flows.

Acknowledgment

This research has been partially funded by CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico; grants 201046/1994-6, 301809/1996-8, 307702/2004-9, 472066/2004-8, 304944/2007-6, 561259/2008-9, 553019/2009-0, 550207/2010-4, 501532/2010-2, 303388/2010-2), by CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior; grant BEX-0522/07-4), and by FAPEMIG (grants CEX-289/98, CEX-855/98, TEC-875/07, CEX-PPM-00401/08, and CEX-PPM-00390-10).

References

[1] Y. Sheffi, *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1985.

[2] J. Poli Jr. and L. H. A. Monteiro, "Improving vehicle flow with traffic lights," *Advances in Complex Systems*, vol. 8, no. 1, pp. 59–63, 2005.

[3] J. N. Prashker and S. Bekhor, "Some observations on stochastic user equilibrium and system optimum of traffic assignment," *Transportation Research Part B*, vol. 34, pp. 277–291, 2000.

[4] H. Ceylan and M. G. H. Bell, "Genetic algorithm solution for the stochastic equilibrium transportation networks under congestion," *Transportation Research Part B*, vol. 39, pp. 169–185, 2005.

[5] J. S. Drake, J. L. Schofer, and A. D. May, "A statistical analysis of speed density hypotheses," *Highway Research Record*, vol. 154, pp. 53–87, 1967.

[6] L. C. Edie, "Car following and steady-state theory," *Operations Research*, vol. 9, pp. 66–76, 1961.

[7] B. D. Greenshields, "A study of traffic capacity," *Highway Research Board Proceedings*, vol. 14, pp. 448–477, 1935.

[8] Transportation Research Board, "Highway capacity manual," National Research Council, Special Report 209, 1985.

[9] R. T. Underwood, "Speed, volume, and density relationships: Quality and theory of traffic flow," *Yale Bureau of Highway Traffic*, pp. 141–188, 1961.

[10] D. Helbing, M. Schönhof, H.-U. Stark, and J. A. Holyst, "How individuals learn to take turns: Emergence of alternating cooperation in a congestion game and the prisoner's dilemma," *Advances in Complex Systems*, vol. 8, no. 1, pp. 87–116, 2005.

[11] M. G. H. Bell and C. Cassir, "Risk-averse user equilibrium traffic assignment: An application of game theory," *Transportation Research Part B*, vol. 36, no. 8, pp. 671–681, 2002.

[12] R. Jain and J. M. Smith, "Modeling vehicular traffic flow using $M/G/C/C$ state dependent queueing models," *Transportation Science*, vol. 31, no. 4, pp. 324–336, 1997.

[13] F. R. B. Cruz, T. van Woensel, J. M. Smith, and K. Lieckens, "On the system optimum of traffic assignment in $M/G/c/c$ state-dependent queueing networks," *European Journal of Operational Research*, vol. 201, no. 1, pp. 183–193, 2010.

[14] B. V. Babu and K. K. N. Sastry, "Estimation of heat transfer parameters in a trickle bed reactor using differential evolution and orthogonal collocation," *Computers & Chemical Engineering*, vol. 23, no. 3, pp. 327–339, 1999.

[15] Bureau of Public Roads, "Traffic assignment manual," U.S. Department of Commerce, Tech. Rep., June 1964.

[16] K. B. Davidson, "The theoretical basis of a flow - Travel time relationship for use in transportation planning," *Australian Road Research*, vol. 8, no. 1, pp. 32–35, 1978.

[17] R. Akçelik, "Travel time function for transport planning purposes: Davidson's function, its time dependent form and an alternative travel time function," *Australian Road Research*, vol. 21, no. 3, pp. 49–59, 1991.

[18] R. G. Dowling, R. Singh, and W. W. K. Cheng, "The accuracy and performance of improved speed-flow curves," *Road and Transport Research*, vol. 7, no. 2, pp. 159–174, 1998.

[19] D. W. Hearn and S. Lawphongpanich, "A dual ascent algorithm for traffic assignment problems," *Transportation Research Part B*, vol. 24, no. 6, pp. 423–430, 1990.

[20] J. K. Ho, "Solving the dynamic traffic assignment problem on a hypercube multicomputer," *Transportation Research Part B*, vol. 24, no. 6, pp. 443–451, 1990.

[21] T. Larsson and M. Patriksson, "An augmented Lagrangean dual algorithm for link capacity side constrained traffic assignment problems," *Transportation Research Part B*, vol. 29, no. 6, pp. 433–455, 1995.

[22] L. Kerbache and J. M. Smith, "The generalized expansion method for open finite queueing networks," *European Journal of Operational Research*, vol. 32, pp. 448–461, 1987.

[23] J. Labetoulle and G. Pujolle, "Isolation method in a network of queues," *IEEE Transactions on Software Engineering*, vol. SE-6, no. 4, pp. 373–381, 1980.

[24] L. Kleinrock, *Queueing Systems*. New York, NY, USA: John Wiley & Sons, 1975, vol. I: Theory.

[25] K. Price and R. Storn, "Differential evolution," International Computer Science Institute, University of California, Berkeley, Website of DE as on August 2006, 2006. [Online]. Available: http://www.icsi.berkeley.edu/˜storn/code.html