

Calibrating a Traffic Flow Model with Parallel Differential Evolution

G.A. Strofylas*, K.N. Porfyri*, I.K. Nikolos⁺, A.I. Delis*, and M. Papageorgiou*

^{*,+}Technical University of Crete, School of Production Engineering and Management,
University Campus, GR-73100, Chania, Greece

*email: gstrofylas@isc.tuc.gr, kporfyri@dssl.tuc.gr, adelis@science.tuc.gr, markos@dssl.tuc.gr

⁺email: jnikolo@dpem.tuc.gr, Tel.: +30 28210 37300, Contact Author

Keywords: parallel differential evolution, surrogate models, artificial neural networks, macroscopic traffic flow modeling

ABSTRACT

Given the importance of the credibility and validity required in macroscopic traffic flow models while performing real-world simulations, the necessity of employing an efficient, computationally fast, and reliable constrained optimization scheme for model calibration appears to be mandatory to ensure that the traffic flow characteristics are accurately represented by such models. To this end, a parallel, metamodel-assisted Differential Evolution (DE) algorithm is employed for the calibration of the second-order macroscopic gas-kinetic traffic flow (GKT) model using real traffic data from Attiki Odos freeway in Athens, Greece. The parallelization of the DE algorithm is performed using the Message Passing Interface (MPI), while artificial neural networks (ANNs) are used as surrogate models. Numerical simulations are performed, which demonstrate that the DE algorithm can be effectively used for the search of the globally optimal model parameters in the GKT model; in fact, the method appears to be promising for the calibration of other similar traffic models as well.

Keywords: parallel differential evolution, surrogate models, artificial neural networks, macroscopic traffic flow modeling.

1 Introduction

The need for robust and realistic modeling tools for evaluating different traffic systems and intelligent transportation system (ITS) technologies has increased remarkably over the last decades. Traffic flow models can be employed for the planning and assessment of road infrastructures, traffic surveillance and monitoring, incident detection, as well as for the development and testing of traffic control strategies and other operational tools. Currently, several commercial traffic simulation models are available, and even more mathematical models have been developed by researchers all over the world. Clearly, the accurate and effective application of traffic flow models is of significant importance. In this context, an effective calibration and validation process is deemed mandatory for any simulation model, so as to ensure its credibility and validity in performing real-world simulations, by reflecting realistically all possible traffic conditions.

In general, traffic simulation models can be classified as either microscopic or macroscopic, according to their level of detail. In a microscopic simulation, the traffic flow is modeled at a high level of detail, by capturing the behavior of individual vehicles; while at the macroscopic level, the traffic flow is described in lesser detail by using aggregated variables, such as flow, density and mean speed [1]. In fact, the feature of macroscopic models to require a relatively small number of parameters, compared to microscopic ones, results in a less demanding and computationally less expensive calibration and validation process and, therefore, in a more flexible development and implementation of such models for real-world applications.

Despite the increasing spread in the use of macroscopic traffic flow simulation models, relatively little attention has been given to their calibration and validation. Compared to microscopic models, surprisingly few studies include methods for addressing or actually conducting the calibration and validation process for macroscopic models. Examples of optimization algorithms used are the deterministic Complex algorithm of Box in [2, 3, 4, 5, 6], the deterministic Nelder-Mead algorithm utilized in [7, 8], genetic algorithms in [9, 10], the stochastic cross-entropy method employed in [11] and the parallel Differential Evolution algorithm in [12].

To contribute to the state-of-the-art, this work puts forward an optimization scheme, based on a parallel, metamodel-assisted Differential Evolution (DE) algorithm [27, 28] to determine the optimal parameters of the second-order macroscopic gas-kinetic traffic flow (GKT) model [22, 23, 24], which minimizes the relative error between the model prediction and the observed real data. Such a calibration process is a quite complex problem, since it takes the form of minimizing a cost function with numerous local minima, which traditional gradient-based algorithms are likely to fail to avoid.

Among the various search and optimization techniques, Evolutionary Algorithms (EAs) have emerged during the past decades as an essential and versatile tool for dealing with demanding high-dimensional real-world optimization problems. EAs are capable of handling non-differentiable, nonlinear and multimodal cost functions based on the principles of natural selection and evolution. From a population of candidate solutions, each individual is evaluated on the basis of its fitness function, and the best ones are selected (with accordingly increased probability) to proceed to the next generation and evolve farther. However, despite the important contribution of EAs in solving complicated problems, they suffer from a significant drawback; a considerable number of evaluations are needed, which usually calls for significantly increased computational resources. In order to overcome this barrier, the use of surrogate models (metamodels), in conjunction with parallel processing, appears to be an efficient approach.

The population-based searching mechanism of EAs makes them eminently suitable for parallelism. The use of parallel EAs (PEAs) not only improves the efficiency, but also enhances the arithmetic performance, if structured populations are adopted. The different types of parallel EAs can be divided mainly in two categories, panmictic EAs and structured ones [13, 14]. Combinations of the aforementioned categories have been reported in the literature resulting in hybrid models such as hierarchical hybrids [15, 16, 17]. In panmictic EAs a global parallelization model is usually followed, which utilizes a unique population of candidate solutions, while the selection operation is applied to all members of the population. The whole procedure is tailored by a central processor, which distributes the members of the population to different processors in order to be evaluated in parallel, while the selection step is performed only by the central processor sequentially. This model is usually combined with a Master-Slave

architecture. Synchronous and asynchronous parallel implementations of panmictic PEAs have been proposed in the literature [18, 19].

In general, parallel EAs mostly use structured populations, following either the island (dEAs) or the cellular model (cEAs) [14, 20, 21]. For the parallel implementation, the population is divided into subpopulations called demes, and each evolves separately on its assigned processor. In dEAs each subpopulation is compromised by many individuals; while in cEAs the subpopulation corresponds to a single individual. The demes exchange individuals with some migration frequency to ensure the propagation of good solutions. In a dEA the subpopulations are loosely connected to each other, whereas in cEAs every individual can interact only with its neighbours. When implementing the island model, only a small number of subpopulations are used, contrary to cEA, where the number of subpopulations is equal to the population size [13, 14]. Hybrid PEAs implementations utilize the multi-population dEA model with the fine-grained cEA model resulting in better convergence behavior [15, 16, 17].

In this work, a parallel DE algorithm was developed based on the panmictic approach, using a unique population that is distributed among the processors, with a Master-Slave architecture. Separate executable programs perform the evaluation of each individual in the population, while the required data exchange and communication between the processors are achieved using MPI (Message Passing Interface) library functions. Furthermore, the utilization of surrogate models within the DE algorithm enhances its performance by substituting the computationally time-consuming exact evaluations of the fitness functions with low-cost approximations.

Numerical simulations are presented using real traffic data from a particular freeway stretch in Attiki Odos motorway, Greece, where recurrent congestion, triggered by a saturated off-ramp, occurs during the morning peak hours. Although traffic congestion originating from a diverging area is quite frequent, appearing mainly at freeways during the peak periods, is not a trivial task to deal with, since it is difficult to determine the freeway exit flow. A high-order finite volume numerical scheme is implemented for the spatial discretization of the differential equations in the GKT model, while time integration is based on a high-order implicit-explicit Runge-Kutta method [25, 26]. The obtained results demonstrate that the proposed model is reasonably accurate in reproducing traffic dynamics, while the parallel DE algorithm can be effectively used for its calibration.

The rest of the paper is organized as follows: In Section 2 a brief description of the second-order GKT model is presented, whereas in Section 3 a description of the major elements composing the proposed numerical optimization scheme is given. In Section 4 the model calibration and validation procedure is presented along with the considered freeway stretch and the traffic data used in the current implementation; the calibration results are evaluated regarding their ability to replicate traffic congestion at a particular saturated off-ramp area. The research findings are summarized in Section 5.

2 The GKT model

This section briefly presents basic definitions and equations governing the second-order GKT model according to [22, 23, 24, 25, 26]. Let $\rho = \rho(x, t)$ and $u = u(x, t)$ denote the traffic

density (vehicles per unit length) and mean speed, respectively, at location x and time instant t , while the traffic flow rate (vehicles per unit time) is given as $q(\rho, u) = \rho(x, t)u(x, t)$. The GKT model can be written in conservation law form with source terms as follows

$$\partial_t \rho + \partial_x(\rho u) = r_{rmp}, \quad (1)$$

$$\partial_t(\rho u) + \partial_x(\rho u^2 + \theta \rho) = \rho \left(\frac{V_e^*(\rho) - u}{\tau} \right) + h_{rmp}. \quad (2)$$

To describe the traffic flow from on-ramps (or to off-ramps), the terms r_{rmp} and h_{rmp} were incorporated on the right-hand sides of Equations (1) and (2). Following from [24], the source term r_{rmp} , which denotes the effective flow density (source density), is only active within the merging (or diverging) zone with length l_{rmp} with inflow $q_{rmp} > 0$ from (or outflow $q_{rmp} < 0$ to) the ramp, and is defined as

$$r_{rmp}(x, t) = \begin{cases} \frac{q_{rmp}(t)}{l_{rmp}} & \text{if } x \text{ is withing merging zone,} \\ 0 & \text{elsewhere.} \end{cases} \quad (3)$$

The source term h_{rmp} in the momentum dynamics Equation (2) describes changes to the macroscopic local speed by assuming that on-ramp vehicles merge to the mainstream road at speed $u_{rmp} < u$. Conversely, the vehicles considered to leave the mainstream road reduce their speed to u_{rmp} before they diverge to the off-ramp. Hence, this term can be written as

$$h_{rmp}(x, t) = \frac{q \cdot r_{rmp}}{\rho} + \frac{(u_{rmp} - u)|q_{rmp}|}{l_{rmp}}. \quad (4)$$

Further, the pressure-like term θ in Equation (2) is computed as $\theta = A(\rho)u^2$, where $A(\rho)$ is given by the Fermi-like function

$$A(\rho) = A_0 + \delta A \left[1 + \tanh \left(\frac{\rho - \rho_{cr}}{\delta \rho} \right) \right], \quad (5)$$

in which ρ_{cr} is the critical density, reflecting the boundary for the transition from the free flow to congested traffic states, with A_0 and $A_0 + 2\delta A$ the variance pre-factors between the aforementioned two states, while $\delta \rho$ is the width of the transition region. Typical parameter

value ranges for A_0 , δA , and $\delta \rho$, along with other typical used model parameters of the GKT model are specified in [22, 23, 24, 25, 26].

The model also includes a traffic relaxation term to keep traffic flow in equilibrium state, with $V_e^* = V_e^*(\rho, u, \rho_\alpha, u_\alpha)$ representing the non-local and dynamic equilibrium speed, while τ is a relaxation time. V_e^* depends not only on the local (ρ, u) but also on the non-local traffic state (ρ_α, u_α) , and is determined as

$$V_e^* = u_{max} \left[1 - \frac{\theta + \theta_\alpha}{2A\rho_{max}} \left(\frac{\rho_\alpha T}{1 - \rho_\alpha/\rho_{max}} \right)^2 B(\delta u) \right]. \quad (6)$$

According to Equation (6), V_e^* is computed as the maximum desired speed, u_{max} , reduced by a term that reflects necessary deceleration maneuvers in traffic flow. The quantities ρ_α, u_α are computed at the downstream anticipation location $x_\alpha = x + \gamma(1/\rho_{max} + T \cdot u)$, where T is the average time-headway, ρ_{max} the maximum density and γ a scale factor. Finally, the Boltzmann interaction factor

$$B(\delta u) = 2 \left[\delta u \frac{e^{-\delta u^2/2}}{\sqrt{2\pi}} + (1 + \delta u^2) \int_{-\infty}^{\delta u} \frac{e^{-y^2/2}}{\sqrt{2\pi}} dy \right] \quad (7)$$

increases monotonically with the normalized velocity difference $\delta u = (u - u_\alpha)/\sqrt{\theta + \theta_\alpha}$ and describes the dependence of the braking interaction on the velocity difference δu between the actual position x and the interaction point x_α .

The major difference with respect to other macroscopic traffic flow models is the non-local character of the GKT model. Specifically, the non-local term in Equation (6) has smoothing attributes like those of a diffusion or viscosity term, but its effect is forwardly directed and, therefore, more realistic. Moreover, unlike other macroscopic models, the steady-state (equilibrium) speed-density relation of GKT model, V^e , is implicitly given from the steady-state condition of homogeneous traffic.

The numerical integration of system Equations (1)-(2), is based here on an accurate and robust higher-order finite-volume relaxation scheme; a fifth-order Weighted Essential Non-Oscillatory-type (WENO) interpolant approach is used for the spatial discretization, while time integration is based on a third-order implicit-explicit (IMEX) Runge-Kutta method. For a detailed description of the spatial and temporal discretization schemes, as well as the superiority and performance of the applied higher-order scheme, compared to low-order ones, in traffic flow simulations, we refer to [25].

3 The optimization scheme

3.1 Surrogate assisted Differential Evolution algorithm

Evolutionary Algorithms imitate nature's selection process using a population-based search mechanism for dealing with demanding high-dimensional real-world optimization problems. They are a class of search methods with efficient balance between exploitation of the best solutions and exploration of the search space, as well as low sensitivity to local minima treatment. They combine elements of directed and stochastic search and, therefore, are more robust than directed search methods. Within the proposed numerical optimization scheme, a Differential Evolution Algorithm (DEVA) is utilized, which is a versatile parallel stochastic search method, introduced by Storn and Price [27], capable of handling non-differentiable, nonlinear and multimodal cost functions, providing superior convergence performance than other EAs [27, 28]. In contrast to other EAs, the DE compares each individual of the population only against a new one, which is its counterpart in the current population. The new parameter vector is a linear combination between a randomly selected one and a weighted difference between two other randomly chosen chromosomes.

Below, an analytical description of the basic elements composing a classic DE algorithm is presented. Given a cost function

$$f_{cost}(\mathbf{x}) = f_{cost}(x_1, x_2, \dots, x_n) \rightarrow \min \quad (8)$$

where, \mathbf{x} denotes the vector containing the n design variables of the problem under consideration and $f(\mathbf{X}): \mathbb{R}^n \rightarrow \mathbb{R}$ a real function. The optimization target is the minimization of the cost function f by modulating the values of its parameters (design variables), while each parameter is bounded between an upper x_i^u and a lower x_i^l value. Differential Evolution evolves a fixed-size population of N_p individuals (chromosomes) for a finite number of generations G_{max} . The initialization of the first population is established by randomly assigning values to the parameters within the given boundaries of the design variables

$$x_{k,i}^0 = r \cdot (x_i^u - x_i^l) + x_i^l, \quad i = 1, \dots, n, \quad k = 1, \dots, N_p, \quad (9)$$

where r denotes a random number generated with uniform probability within the range $[0, 1]$. After the evaluation of each individual's cost function, operators are applied to the population, simulating the according natural processes. The first operator applied is the mutation scheme, which generates a new chromosome, based on three randomly selected individuals of the current generation G . The formation of the new parameter vector is realized by adding a weighted difference vector between the two members of the triad to the third one, the so-called "donor". Then, the uniform crossover scheme is applied; the mutant and the chromosome of the current population are subjected to a discrete recombination which produces the final candidate solution. $x_{k,i}'^{G+1} =$ (10)

$$\begin{cases} x_{C_k,i}^G + F(x_{A_k,i}^G - x_{B_k,i}^G) & \text{if } r \leq C_r \quad \text{or} \quad i = i^* \\ x_{k,i}^G & \text{otherwise,} \end{cases}$$

where $x_{C_k,i}^G$ are the elements of the "donor" vector, G is the current generation and i^* is a randomly selected integer within $[1, n]$, chosen once for all members of the population. The random number r is seeded for every gene of each chromosome, whereas the parameters F and C_r consider the mutation and crossover operations, respectively. Specifically, the scale factor F controls the diversification rate of the population, while the crossover probability C_r controls the fraction of design values that are inherited from the mutant. Moreover, the design variable, which corresponds to the randomly selected index, i^* , is taken from the mutant to ensure that the trial vector does not duplicate the initial one. Subsequently, each member of the resulting intermediate population is evaluated and competes against its counterpart in the current population; the best fitted individuals are the ones that will form the next generation. The DE selection scheme ensures the survival of the elitists and can be described as follows:

$$\mathbf{x}_k^{G+1} = \begin{cases} \mathbf{x}_k'^{G+1} & \text{if } f(\mathbf{X}_k'^{G+1}) \leq f(\mathbf{X}_k^G), \\ \mathbf{x}_k^G & \text{otherwise.} \end{cases} \quad (11)$$

The process is successively repeated, providing populations with better fitted individuals.

The utilized DE is combined with two Artificial Neural Networks (ANNs), a multi-layer perceptron (MLP) and a radial basis functions network (RBFN), which serve as surrogate models. The assistance of these models lies in time-savings, due to avoiding the computationally intensive exact evaluations of each candidate solution, by using a trained neural network instead, while maintaining the robustness and the convergence capabilities of the DE algorithm [29, 30]. These approximate models are established using a data-driven approach, where only the input and output behavior of the simulation model are taken into account, in order to create a mechanism that mimics that behavior. During the evaluation stage of the DE, each trial vector is pre-evaluated by a surrogate model; if it is worse fitted than the corresponding vector of the current population, the current vector is transferred to the next generation. In this way, the efficient damping of less-promising solutions is achieved, yielding an acceleration of the procedure. To enhance the robustness of the methodology, all better fitted trial vectors are exactly re-evaluated. Only exactly evaluated trial vectors with lower values of cost function are transferred to the next generation, ensuring that the derived population is composed only by exactly evaluated chromosomes [29, 30]. In order to create an initial central database for the training and testing data sets of the surrogate models, exact evaluations are performed for each individual chromosome for the first two generations. Subsequently, in each generation a predefined number of the best members are selected to construct the pool of the training and testing data sets; the aforementioned feature results in the creation of a local approximation

model which evolves with the population. Moreover, an additional small percentage of the candidate solutions are selected to be exactly evaluated, to further enhance the robustness of the procedure. The two surrogate models are used simultaneously since each one might point to different regions of the design space constituting a cheap and direct assistance for global optimization. Therefore, in each generation both ANNs are re-trained and tested, while the one with the best performance takes action, considering the output testing errors.

3.2 Parallel implementation

Despite the important contribution of EAs in solving complicated problems, they tend to be excessively time-consuming since they require a considerable number of evaluations; thus, appropriate acceleration through parallel processing appears to be mandatory. The concept behind the developed parallelization strategy is to enable the cooperation of the DE with different simulation software in the form of executables. The required data transfer between the DE and the simulation software is succeeded with appropriate text files, while the communication among the processors and the parallel implementation is achieved using MPI (Message Passing Interface) library functions. The proposed strategy appears to be quite efficient, regardless the use of text files, considering that the computational time of data transfer is negligible compared to the one of the evaluation step.

The population members are distributed a priori among the available processors and each processor is in charge for the evaluation of one individual. Next, a unique rank is assigned to each processor, while one of them is identified as the master node that keeps track of the whole procedure. This node performs all the pre-process that is required prior to the beginning of the optimization procedure, which includes the creation of a working folder for each processor where the executables comprising the evaluation step and their corresponding text files are replicated. Furthermore, the master node distributes all the necessary information concerning the DE algorithm to all other processors, i.e., the number of the design variables, their upper and lower bounds and control parameters for the DE algorithm.

After the completion of the initialization step, the main procedure begins; each processor is generating a random individual within the specified bounds for each gene of the chromosome and evaluates it. Next, the fitness values of the candidate solutions and their corresponding chromosomes are broadcasted to all processors, to update their databases with the new population members. Each processor evolves separately one chromosome, and the new resulting one is stored in its corresponding working folder. All the operations needed (mutation, crossover and selection) for the evolution process, are implemented after the evaluation step of each generation on each node separately for its assigned chromosome. Nevertheless, the auxiliary evaluations of the surrogate models are performed only by the master node for all chromosomes. According to the prescribed procedure, a Boolean array is filled, indicating if the new trial vector is better fitted than its parent and should be exactly evaluated. Subsequently, all selected candidates are exactly evaluated, while, for the rest, the fitness values and trial vectors are explicitly broadcasted by the master node to their corresponding processors for the consistency of the procedure. The optimization process is terminated when a prescribed number of generations is reached.

3.3 Cost Function formulation

As already mentioned, this study addresses the problem of calibrating the parameters of the GKT model to best match the real-measured traffic data by means of the optimization scheme for the calibration and validation of macroscopic traffic flow models. In particular, this methodology attempts to minimize the discrepancy between model generated data and the measurements taken from the detectors in terms of a cost function with appropriate specification of the parameters included in the model. The proposed traffic flow model is fed with real input traffic data to reproduce the complete traffic state, while for evaluating the resulting model accuracy a cost function is introduced as a combined total mean square normalized error of the model-calculated and observed speeds and flows as follows

$$f(\mathbf{X}) = \frac{1}{C} \sum_{k=1}^K \sum_{i=1}^n \left[(1 - \mu) \left(1 - \frac{u_{i,k}}{u_{i,k}^d} \right)^2 + \mu \left(1 - \frac{q_{i,k}}{q_{i,k}^d} \right)^2 \right], \quad (12)$$

where, $u_{i,k}$ and $q_{i,k}$ represent, respectively, the predicted mean speed and flow, computed at detector location k (K is the number of detectors that are available for calibration) and time instant i (n is the simulation time horizon); $u_{i,k}^d$ and $q_{i,k}^d$ represent, respectively, the observed mean speed and flow computed at location k and time instant i , while $C = nK$, and μ is a weighting factor equal to 0.5. The overall optimization scheme for the calibration of the GKT model is illustrated as a flow chart in Figure 1.

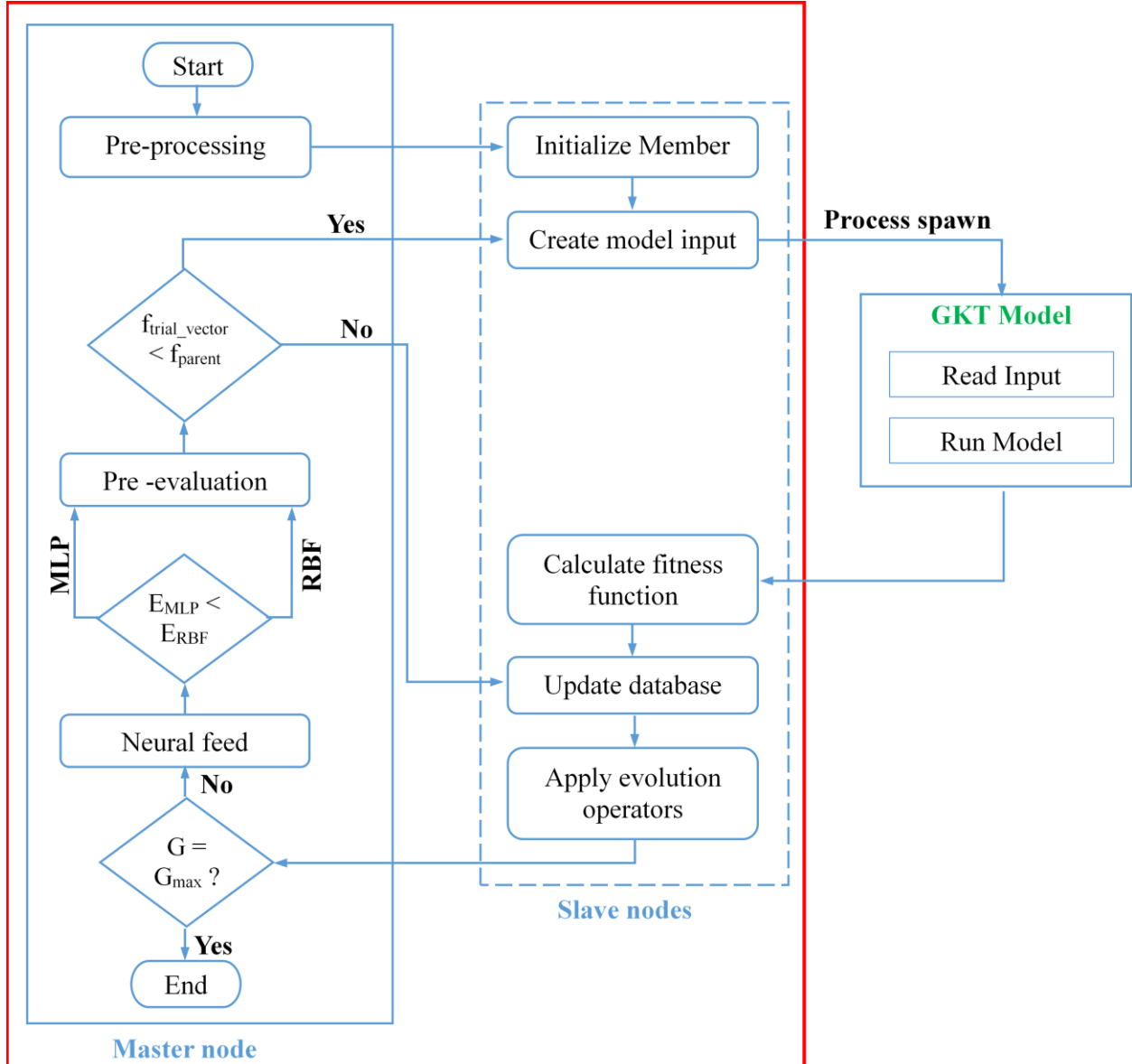


Figure 1: Flowchart with the major steps of the optimization process.

4 Model calibration and validation

The calibration process constitutes a crucial step for the successful development and application of any macroscopic traffic flow model in transportation planning and traffic management. Indeed, the reliability and credibility of a model to reproduce different traffic flow conditions in a freeway network with the best possible accuracy is of major importance. However, the estimation of proper parameter values used in the model is a challenging problem, since the GKT model equations are highly nonlinear in both its parameters and state variables.

The numerically discretized GKT model is tested for a particular network where recurrent freeway congestion is triggered by a saturated off-ramp and propagates upstream along the freeway mainstream. Hence, the model is first calibrated to identify the optimal parameter values for the examined freeway test network, using measured data from a specific date. Eventually, in order to investigate the representativeness of the calibration results in replicating the traffic conditions of the considered site, the optimal parameter values resulting from the calibration process are applied to the same freeway network for a different day.

Herein, the parameter vector for the calibration of the GKT model is $\mathbf{x} = [u_{max}, \rho_{max}, \rho_{cr}, T, \gamma, \tau, A_0, \delta A, \delta \rho]$ and it should be selected so as to minimize the deviation between the model results and the real measured traffic data, using cost function (12). Thus, the calibration process becomes a problem of finding an optimal parameter vector \mathbf{x} for the model Equations (1) and (2) subject to $\mathbf{x} \in \Omega$, where Ω is a constrained admissible region of the parameter space, determined on the basis of physical constraints.

4.1 Test network and traffic data

The test network considered in this study, for the calibration of the GKT model parameters under recurrent traffic conditions, is a stretch of Attiki Odos motorway in Athens, Greece (with direction from the Airport to Elefsina). Specifically, the chosen network, as shown in Figure 2, is 6.2 km long (from 34th to 27.8th km), is composed by three lanes, which become four between the 30.8th to 30.3rd km, and includes three on-ramps and three off-ramps. Referring to Figure 2, the location of the available detector stations are depicted with bullets.

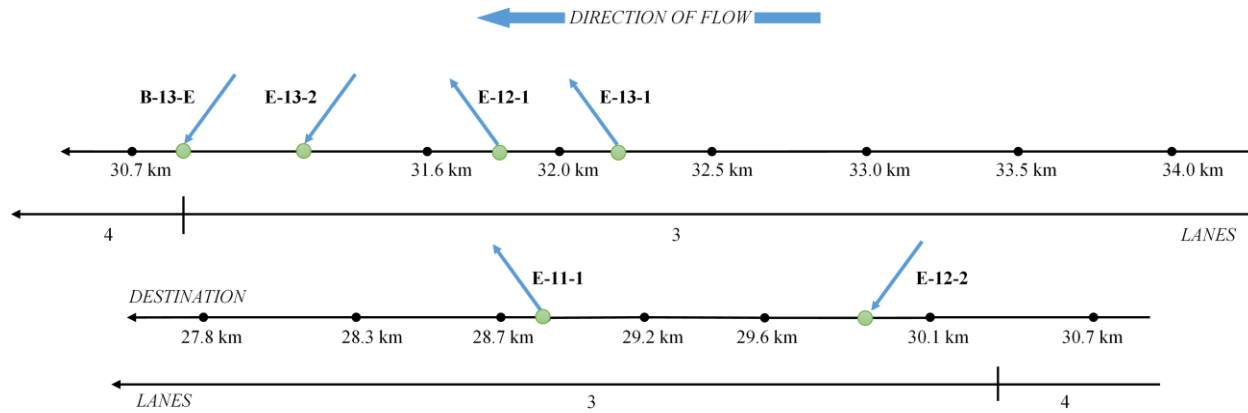


Figure 2: Graphical representation of the considered Attiki Odos freeway stretch.

The real traffic data were provided by the company ATTIKES DIADROMES S.A., which is responsible for the operation and maintenance of the motorway, as well as for traffic management and customer service. In particular, data is collected through 13 detector-stations (Figure 2) that deliver measurements of flow and speed per lane with a time resolution of 20 s, for the time period May–June 2009. Detectors are also installed on the ramps, measuring the corresponding ingoing and outgoing traffic flows. Measured data corresponding to the stretch’s boundaries is also available.

A qualitative analysis of real-time traffic data showed that, within this examined freeway stretch, a recurrent congestion is formed during a typical morning rush hour. Figure 3 illustrates the space-time evolution of the real speed measurements for 16th June 2009 and 23rd June 2009. It can be observed that traffic congestion is formed upstream of the off-ramp E-11-1, persisting between 8-10 a.m. for both days, albeit with slightly different space-time shapes due to accordingly different flow propagation on both days. In particular, this congestion originates from the 29th km of the considered network and spills back on the upstream infrastructure for several kilometers, reaching up to the 33rd km. The reason behind this major congestion created in the specific diverging area is a combination of the increased exit flow that the off-ramp E-11-1 receives and its limited capacity. It is worth noting that these two days were carefully selected so that, during the morning hours, no incident and no sensor failure occurred at the consider test network.

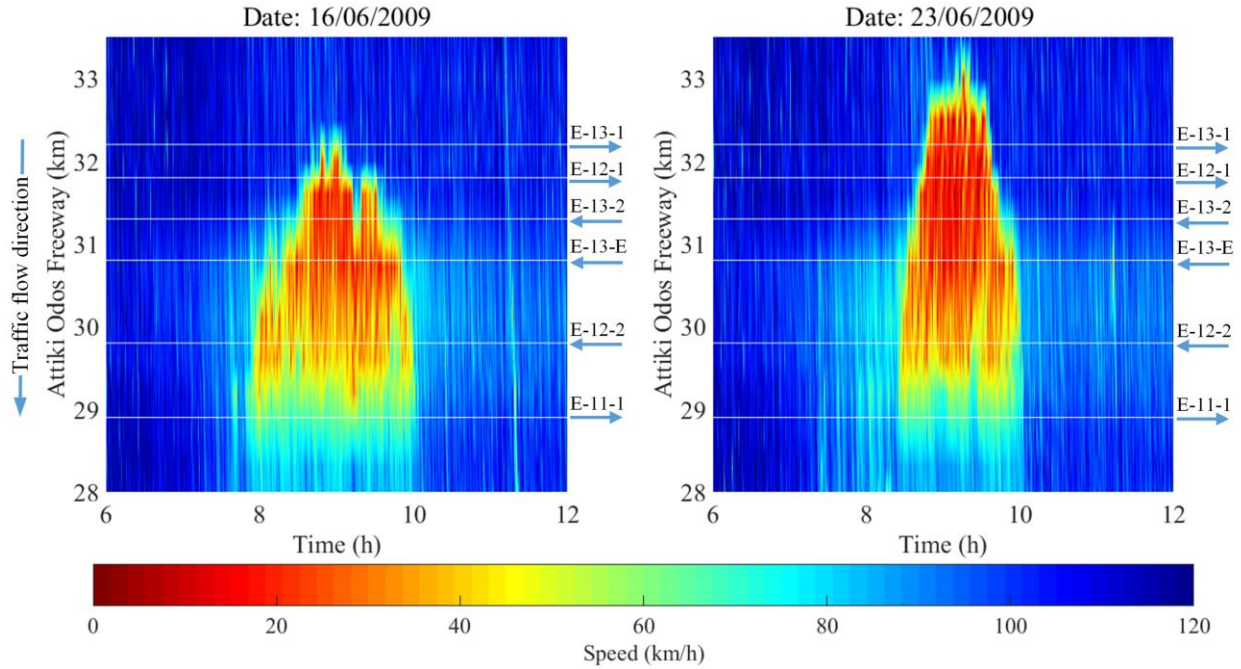


Figure 3: Phase space speed dynamics at the considered freeway stretch for two different days.

4.2 Calibration results

The real traffic data used in this section were collected on the Attiki Odos motorway in Greece on the 16th of June, 2009. The GKT model parameters, with their respective upper and lower bounds being in line with those given in [12, 22, 23, 24, 25], are presented in Table 1. The population size of the DE algorithm was specified equal to 60, while the algorithm was executed for 1500 generations; the control parameters for the mutation and crossover operations were $F = 0.6$ and $C_r = 0.45$, respectively.

The considered 6.2 km stretch was simulated for 6 morning hours (i.e. from 6 a.m. to 12 p.m.), whereas the space discretization was for $\Delta x = 50$ m and the Courant-Friedrichs-Lewy (CFL) numerical stability parameter value was set equal to 0.4. The runs of the DE algorithm have been performed on DELLTM R815 PowerEdge™ server with four AMD Opteron™ 6380 sixteen-core processors at 2.50 GHz (64 cores in total). The clock computational time for 1200 generations was 1081 min. The resulted optimal model parameters and the minimum value of the cost function are presented in Table 2. In Figure 4, the convergence history for the best chromosomes of each generation is illustrated.

Model parameters	Units	Bounds
Desired free speed, u_{max}	km/h	[110, 130]
Maximum density per lane, ρ_{max}	veh/km	[100, 200]
Critical density per lane, ρ_{cr}	veh/km	[30, 60]
Desired time gap, T	s	[0.8, 2.2]
Anticipation factor, γ		[1, 2]
Relaxation time, τ	s	[10, 40]
Variance pre-factor for free traffic, A_0		[0.0025, 0.015]
Pre-factor, δA		[0.03, 0.035]
Transition width, $\delta \rho$	veh/km	[3.5, 20]

Table 1: Range of the parameter vector used for the GKT model calibration.

Figure 5 presents the space-time diagrams of the real measured speeds, contrasted to the model's estimation of speed for the calibration day. Figure 6 displays the measured and estimated speed dynamics for all detector stations, while in Figure 7 the flow dynamics for all detector stations are shown. From these figures, it can be deduced that the real traffic conditions are well reproduced by the calibrated model, which captures with sufficient accuracy when and where the traffic flow becomes congested, for the correct duration and extent. Quite importantly, it is clear in Figure 7 that the GKT model is able to reproduce the capacity drop phenomenon at the congestion head.

<i>Model parameters</i>	u_{max} (km/h)	ρ_{max} (veh/km)	ρ_{cr} (veh/km)	T (s)	γ	τ (s)	A_0	δA	$\delta \rho$ (veh/km)
<i>Optimal values</i>	1105	172	30	1.06	1.99	30.4	0.015	0.003	19
<i>Cost function (%)</i>	4.98								

Table 2: Resulted optimal parameter values for the GKT model.

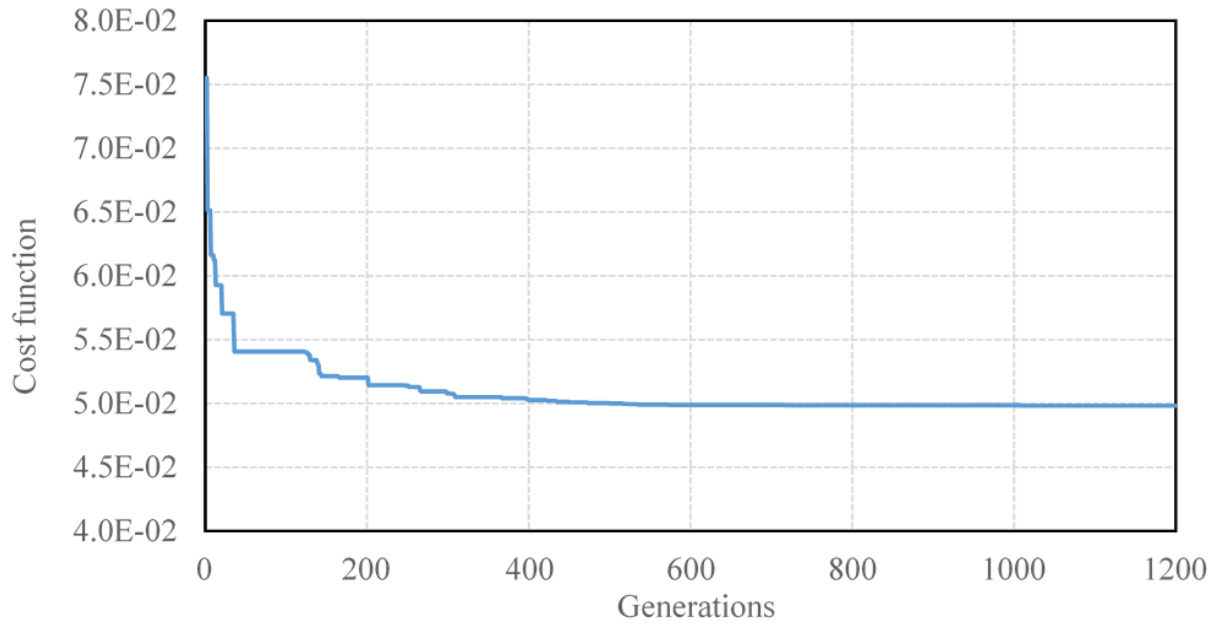


Figure 4: The convergence history of the best solution of each generation of the DE algorithm.

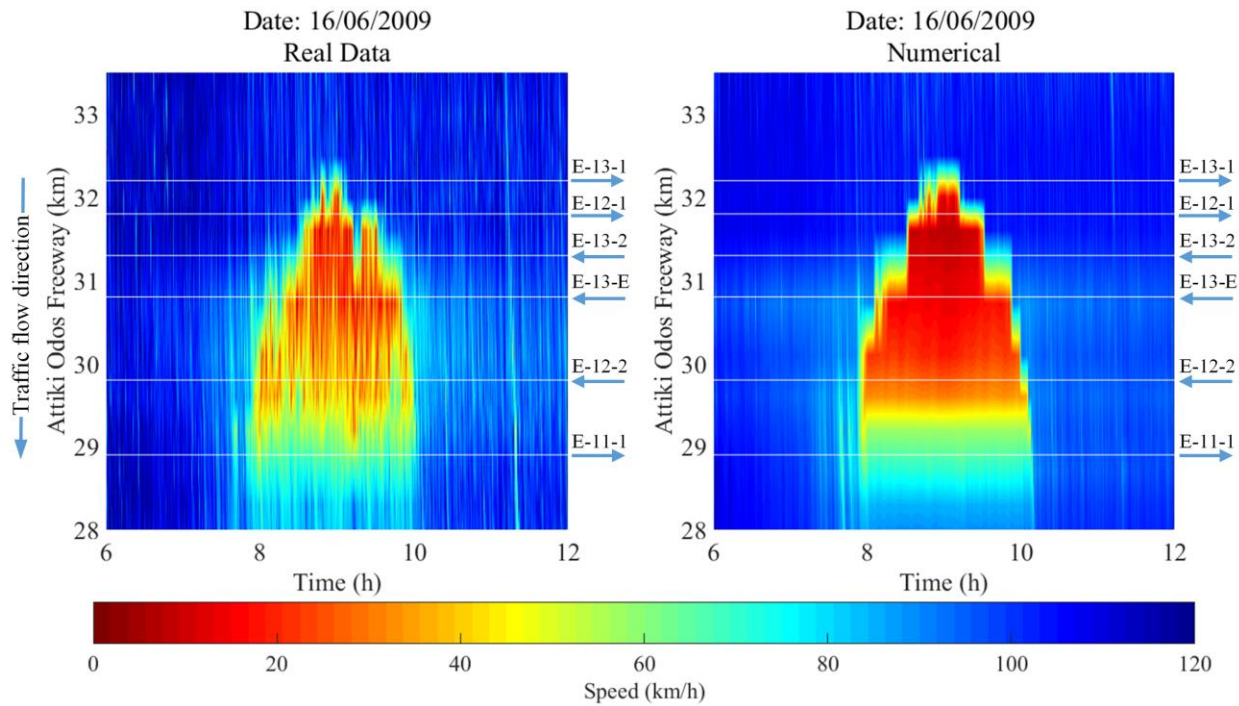


Figure 5: Phase space speed dynamics for real measured speed (left) and the model prediction (right) for the calibration day.

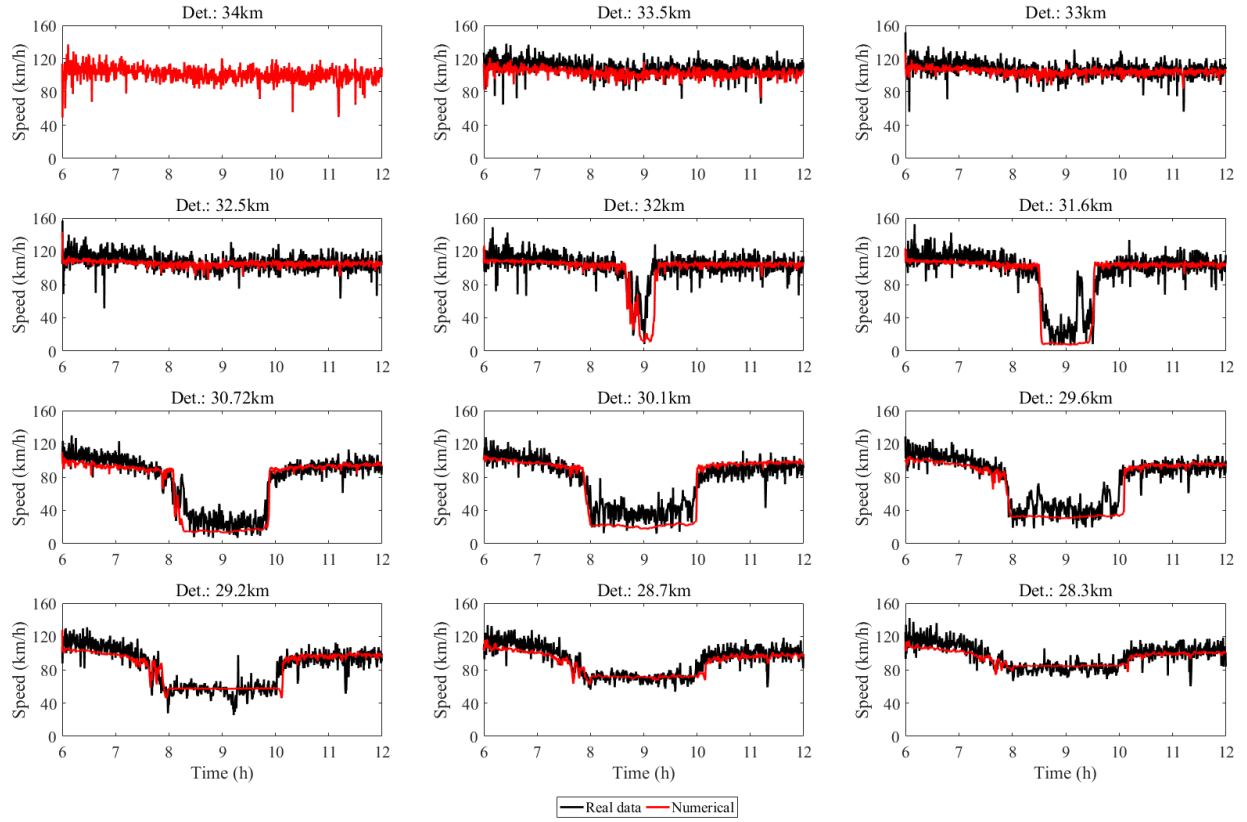


Figure 6: Time series of the real speed measurements (black) and the model prediction of speed (red) at various detector locations for the calibration day.

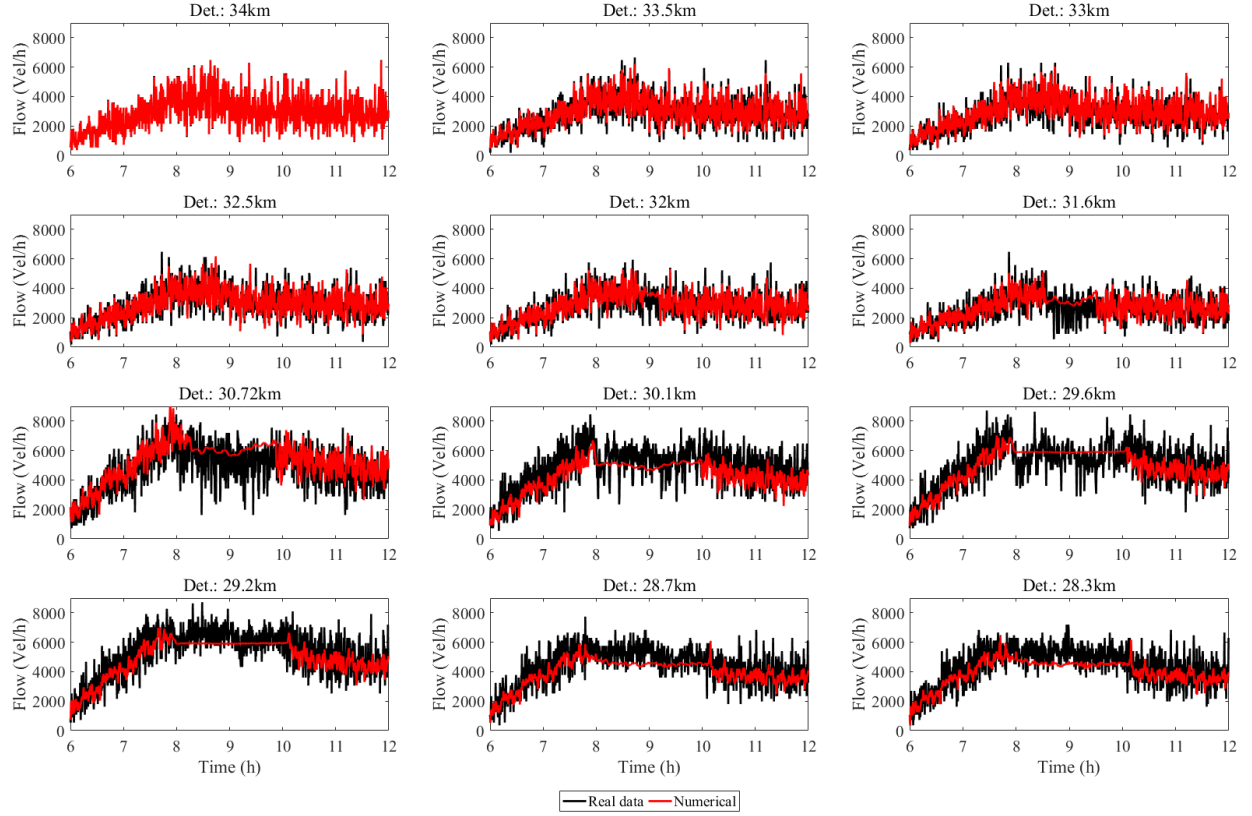


Figure 7: Time series of the real flow measurements (black) and the model prediction of flow (red) at various detector locations for the calibration day.

4.3 Validation results

In order to test and assess the robustness of the resulting GKT model, the optimal parameters resulting from the previous calibration procedure are applied to the same freeway stretch, but for a different day, the 23rd of June 2009. For the validation day, the recurrent congestion in the traffic flow behavior is similar to the one presented for the calibration day, but has a slightly different space-time shape due to accordingly different traffic conditions.

The validation results are presented in Figures 8, 9 and 10. The obtained results can be considered as satisfactory, since the traffic flow model is able to capture with sufficient accuracy the real traffic flow conditions in the particular freeway stretch; the proposed model is able to reproduce how the congestion occurs and propagates in time and space, although not at the exact same level of accuracy as for the calibration date. The main difference is that the duration of the congestion is overestimated at some freeway locations. The cost function value for this validation procedure was 14%.

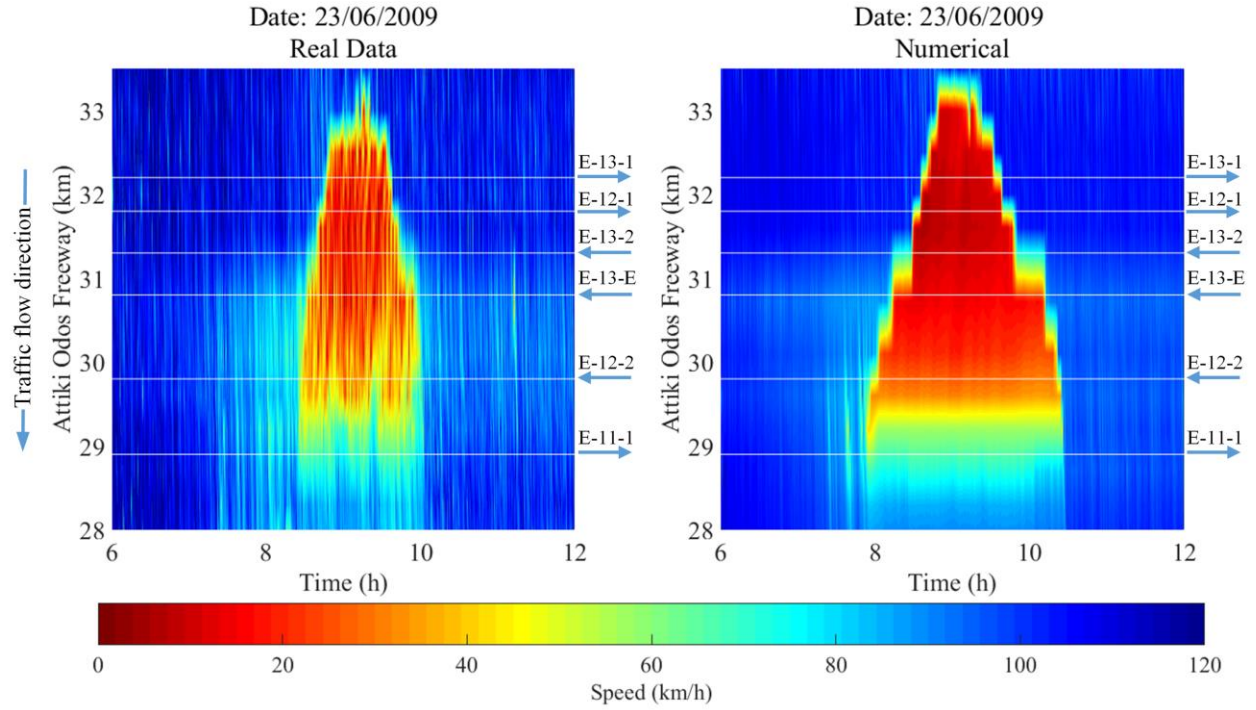


Figure 8: Phase space speed dynamics for real measured speed (left) and the model prediction (right) for the validation day.

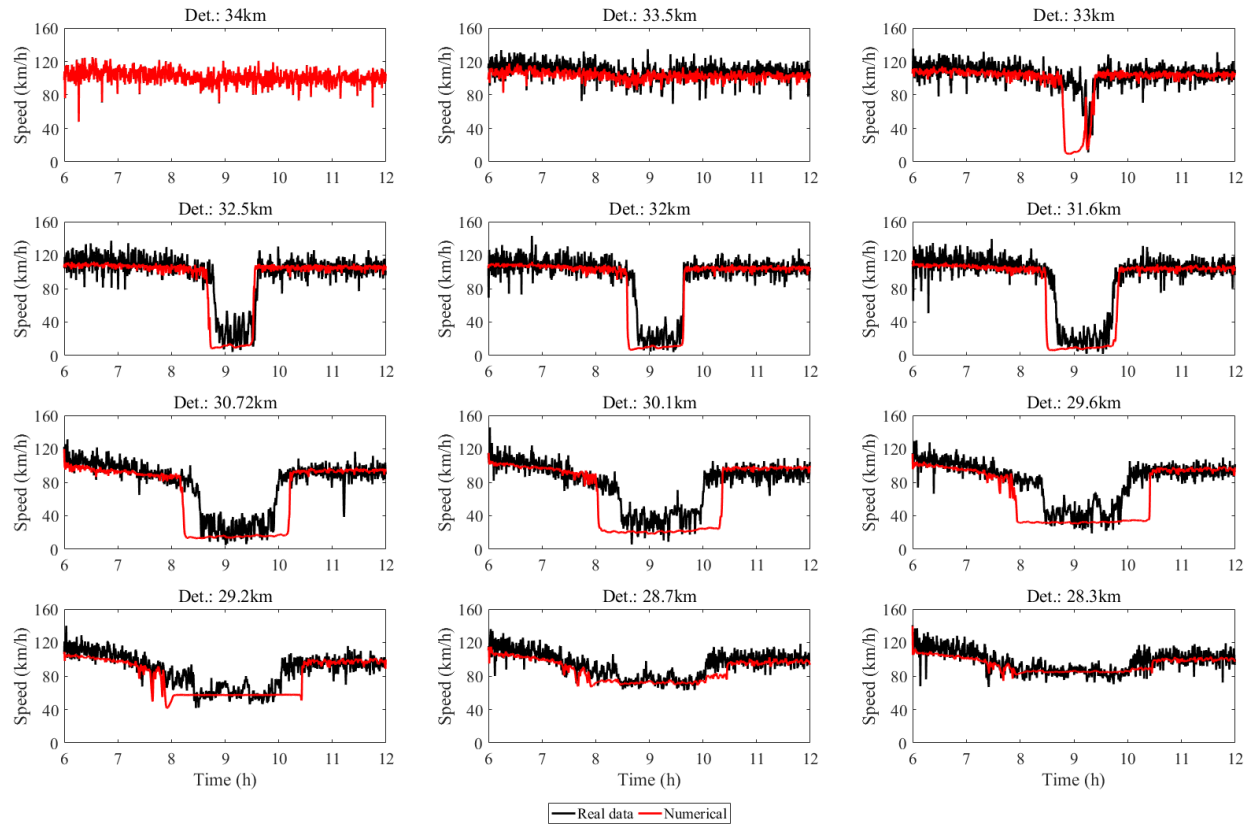


Figure 9: Time series of the real speed measurements (black) and the model prediction of speed (red) at various detector locations for the validation day.

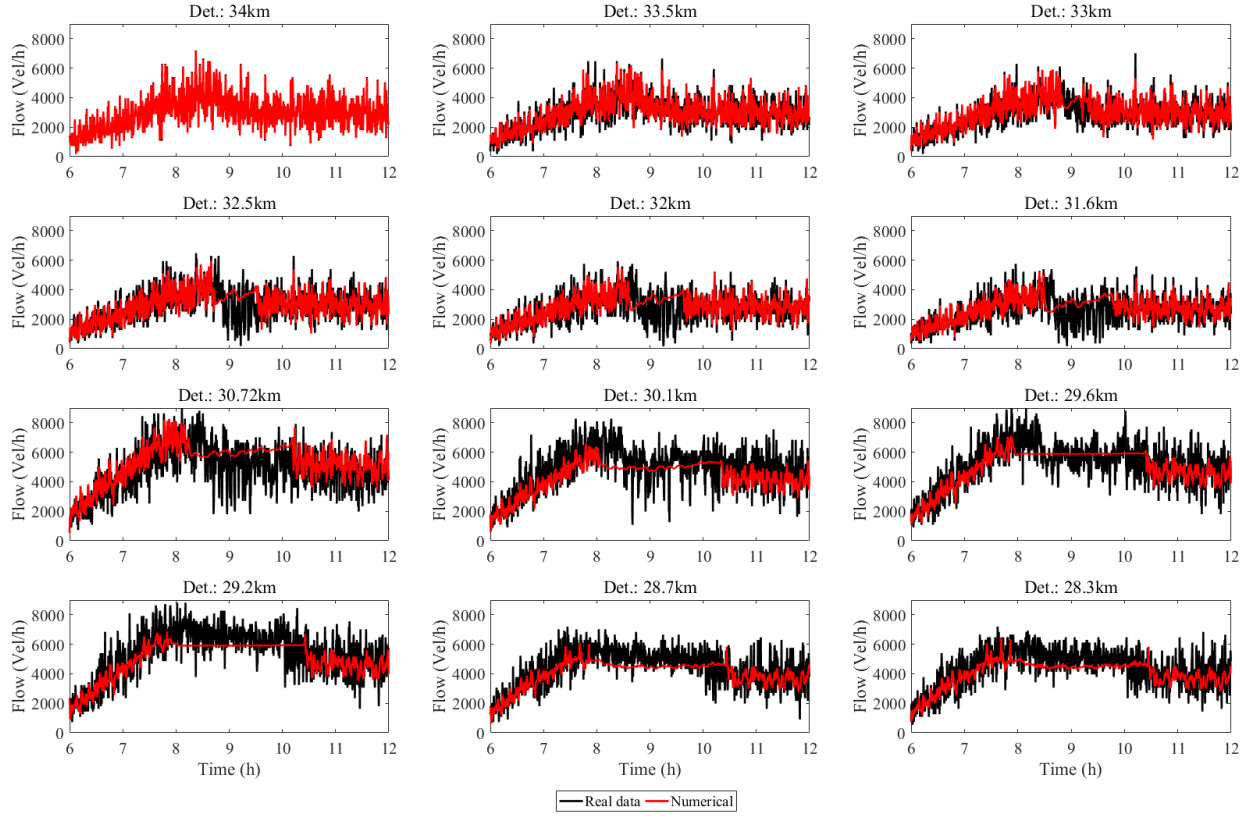


Figure 10: Time series of the real flow measurements (black) and the model prediction of flow (red) at various detector locations for the validation day.

5 Conclusions

In this study, a recently developed parallel, metamodel-assisted DE algorithm was employed for the automated calibration of the parameters of a second-order macroscopic GKT traffic flow model, using real traffic data. Following from the numerical results, the DE algorithm proved to be a viable candidate and versatile tool for the calibration of macroscopic traffic flow models having counteracting calibration parameters in which the cost function exhibits multiple local minima. The resulted optimal values of the model parameters indicated that the proposed DE algorithm guarantees the convergence of the best solutions; as well as that the GKT model is able to replicate with sufficient accuracy the prevailing traffic flow conditions; the credibility of the calibrated parameters is demonstrated through the validation procedure.

Acknowledgements

This research was supported by TRAffic MANagement for the 21st century (TRAMAN21) ERC Advanced Investigator Grand under the European Union's Seventh Framework Program (FP/2007-20013). The authors are grateful to ATTIKES DIADROMES S.A. for providing the utilized traffic data from Attiki Odos freeway in Athens, Greece.

References

1. S. P. Hoogendoorn and P. H. L. Bovy, "State-of-the-art of Vehicular Traffic Flow Modelling", *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 215(4), 283–303, 2001.
2. M. Cremer and M. Papageorgiou, "Parameter Identification for a Traffic Flow Model", *Automatica*, 17(6), 837–843, 1981.
3. M. Cremer and J. Ludwig, "A Fast Simulation Model for Traffic Flow on the Basis of Boolean Operations", *Mathematics and Computers in Simulation*, 28(4), 297–303, 1986.
4. M. Papageorgiou, M., J.-M. Blosseville and H. Hadj-Salem, "Macroscopic Modelling of Traffic flow on the Boulevard Périphérique in Paris", *Transportation Research Part B Methodological*, 23(1), 29–47, 1989.
5. A. Kotsialos, M. Papageorgiou, C. Diakaki, Y. Pavlis and F. Middelham, "Traffic Flow Modeling of Large-scale Motorway Networks Using the Macroscopic Modeling Tool METANET", *IEEE Transactions on Intelligent Transportation Systems*, 3(4), 282–292, 2002.
6. T. Monamy, H. Hadj-Salem and J.-P. Lebacque, "A Macroscopic Node Model Related to Capacity Drop, *Procedia - Social and Behavioral Sciences*", 54, 1388–1396, 2012.
7. D. Ngoduy, S. Hoogendoorn and H. Van Zuylen, "Comparison of Numerical Schemes for Macroscopic Traffic Flow Models", In *Transportation Research Record: Journal of the Transportation Research Board*, 1876, 52–61, 2004.
8. A. Spiliopoulou, M. Kontorinaki, M. Papageorgiou and P. Kopelias, "Macroscopic Traffic Flow Model Validation at Congested Freeway Off-ramp Areas", *Transportation Research Part C Emerging Technologies*, 41, 18–29, 2014.
9. A. Poole and A. Kotsialos, "METANET Model Validation Using a Genetic Algorithm", In *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 7-12, 2012.
10. A. Spiliopoulou, I. Papamichail, M. Papageorgiou, I. Tyrinopoulos and J. Chrysoulakis, "Macroscopic Traffic Flow Model Calibration Using Different Optimization Algorithms", *Transportation Research Procedia*, 6, 144–157, 2015.
11. D. Ngoduy and M. J. Maher, "Calibration of Second-order Traffic Models Using Continuous Cross Entropy Method", *Transportation Research Part C Emerging Technologies*, 24, 102–121, 2012.
12. K.N. Porfyri, I.K. Nikolos, A.I. Delis and M. Papageorgiou, "Calibration of a second-order traffic flow model using a metamodel-assisted Differential Evolution algorithm", *Proceedings of the IEEE 19th International Conference on Intelligent Transportation Systems*, Rio de Janeiro, Brazil, 366–371, 2016.
13. E. Alba, M. Tomassini, "Parallelism and Evolutionary Algorithms", *IEEE Transactions on Evolutionary Computation*, 6, 443–462, 2002.

14. E. Cantu-Paz, "Designing Efficient and Accurate Parallel Genetic Algorithms:", Kluwer Academic Publishers, Norwell, MA, 2000.
15. J.J. Hu, E.D. Goodman, "The Hierarchical Fair Competition (HFC) Model for Parallel Evolutionary Algorithms", Congress on Evolutionary Computation, CEC2002, IEEE Press, 2002.
16. E.D. de Jong, D. Thierens, R.A. Watson, "Hierarchical Genetic Algorithms", Parallel Problem Solving from Nature – PPSN VIII, Vol. 3242 of CNCS, Birmingham, UK, Springer – Verlag, 2004.
17. A.C.M. Oliveira, L.A.N. Lorena, S. Stephani, "An Hierarchical Fair Competition Genetic Algorithm for Numerical Optimization", 8th Brazilian Symposium on Neural Networks, First Brazilian Workshop on Evolutionary Computing, Sao Luis, 2004.
18. E. Alba, J.M. Troya, "An Analysis of Synchronous and Asynchronous Parallel Distributed Genetic Algorithms with Structured and Panmictic Islands", IPPS/SPDP Workshops, 1999.
19. E. Alba, J.M. Troya, "Analyzing Synchronous and Asynchronous Parallel Distributed Genetic Algorithms", Future Generation Computer Systems, 17, 14, 2001.
20. X. Llorca and J.M. Garrell, "Knowledge-Independent Data Mining With Fine-Grained Parallel Evolutionary Algorithms", Genetic and Evolutionary Computation Conference, San Francisco, California, Morgan Kaufmann, 2001.
21. X. Li, M. Kirley, "The Effects of Varying Population Density in a Fine-grained Parallel Genetic Algorithm", IEEE World Congress on Computational Intelligence, Piscataway, NJ, 2002.
22. M. Treiber, A. Hennecke and D. Helbing, "Derivation, properties, and simulation of a gas-kinetic-based traffic model", Physical Review E, 59(1), 239-253, 1999.
23. D. Helbing, A. Hennecke, V. Shvetsov and M. Treiber, "MASTER: Macroscopic traffic simulation based on a gas-kinetic, non-local traffic model", Transportation Research Part B: Methodological, 35(2), 183-211, 2001.
24. M. Treiber and A. Kesting, Traffic flow dynamics: Data, models and simulation, Springer, 2013.
25. A. I. Delis, I. K. Nikolos, and M. Papageorgiou, "High-resolution numerical relaxation approximations to second-order macroscopic traffic flow models", Transportation Research Part C: Emerging Technologies, 44, 318-349, 2014.
26. A.I. Delis, I.K. Nikolos and M. Papageorgiou, "Adaptive cruise control for a macroscopic traffic flow model: development and numerical solution", Computers and Mathematics with Applications, 70, 1921-1947, 2015.
27. R. Storn and K. Price, Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces, ICSI Berkeley, 1995.
28. K.V. Price, R.M. Storn, J.A. Lampinen, Differential Evolution: A Practical Approach to Global Optimization, Springer-Verlag, Berlin/Heidelberg, 2005.
29. I.K. Nikolos, "Inverse design of aerodynamic shapes using differential evolution coupled with artificial neural network," Proceedings of the ERCOFTAC Conference in Design Optimization: Methods and Applications, Athens, 2004.
30. I.K. Nikolos, "Surrogate Modeling in Evolutionary Based Engineering Design Optimization," Computational Science, Engineering and Technology Series, Y. Tsompanakis, and B.H.V. Topping, Saxe-Coburg Publications, Stirlingshire, UK, 173-203, 2011.