Swarm Intelligence Algorithms for Macroscopic Traffic Flow Model Validation with Automatic Assignment of Fundamental Diagrams

Adam Poole and Apostolos Kotsialos

School of Engineering and Computing Sciences, Durham University, South Road DH1 3LE, Durham, United Kingdom

Abstract

This paper is concerned with the problem of macroscopic road traffic flow model calibration and verification. Thoroughly validated models are necessary for both control system design and scenario evaluation purposes. Here, the second order traffic flow model METANET was calibrated and verified using real data.

A powerful optimisation problem formulation is proposed for identifying a set of model parameters that makes the model fit to measurements. For the macroscopic traffic flow model validation problem, this set of parameters characterise the aggregate traffic flow features over a road network. In traffic engineering, one of the most important relationships whose parameters need to be determined is the fundamental diagram of traffic, which models the non-linear relationship between vehicular flow and density. Typically, a real network does not exhibit the same traffic flow aggregate behaviour everywhere and different fundamental diagrams are used for covering different network areas. As a result, one of the initial steps of the validation process rests on expert engineering opinion assigning the spatial extension of fundamental diagrams. The proposed optimisation problem formulation allows for automatically determining the number of different fundamental diagrams to be used and their corresponding spatial extension over the road network, simplifying this initial step. Although the optimisation problem suffers from local minima, good solutions which generalise well were obtained.

The design of the system used is highly generic and allows for a number of evolutionary and swarm intelligence algorithms to be used. Two UK sites have been used for testing it. Calibration and verification results are

discussed in detail. The resulting models are able to capture the dynamics of traffic flow and replicate shockwave propagation.

A total of ten different algorithms were considered and compared with respect to their ability to converge to a solution, which remains valid for different sets of data. Particle Swarm Optimisation (PSO) algorithms have proven to be particularly effective and provide the best results both in terms of speed of convergence and solution generalisation. An interesting result reported is that more recently proposed PSO algorithms were outperformed by older variants, both in terms of speed of convergence and model error minimisation.

Keywords: Traffic flow models, model validation, parameter estimation, Intelligent Transportation Systems, particle swarm optimisation, genetic algorithms, cuckoo search.

---

1. Introduction and Background

Traffic modelling is an essential element of traffic planning and management systems. Traffic models are mainly used for evaluation and system design purposes. Intelligent Transportation Systems (ITS) operating in motorway networks require the use of valid models for tasks like traffic prediction, state and travel time estimation, and real time model based predictive control. Automatic incident systems also make use of such models. Irrespective of their purpose, models have to be valid for the specific road network they are used for.

This paper is concerned with the problem of macroscopic traffic flow model validation and, hence, microscopic and mesoscopic models are beyond its scope. For a more detailed discussion on modelling approaches, see [1].

Macroscopic models describe the traffic flow as a liquid using aggregate variables. At point x on the road and time t, these are the vehicular density $\rho(x,t)$ (veh/km), mean speed $v(x,t)$ (km/h) and flow (or volume) $q(x,t)$ (veh/h). The macroscopic description of traffic along a motorway was introduced in the seminal papers of Lighthill and Whitham [2] and Richards [3], resulting to the LWR model.

The LWR model employs the vehicle conservation equation to calculate densities and flows, which reads

$$\frac{\partial \rho(x,t)}{\partial t} + \frac{\partial q(x,t)}{\partial x} = 0. \tag{1}$$

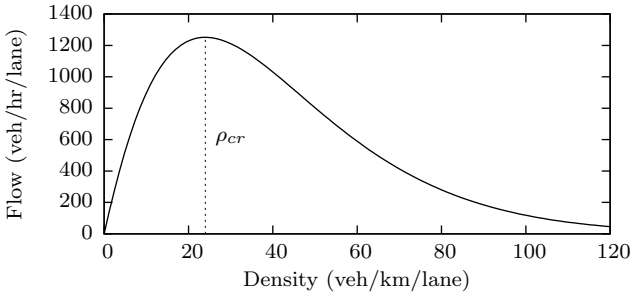


Figure 1: Example of the Fundamental Diagram.

In order for eqn. (1) to be solved, the relationship between flow and density must be explicitly considered. In the LWR theory it is given in the form

$$q(x,t) = \rho(x,t)V\left[\rho(x,t)\right] \tag{2}$$

where $V\left[\rho(x,t)\right]$ (km/h) is an equilibrium relationship between density and mean speed, i.e. the so-called Fundamental Diagram (FD) of traffic. The FD models the traffic flow's tendency to settle to the equilibrium mean speed $V[\rho(x,t)]$ for a given density level $\rho(x,t)$. The typical shape of the density-flow FD, i.e. the way the quantity $(\rho(x,t)V[\rho(x,t)])$ changes with respect to $\rho(x,t)$ is shown in Fig. 1. The FD accounts for the fact that until the critical density ρ_{cr} is observed, the vehicular flow increases with increasing density. The flow is maximised at ρ_{cr}, and when the density increases past that level, the number of vehicles contained per unit of length is such that drivers are forced to slow down, reaching to zero speed at jam density level $\rho_{\max}$. Different functional expressions for the FD have been proposed in the literature, see [4, 5, 6].

Typically, a space and time discretised version of eqn. (1), along with the FD constitute the basic elements of a first order macroscopic traffic flow model, [2, 3, 7, 8, 9, 10, 11].

Payne-type second order models result from coupling (1) with an empirical equation governing the mean speed $v(x,t)$ dynamics, [12]; this equation has the form

$$\frac{\partial v(x,t)}{\partial t} + v(x,t)\frac{\partial v(x,t)}{\partial x} + \frac{1}{\rho(x,t)}\frac{\partial P(x,t)}{\partial x} = \frac{1}{\tau}\left\{V\left[\rho(x,t)\right] - v(x,t)\right\} \tag{3}$$

where τ is a relaxation constant and $P(x,t)$ a pressure term, which gives rise to a range of different models, [13].

Irrespective of the model's order, a number of parameters characterising the aggregate driver-vehicle-infrastructure behaviour are used. For any practical purpose the values assigned to them are based on real data collected from the road network. Using data sets of traffic counts and vehicle speeds, typically obtained by means of inductive loop detectors embedded in the motorway, a rigorous model validation procedure needs to take place, for identifying an optimal set of parameters, [14]. The validation process consists of two parts, model calibration and model verification.

The calibration phase aims at determining an optimal set of model parameters that minimises the error for a specific data set. Verification is then performed to corroborate the model's accuracy using a different set of data, not used during calibration. Model validation is a difficult procedure due to the sensitivity and the non-linear nature of the traffic flow process. Furthermore, the resulting optimisation problem has numerous local minima, [15].

Here, the second order model Modéle d'Écoulement de Trafic sur Autoroute (META) [16] and its extension to networks, META-NETworks (METANET) [17], is used as a modelling tool. The METANET simulator and its predecessors have been successfully validated for networks of various sizes. A more extensive model validation exercise for the META model was conducted for the Paris ring road in [18]. The validation of the large scale network of the Amsterdam peripheral network is described in [16]. For the modelling of the Paris and Amsterdam sites, the deterministic search algorithm of Box [19] was used. A simplex based algorithm was used by [20] to validate various numerical schemes.

In [21] a method calculating the model parameters by comparing the congestion pattern of the data and model output aiming at avoiding incorrect data forms, was used. A cross entropy method is used in [15] to validate the model used in [21] for a 10km section of a UK highway. A comparative study of the first order Cell Transmission Model (CTM) [7, 8] and METANET for a motorway in Greece based on the Nelder-Mead algorithm [22] is provided in [23]. The use of a genetic algorithm to validate METANET on a simple site in the UK is reported in [24]. In [25] a METANET model parameter identification algorithm is discussed using data from a 4.65 km stretch of a California highway; the original expression used for FD in METANET is replaced with a two-regime model and the resulting optimisation problem is solved using a sequential quadratic programming algorithm.

Motivated by the requirements generated for designing autonomic traffic

management systems, i.e. systems that exhibit self-$*$ (e.g. self-optimising, self-healing, self-configuring and so forth) properties, the model validation problem's scope was extended to automate elements that traditionally are based on engineering expert opinion [26]. This is one of this paper's contributions. The proposed problem formulation is able to replace expert engineering opinion and judgement by automatically selecting the spatial extension of the application of a FD along with its parameters. This extra requirement increases the problem complexity but removes the need for prior expert knowledge about congestion patterns. Bottleneck identification as used in [27, 28] or choosing an arbitrary (based on educated opinion) point for a change in the FD parameter set [16] is not explicitly required. These are left to the optimisation algorithm to deal with, aiming at avoiding over-parametrisation as well. Within this setting, this paper aims at providing the details of mainly particle swarm optimisation (PSO) performance when used for the METANET model validation problem.

In this manner many different categories of algorithms have been applied to various engineering problems. An application of differential evolution to constrained combinatorial problems is shown in [29]. A multi-objective genetic algorithm has been used to optimise electrical drives [30]. A gravitational search is conducted to optimise a fuzzy servo controller in [31]. Particle swarm optimisation has been used for reservoir optimisation [32], and hydrothermal scheduling [33].

In this paper three classes of algorithms are evaluated, Particle Swarm Optimisation (PSO), Genetic Algorithm (GA) and Cuckoo Search (CS). The emphasis of this paper is on the particle swarm optimisation and a variety of PSOs are used. The GA used is a simple one and is included as a baseline. The CS algorithm, is included because it has been shown to outperform PSO for some problems, [34, 35].

The suggested system has been applied to two UK sites that have their own congestion patterns. The data were obtained from the Highways Agency owned system Motorway Incident Detection and Automated Signalling (MIDAS).

The rest of this paper is organised as follows. Section 2 provides an overview of the METANET model. Section 3 provides the optimisation problem formulation. Section 4 provides the used site descriptions. Section 5 is concerned with the details of the algorithms used. Results are provided and discussed in section 6. Section 7 concludes this paper.

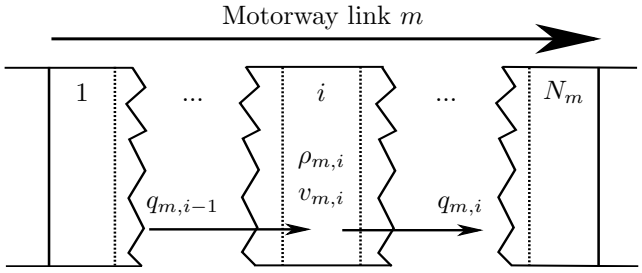


Figure 2: Link discretisation.

2. METANET Model Overview

The METANET simulator is a discrete form of Payne's model (3) and is able to model arbitrary motorway networks of any topology. A network is represented as a directed graph consisting of nodes and links. With links representing homogeneous road sections, where the number of lanes is a constant, and there is no significant change of curvature or gradient. Furthermore, no traffic sources or sinks (junctions) exist within a link. Nodes connect the links and are used at places where the geometry of the motorway changes or at on-/off-ramp junctions. Each node has at least one incoming and one outgoing link. Traffic enters via origin links and leaves through destination links.

Time is discretised globally by a time step T and there are K steps in the time horizon. Each motorway link m is discretised into N_m segments of equal length L_m as shown in Fig. 2. The traffic variables describing traffic conditions in segment i of link m, at time instant $t = kT$, $k = 0, 1, \ldots, K$, are: (a) the traffic density $\rho_{m,i}(k)$ (veh/km/lane), which represents the number of vehicles in the segment divided by L_m and the link's number of lanes λ_m; (b) the mean speed $v_{m,i}(k)$ (km/h) representing the mean speed of the vehicles in segment i at time kT; (c) the traffic flow $q_{m,i}(k)$ (veh/h), which is the number of vehicles leaving segment i of link m during interval $[kT, (k+1)T]$, divided by T.

By discretising equations (1) and (3), [18, 17, 36], the discrete time motorway traffic flow model is the following.

$$\rho_{m,i}(k+1) = \rho_{m,i}(k) + \frac{T}{L_m \lambda_m}[q_{m,i-1}(k) - q_{m,i}(k)] \tag{4}$$

$$q_{m,i}(k) = \rho_{m,i}(k) v_{m,i}(k) \lambda_m \tag{5}$$

$$\begin{aligned} v_{m,i}(k+1) &= v_{m,i}(k) + \frac{T}{\tau}\{V[\rho_{m,i}(k)] - v_{m,i}(k)\} \\ &\quad + \frac{T}{L_m} v_{m,i}(k)[v_{m,i-1}(k) - v_{m,i}(k)] \\ &\quad - \frac{\nu T}{\tau L_m} \frac{\rho_{m,i+1}(k) - \rho_{m,i}(k)}{\rho_{m,i}(k) + \kappa} \end{aligned} \tag{6}$$

where ν is an anticipation constant and κ a numerical stability constant; $V[\rho_{m,i}(k)]$ is the FD and the functional expression used is the following.

$$V[\rho_{m,i}(k)] = v_{f,m} \cdot \exp\left[-\frac{1}{\alpha_m}\left(\frac{\rho_{m,i}(k)}{\rho_{cr,m}}\right)^{\alpha_m}\right] \tag{7}$$

where $\rho_{cr,m}$ is the critical density of link m and α_m a parameter. Parameters $v_{f,m}$, $\rho_{cr,m}$, α_m, define a link's FD and the model validation process aims at identifying their values.

In order to account for speed drops due to on-ramp inflow and merging phenomena the term $-\delta T q_\mu(k) v_{m,1}(k)/(L_m \lambda_m(\rho_{m,1}(k) + \kappa))$ is added at the right hand side of (6), where δ is a constant parameter, μ the merging link and m is the leaving link. This term is included only when the speed equation is applied to the first segment of the downstream link m. Speed decreases due to weaving at locations where the total number of lanes is reduced is accounted for by adding the term $-\phi T \Delta\lambda \rho_{m,N_m}(k) v_{m,N_m}(k)^2/(L_m \lambda_m \rho_{cr,m})$ to the right hand side of (6), where $\Delta\lambda$ is the reduction in the number of lanes and ϕ is another parameter. This term is only applied to the last segment of the link upstream the lane drop.

Constraints are imposed on the mean speed to ensure that it can not be lower than a network-wide minimum speed $v_{\min}$ and on the density to ensure it is not larger than a maximum $\rho_{\max}$. This means that after executing the calculation of eqns. (4) and (6), the following rules are applied

$$\rho_{m,i}(k+1) := \min\{\rho_{m,i}(k+1), \rho_{\max}\} \tag{8}$$

$$v_{m,i}(k+1) := \max\{v_{m,i}(k+1), v_{\min}\} \tag{9}$$

METANET employs a simple queuing model for collecting the demand during period k at origin o, and subsequently forwarding it into the mainstream. This is used for the calculation of the origin link's o outflow $q_o(k)$

into the motorway. However, for the model validation problem, $q_o(k)$ is a direct measurement from the loop detectors. Hence, the queueing model is bypassed, since the measurement is directly fed into the model and there is no need for it.

In order for the speed equation to be applied at every exit location s, the density trajectories $\rho_s(k)$ over the entire time horizon are provided as boundary conditions.

Finally, a node model is used to assign flows at motorway junctions. Let I_n and O_n denote the set of incoming and outgoing links to and from, respectively, node n. Then the sum of all flow entering the node n during time period k, $Q_n(k)$ is given by

$$Q_n(k) = \sum_{\mu \in I_n} q_{\mu,N_\mu}(k) \quad \forall n. \tag{10}$$

The turning rate $\beta_n^m(k)$ is defined as the percentage of $Q_n(k)$ that leaves through out link $m \in O_n$ during period k. This means that $q_{m,0}(k)$ required by eqn. (4) when $i = 1$ is calculated as

$$q_{m,0}(k) = \beta_n^m(k) \cdot Q_n(k) \quad \forall m \in O_n. \tag{11}$$

For a full description of the METANET, see [17] or Appendix A of [36]. By substituting (5), (10), (11) into (4); and the calculation of velocities and densities acting over nodes as in [17, 36] into (6) the model can be expressed as the following discrete dynamic state-space system

$$\mathbf{x}(k+1) = \mathbf{f}\left[\mathbf{x}(k), \mathbf{d}(k); \mathbf{z}\right]. \tag{12}$$

The state vector consists of the density and mean speed of every link segment, i.e.

$$\mathbf{x} = \left[\rho_{1,1}\, v_{1,1} \ldots \rho_{1,N_1}\, v_{1,N_1} \ \ldots \rho_{1,M_1}\, v_{1,1} \ldots \rho_{1,N_{M_1}}\, v_{1,N_{M_1}}\right]^T \tag{13}$$

where M_1 is the number of motorway links in the network.

The disturbance vector $\mathbf{d}$ consists of (a) the inflows q_o entering the system from entry points (origin links) like on-ramps or the upstream main site boundary and optionally the speeds v_o at these locations; (b) the densities ρ_s at the exit locations (destination links) like off-ramps or downstream main site boundary; and (c) the turning rates at every split junction. These quantities are organised into vector

$$\mathbf{d} = \left[q_1\, v_1 \ \ldots q_{M_2}\, v_{M_2} \rho_1 \ldots \rho_{M_3} \beta_1^{\mu_1} \ \beta_{M_4}^{\mu_{M_4}}\right]^T \tag{14}$$

where M_2 is the number of origins, M_3 the number of destinations, M_4 the number of split junctions and $\mu_n \in O_n$ is the index of the destination link sending flow out of the network at split junction node n.

Vector $\mathbf{z}$ consists of the model parameters as encountered in the dynamic density (4), speed (6) and fundamental diagram (7) equations. Hence, $\mathbf{z}$ includes the following network-wide global parameters: the maximum density $\rho_{\max}$; the minimum speed $v_{\min}$; and the mean speed equation (6) parameters τ, ν, ϕ, δ and κ. It also contains parameters related to the fundamental diagram; v_f, α, ρ_{cr}.

3. The Calibration Problem Formulation

From a known initial state $\mathbf{x}_0$ and known disturbance trajectories $\mathbf{d}(k)$, $k = 1, \ldots, K$, a forward integration of (12) results to a full profile of the traffic conditions in the network over the same time horizon. When $\mathbf{x}_0$ and $\mathbf{d}$ are given as measurements, the model output can be compared with measurements taken from locations inside the motorway. The model accuracy in this case depends on the selection of the parameter vector $\mathbf{z}$. A set of measurements $\widehat{\mathbf{x}}$ from a number of locations along the motorway, can be used for comparison between reality and model output. The resulting error minimisation problem takes the form

$$\min_{\mathbf{z}} J\left[\mathbf{x}(k), \widehat{\mathbf{x}}(k)\right] \tag{15}$$

subject to

$$\mathbf{x}(k+1) = \mathbf{f}\left[\mathbf{x}(k), \mathbf{d}(k); \mathbf{z}\right], \ \mathbf{x}(0) = \mathbf{x}_0 \tag{16}$$

$$\mathbf{z}_{\min} \leq \mathbf{z} \leq \mathbf{z}_{\max} \tag{17}$$

where $J\left[\mathbf{x}(k), \widehat{\mathbf{x}}(k)\right]$ is a suitable error function and $\mathbf{z}_{\min}$ and $\mathbf{z}_{\max}$ are the lower and upper bounds, respectively, of $\mathbf{z}$'s elements. Hence, the objective function's dependence on the decision variables $\mathbf{z}$ is implicitly considered through the constraints (16). The evaluation of J for a particular feasible value of $\mathbf{z}$ requires the forward integration of (16) given as input the measured $\mathbf{x}_0$ and $\mathbf{d}(k)$. After this integration, i.e. a simulation run, the model state trajectories are used for obtaining J's value.

As mentioned earlier the parameter vector $\mathbf{z}$ consists of two parts, a set of network-wide parameters and those pertaining to the fundamental diagram. Typically, expert engineering opinion is used for pre-setting the number of distinct fundamental diagrams that are going to be used and the spatial

extension of their application. In other words, prior to the solution of the optimisation (15)–(17), there has to be a decision based on knowledge of the network's congestion dynamics, e.g. bottleneck location, related to the number of FDs to be used and for each one the list of motorway links using them. During the validation process, these two features change only manually and it is based on their specification that problem (15)–(17) is solved. Hence, if $\widehat{N}$ is the number of FDs used, each ones parameters ρ_{cr}, α and v_f need to be included in $\mathbf{z}$, which then takes the form

$$\mathbf{z} = \left[\tau\, \kappa\, \nu\, \rho_{\max}\, v_{\min}\, \delta\, \phi\, v_f^1\, \alpha^1\, \rho_{cr}^1\, \ldots v_f^{\widehat{N}}\, \alpha^{\widehat{N}}\, \rho_{cr}^{\widehat{N}}\right]^T. \tag{18}$$

Notice the superscript number on the FD parameters is the index of the FD and is different from the numbers on the same symbols appearing as subscripts in the modelling section; the latter refer to link and segment indices. When $\widehat{N} = 1$ then a single fundamental diagram is used for the site where the model is calibrated. If $\widehat{N} = M_1$ then every link has its own FD. From a modelling perspective the former is preferred to the latter, since having a unique FD for every link is a case of over-parametrisation. However, the different physical properties of the road network resulting to different aggregate traffic behaviour due to narrower lanes, upgrades, downgrades, changes of curvature, bottlenecks, junction merging and merge lane length, cannot be overlooked. Hence, there is a natural need for using a minimal number of different FDs. In the approach suggested here, these decisions become part of the solution of problem (15)–(17).

This is done firstly by introducing into $\mathbf{z}$ a new FD parameter l^ℓ, which is defined as the number of links FD ℓ is applied to; secondly by using a simple rule for mapping FD onto the list of motorway links. The revised vector of model parameters is now

$$\mathbf{z} = \left[\tau\, \kappa\, \nu\, \rho_{\max}\, v_{\min}\, \delta\, \phi\, v_f^1\, \alpha^1\, \rho_{cr}^1\, l^1 \ldots v_f^{\widehat{N}}\, \alpha^{\widehat{N}}\, \rho_{cr}^{\widehat{N}}\, l^{\widehat{N}}\right]^T \tag{19}$$

where $l^\ell \in [0, M_1 + 1]$, $\ell = 1, \ldots, \widehat{N}$.

The mapping of FD to motorway links for the unidirectional flow sites considered here is a simple rule that assigns the first FD to the most upstream motorway link, which conventionally is assigned the number 1. Subsequently, the remaining FDs are assigned to links in an iterative manner. This rule is applied for a given $\mathbf{z}$ prior to the forward integration of (16) for the evaluation

of the corresponding value of J. Hence, when the METANET executable is invoked, the motorway links have been assigned to FDs.

In order to describe this rule when applied to a particular solution $\mathbf{z}$, let $\mathcal{A}_{\mathbf{z}} = \left\{1, 2, \ldots, \widehat{N}\right\}$ denote the set of FD indices in ascending order. Let $B^{\ell}_{\mathbf{z}}$ and $E^{\ell}_{\mathbf{z}}$ denote the starting and the end link index, respectively for which FD ℓ is applied. Let $\mathcal{F}_{\mathbf{z}} \subseteq \mathcal{A}_{\mathbf{z}}$ denote the ordered set of FD indices with non-zero extension, i.e.

$$\mathcal{F}_{\mathbf{z}} = \left\{\ell \in \mathcal{A}_{\mathbf{z}} : l^{\ell} > 0\right\}. \tag{20}$$

Let us define $D_{\mathcal{F}_{\mathbf{z}}}(j)$ as the mapping that returns the list element at position j in the ordered set $\mathcal{F}_{\mathbf{z}}$. Then the following rule is applied for assigning FDs to motorway links.

$$\begin{array}{l} \text{initialise with} \\ B_{D_{\mathcal{F}_{\mathbf{z}}}(1)} = 1 \\ E_{D_{\mathcal{F}_{\mathbf{z}}}(1)} = l^{D_{\mathcal{F}_{\mathbf{z}}}(1)} \\ \text{and then iteratively for } j = 2, \ldots, |\mathcal{F}_{\mathbf{z}}| \\ B_{D_{\mathcal{F}_{\mathbf{z}}}(j)} = E_{D_{\mathcal{F}_{\mathbf{z}}}(j-1)} + 1 \\ E_{D_{\mathcal{F}_{\mathbf{z}}}(j)} = B_{D_{\mathcal{F}_{\mathbf{z}}}(j)} + l^{D_{\mathcal{F}_{\mathbf{z}}}(j)}. \end{array} \tag{21}$$

Subsequently, the triplets $(\ell, B_{\ell}, E_{\ell})$, $\ell = 1, \ldots, \widehat{N}$ are used to configure METANET's input file describing network configuration and link features (an ASCII file) using a simple script. Because the l^{ℓ} are considered as continuous variables in the optimisation problem, in the final mapping $l^{\ell} := \lfloor l^{\ell} \rfloor$.

The solution of optimization problem (15)–(17), (21) provides an optimal set of model parameters and the optimal number and spatial extension of FD to be used. The model error minimisation as well as the preference towards minimum use of FDs are incorporated into the structure of the objective function (15), which depends on the available loop detector measurements.

Table 1: Variable limits

Variable	τ	κ	ν	$v_{\min}$	$\rho_{\max}$	δ	ϕ	α_m	$v_{f,m}$	$\rho_{cr,m}$
Maximum	60	90	90	8	190	4	3.0	5.00	130	40.0
Minimum	1	5	1	5	160	0.001	0.1	0.40	80	18.0

Let us assume that there are M_5 loop detectors on the motorway providing flow and speed measurements for segment i of link m. The measurements

vector $\widehat{\mathbf{x}}$ to be used in (15) takes the form

$$\widehat{\mathbf{x}} = [y_{1,q}\, y_{1,v} \ldots y_{M_5,q}\, y_{M_5,v}]^T \tag{22}$$

where $y_{j,q}$ and $y_{j,v}$ are the flow and speed measurements, respectively, from sensor j, $j = 1, \ldots, M_5$. A global list is retained assigning each sensor to the corresponding motorway link it belongs to. Hence, if sensor j is installed at segment i_j of link m_j, then the discrepancy between $y_{j,q}(k)$ and $q_{m_j,i_j}(k)$, and $y_{j,v}(k)$ and $v_{m_j,i_j}(k)$ gives a measure of the model's accuracy.

This comparison needs to be performed properly along the time dimension. Typically, the sensors' measurement sample time T_s is larger than the model sample time T. For example, for the data used here, $T_s = 60$ seconds whereas $T = 8$ seconds. The assumption followed is that the real flow and speed during the minute the measurements were taken, are constant. Hence, for each model time period k the measurement sample period it belongs to is identified and the model outputs are compared to the same set of measurements.

Given the measurements $y_{j,q}(k)$ and $y_{j,v}(k)$ the model's flow square error $J_{j,q}(\mathbf{x}, \widehat{\mathbf{x}})$ from sensor j is given by

$$J_{j,q}(\mathbf{x}, \widehat{\mathbf{x}}) = \sum_{k=1}^{K} \left[y_{j,q}(k) - q_{m_j,i_j}(k)\right]^2 \tag{23}$$

and the speed square error from

$$J_{j,v}(\mathbf{x}, \widehat{\mathbf{x}}) = \sum_{k=1}^{K} \left[y_{j,v}(k) - v_{m_j,i_j}(k)\right]^2. \tag{24}$$

Based on them, the weighted total error J_e is

$$J_e(\mathbf{x}, \widehat{\mathbf{x}}) = \sum_{j=1}^{M_4} \left[A_q J_{j,q}(\mathbf{x}, \widehat{\mathbf{x}}) + A_v J_{j,v}(\mathbf{x}, \widehat{\mathbf{x}})\right] \tag{25}$$

where A_q and A_v are scaling factors accounting for the fact that the flow and speed have different orders of magnitude; under stationary conditions, the flow may vary at 5000 veh/h and the speed at 80 km/h. The weights' values used here are $A_q = 0.001$ and $A_v = 1$ and they are those reported in [14] where a similar error function is used. As has been observed in [23],

the flow and speed error terms are not antagonistic in the sense that if the model mean speed is correct then the flow will also be correct by virtue of the density equation, which accounts for the vehicle conservation, hence, the $A_q = 0$ in the objective function in [23].

In order to achieve the automatic assignment of FDs by taking advantage the extension to $\mathbf{z}$ given by (19), the following penalty term $J_p(\mathbf{z})$ is included in the total objective function.

$$J_p(\mathbf{z}) = \sum_{\ell=1}^{\widehat{N}-1} \sum_{r=\ell+1}^{\widehat{N}} \left[w_v \left(v_f^\ell - v_f^r \right)^2 + w_\rho \left(\rho_{cr}^\ell - \rho_{cr}^r \right)^2 + w_\alpha \left(\alpha^\ell - \alpha^r \right)^2 \right] \quad (26)$$

where w_v, w_ρ and w_α are weighting parameters penalising the variance of FD parameters accounting for variable magnitude and they are set to 0.4, 0.5, 10.0, respectively. The problem's objective function (15) takes the form

$$J\left(\mathbf{x}, \widehat{\mathbf{x}}, \mathbf{z}\right) = J_e(\mathbf{x}, \widehat{\mathbf{x}}) + w_p J_p(\mathbf{z}) \quad (27)$$

where w_p a weighting parameter applied on the total penalty term. The weight w_p depends on the problem size and properties, but a default value of 200 is used here, which is appropriate for the size of the sites considered. The inclusion of J_p in (27) aims at driving the minimisation algorithm towards solutions that minimise the distance between FD, hence, resulting implicitly to a minimum number of different FD in the optimal solution. This process results to what is termed here as Automatic Assignment of Fundamental Diagrams (AAFD).

The resulting optimization problem (15)–(17), (21), (27) is not trivial to solve and traditional gradient based approaches cannot be used. Due to their ability to efficiently search the space of solutions and find global minima, evolutionary and swarm intelligence algorithms are particularly suitable for this task, especially since there is no strong real-time computation requirement. There is ample time for the model validation, as once calibrated it remains valid for a sufficient amount of time. The data ageing problem does occur, but at considerably larger time scales compared to the computation time required to converge to a good solution.

The overall system structure is depicted in Fig. 3. A simple interface allows the population based optimisation algorithm to automatically set up all the necessary input files for METANET to run using any vector $\mathbf{z}$ in view of rule (21). The fitness value (27) is calculated based on the output files created by METANET. Notice that in the suggested approach, the

METANET simulator is used as a black box. As a consequence, the discretisation scheme followed does not depend on the sensors' location. Once the algorithm provides a solution, it is verified using data that were not used during calibration.

In total ten different optimisation algorithms are considered and their results are compared here. A brief overview is provided in section 5.

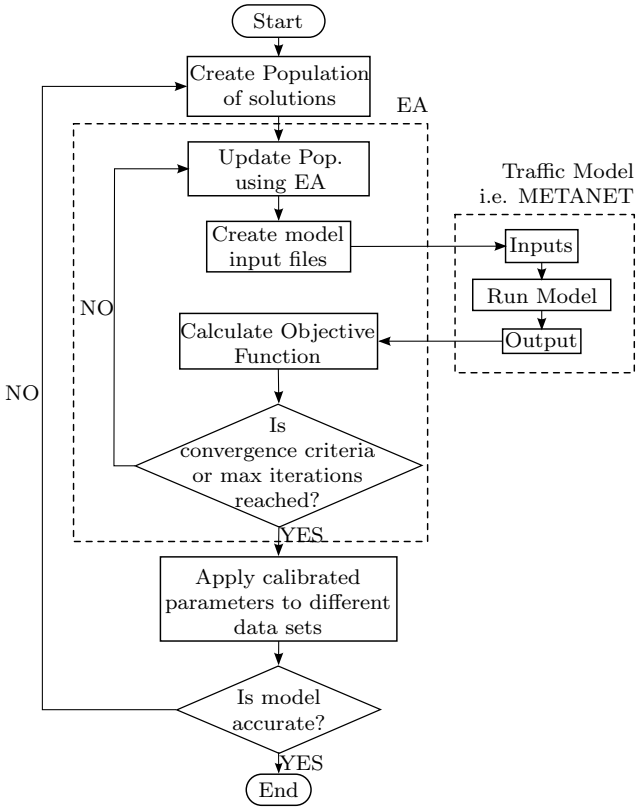


Figure 3: Overall validation system.

4. Site Descriptions and Data Availability

The first site considered here is a motorway stretch near Heathrow Airport and consists of the eastbound M4 linking London to Reading and Bristol. This site serves large traffic volumes due to the connection with London's M25 orbital and parts of Heathrow Airport. The modelled motorway stretch

covers a length of 7.8 km and the METANET model consists of 5 links as shown in Fig. 4.

The model calibration was carried out using data from Monday 8$^{\text{th}}$, 15$^{\text{th}}$ and 22$^{\text{nd}}$ of February 2010. To test the AAFD, the maximum number of FDs allowed to be used is set to be equal to the number of links, i.e. $\widehat{N} = 5$ (the maximum possible) resulting to a 27-dimensional optimisation problem.

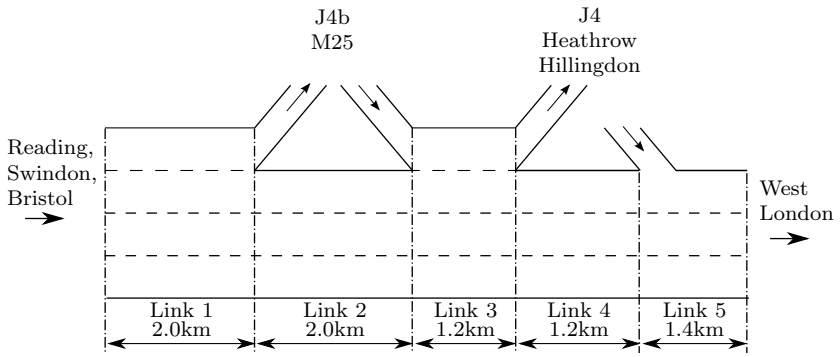


Figure 4: Schematic of Heathrow site.

The second site is the Northbound M1 motorway as it enters Sheffield. It is larger than the Heathrow site extending over 21.9 km and the METANET model consists of 20 links, Fig. 5. Typically, recurrent congestion has the form of a shock wave originating at the centre. Usually it occurs at the end of the link 6 where the off-ramp of Junction 33 is short and unable to cope with the demand of exiting flow, resulting to congestion backing up into the mainstream.

For this site, data was used from Monday the 1$^{\text{st}}$, 8$^{\text{th}}$ and 15$^{\text{th}}$ of June 2009. Again, the number of possible different FDs allowed to be used is set equal to the maximum, i.e. $\widehat{N} = 10$, resulting to a 47-dimensional optimisation problem.

The MIDAS data are collected from loop detectors installed on the motorways and the on-/off-ramps. They provide flow, speed, and occupancy measurements per lane averaged over one minute intervals. It is well known that $v_{m,i}$ in eqn. (5) is the space mean speed, which for a small area centred around a loop detector is estimated by the harmonic mean of individual vehicle speeds passing over the detector. A simple calculation allows the estimation of the space mean speed for the whole cross-section. There is no need for any other transformations like those used in [21].

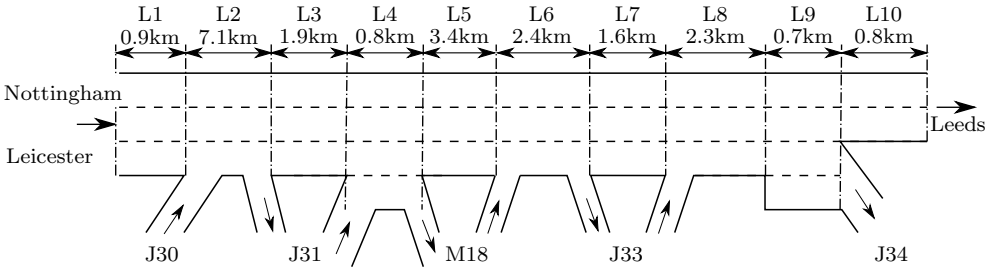


Figure 5: Schematic of Sheffield site.

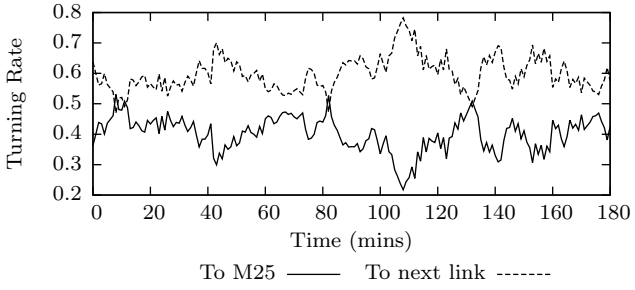
The provided data are used to create the required model inputs, i.e. the initial state vector of eqn. (13) $\mathbf{x}(0)$, $\mathbf{d}$ as defined in eqn. (14) as well as for populating the measurements' vector $\widehat{\mathbf{x}}$ as defined in eqn. (22).

Examples of these data are shown in Fig. 6. The turning rate trajectory for Heathrow's off-ramp at junction J4b. is shown in (a). The density profile at the most downstream site boundary is shown in (b). The inflow and speed profiles, also used as boundary conditions, at the motorway entrance of the Heathrow site are shown in (c). With the exception of the main motorway entrances, the speed is not given for on-ramps as the measured data may be of vehicles that are still accelerating and have not yet adjusted their speed to that of the main carriageway.

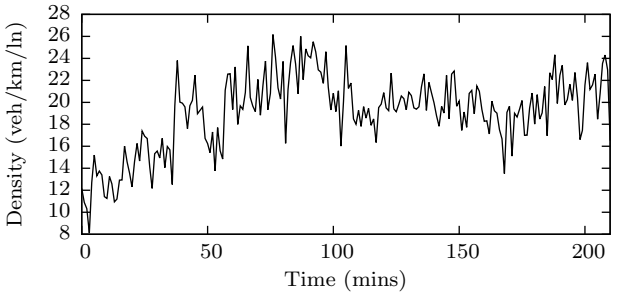
5. Optimisation Algorithms

5.1. The Problem Set-up

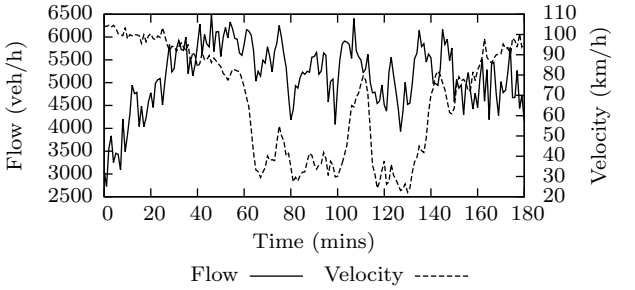
The system designed and developed for this work has been based on the METANET traffic simulator. The optimisation problem formulation in conjunction with population based search methods result in a very flexible and versatile system where the search algorithms and the simulator executable can be replaced with other functionally similar components without much effort, as long as the model parameter vector is defined appropriately. As discussed in section 3 the formulated optimisation problem is concerned with minimising (27) subject to (15)–(17) and (21) by varying the parameter vector $\mathbf{z}$ (19). As shown in Fig. 3, the objective function value calculation is treated as a black box by invoking a call to the simulator executable. All ten algorithms discussed here are based on the same procedure for objective function value evaluation. The search algorithms used adjust $\mathbf{z}$ and extract



(a)



(b)



(c)

Figure 6: Samples of measurement trajectories given to the METANET: (a) Turning rates for Heathrow site J4b. (b) Density at the main end motorway exit of the Sheffield site. (c) Origin input flow and speed at the main entry point of the Heathrow.

$\mathbf{x}(k),\ k = 1, 2 \ldots K$, from the simulator output, and manipulate the population of solutions according to their own specific rules. A brief overview of those algorithms is given next.

5.2. Baseline Genetic Algorithm Optimisation

The nature of optimisation problem dictates the use of derivative-free population based algorithms for solving it. The first choice in this study was the implementation of a simple and straightforward GA based on a real-coding scheme [37], where each member of the population is a parameter vector $\mathbf{z}$. The GA used here is based on [38] and employs three key operators, selection, crossover and mutation.

At every iteration individuals that are far away from the current optimum are disregarded by use of a sigma truncation scheme [39]. In the selection operation solutions are are drawn from this reduced set by use of remainder stochastic sampling without replacement, [39]. This method is used to create a mating pool of equal size to the population. Solutions within the mating pool are paired randomly and passed to the crossover operation.

For each pair of solutions in the mating pool there is probability of crossover occurring. If crossover occurs then the two parent solutions are combined to create child solutions that are passed to the next generation. If crossover does not occur then the parents are passed to the next generation. Various operators have been proposed for the crossover, in this algorithm single point, mask, whole arithmetical and heuristic crossover are investigated, see [38] for details.

Once crossover is complete each solution has a probability to mutate. If a solution mutates it is adjusted in accordance to the uniform or non-uniform mutation operator, [38]. The mutation scheme to be used is selected randomly with the methods weighted equally.

Trials runs were done and the probability of mutation was set at 0.7 and the probability of mutation at 0.05. These values were selected as they allowed for a diverse range of solutions to be maintained without disrupting convergence.

The initial results revealed that the AAFD is possible through the proposed problem formulation. Having verified the idea, a brief investigation was conducted for evaluating different GA crossover operators. In [40] a method is proposed that uses a variety of crossover functions to increase the GA's ability to search the solution space effectively.

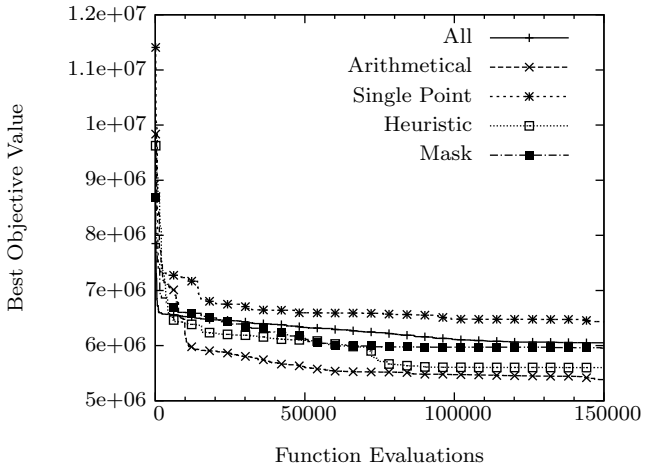


Figure 7: GA crossover operators comparison for Heathrow model calibration.

Figure 7 depicts the evolution of the objective function against the number of function evaluations required by the GA for the Heathrow site for the different crossover schemes as well as a combination of all the investigated operators. The best results were obtained when real coded operators are used, arithmetical and heuristic crossover. Single point and mask crossover perform the worst and a combination of all schemes did not improve convergence. Therefore, further investigations will use a GA using just the arithmetical crossover operator.

5.3. Particle Swarm Optimisation (PSO)

PSO [41, 42] algorithms employ the notion of particles for searching the Γ dimensional solution space $C \subset \mathbb{R}^{\Gamma}$ having knowledge of the best previous solutions found by themselves and by other particles in their neighbourhood. This information is used to update a velocity vector governing each particles' new position resulting to a flock converging towards the current best solution and evaluating the objective function at points along its flight path. An inertia weight [43] allows for the particles' velocity to have a momentum resulting to overshooting over the current optimum and explore more of the solution space.

Each particle ι is characterised by its position $\mathbf{z}_{\iota} \in C$ that defines a solution of the problem, and its velocity $\boldsymbol{\theta}_{\iota} \in \mathbb{R}^{\Gamma}$. At iteration ξ particle ι

is at position $\mathbf{z}_\iota^\xi = [z_{\iota,1}^\xi, z_{\iota,2}^\xi, \ldots, z_{\iota,\Gamma}^\xi]$ within C and has directional velocity $\boldsymbol{\theta}_\iota^\xi = [\theta_{\iota,1}^\xi, \theta_{\iota,2}^\xi, \ldots, \theta_{\iota,\Gamma}^\xi]$. Particle ι's directional velocities and positions are updated according to the rule

$$\theta_{\iota,\gamma}^{\xi+1} = \omega\theta_{\iota,\gamma}^\xi + c_1 r_1 \left[\pi_{\iota,\gamma}^\xi - z_{\iota,\gamma}^\xi\right] + c_2 r_2 \left[z_{h_\iota,\gamma}^\xi - z_{\iota,\gamma}^\xi\right] \tag{28}$$

$$z_{\iota,\gamma}^{\xi+1} = z_{\iota,\gamma}^\xi + \theta_{\iota,\gamma}^{\xi+1} \tag{29}$$

where ω is the inertia weight, c_1, c_2 the acceleration coefficients, r_1, r_2 are random numbers in the domain $[0, 1]$, $\boldsymbol{\pi}_\iota{}^\xi = [\pi_{\iota,1}^\xi, \pi_{\iota,2}^\xi, \ldots, \pi_{\iota,\Gamma}^\xi]$ is the best position previously found by particle ι until iteration ξ and h_ι is the index of the best particle within the neighbourhood of particle ι.

PSO algorithms use different topologies for determining a particle's neighbourhood. In Global-PSO [44] (GPSO) all of the particles communicate with each other, hence h_ι is the best particle within the swarm and returns the same index for all particles ι. Local-PSO (LPSO) [45] uses a ring structure where each particle has two neighbours, for convenience the neighbours are set to those as in the computer memory array list. Hence, h_ι is best current point of particle ι or its two neighbours in memory.

In Adaptive-PSO-09 (APSO-09; the 09 indicates year of publication) [46] the parameters ω, c_1, c_2, which are constant in LPSO and GPSO, are changed in a controlled manner. The change is based on the state of search and iteration number. Fuzzy logic based Evolutionary State Estimation (ESE) is applied to ascertain the state of the search resulting to the appropriate parameters' adjustments. An added feature helping to avoid premature convergence is to have an Elitist Learning Strategy (ELS) mutating the best solution in an attempt to encourage the occurrence of jumping out a local minimum [46].

Two other independent versions of APSO where proposed in subsequent papers, APSO-12 [33] and APSO-14 [32]. In APSO-12 the parameters w, c_1, c_2 are changed following a simple rule, based on current and the maximum number of iterations. A slightly more complex procedure is used in APSO-14, whereby metrics estimating the convergence speed and population diversity are used to adapt the inertia weight w.

The Chaos-Enhanced-Accelerated PSO (CEPSO) algorithm proposed in [47] is also considered here. In this variant, the velocity eqn. (28) is dropped and a revised version of (29) is used. The algorithm's parameters change

over time following a chaotic mapping in order to have a robust convergence strategy. The convergence of this simplified PSO has been proved, although it may suffer in highly non-linear multi-modal problems [48]. Here, the sinusoidal chaotic mapping proposed in [47] is used for updating the position equation parameters.

The final PSO variant investigated is the High Exploration PSO (HEPSO) [49], which combines elements of artificial bee colony and GA. In this case, the inertia weight is adapted in the same manner as APSO-09 and the acceleration coefficients are updated using the same method as APSO-12. The main method for updating the particles' position is based on a GA crossover technique as described in [49]; at the initial stages, however, a particle's position may be updated based on an artificial bee colony algorithm.

5.4. Cuckoo Search

Cuckoo search [50] utilises Lévy flights to update a population of solutions, referred to as nests. Each nest is a version of the parameter vector $\mathbf{z}$ that needs to be identified. At each iteration new solutions (eggs) are created for each nest. An egg is generated by perturbing every dimension of the nest solution by adding a scalar (characteristic step length) multiplied by a random number drawn from the Lévy distribution. If egg is an improvement over the solution at a randomly chosen nest then that nests solution is replaced by the egg, otherwise the egg is discarded. At the end of every iteration a proportion of the worst nests are abandoned and replaced with new ones by a random walk. This simple algorithm has few parameters, the number of nests (population), the proportion of nests to abandon and the characteristic step length. For a full description of CS see [50].

This simplicity of CS may result in an inadequate search of the solution space [35]. In order to address this, a Modified Cuckoo Search (MCS) [35] was proposed, with two adjustments to the original algorithm. First, the characteristic length is now dynamic and reduced over iterations allowing for a gradually refined search. The second modification is to allow for information to be shared between current solutions, which is not present in the original CS. This is done by selecting a fraction of the best nests to form a pool from which new nests are spawned. Two nests are selected from the pool and a new nest is generated along the line connecting them. The nest is placed on this line using the inverse of the golden ratio $(1+\sqrt{5})/2$ so that it is closer to the fitter solution. The new nest replaces the current worst nest in the search. For a full overview of the MCS see [35].

5.5. Algorithm Parameters

All algorithms were initialised using Latin hypercubes. Each variable is assumed to have a uniform distribution within its range, as defined in Table 1. The limits in the table are simply handled by moving solutions that move outside the domain to the boundary. For the PSO algorithms the velocity component for dimensions that are adjusted in this manner is multiplied by -0.5. This prevents the solution from attempting to leave the space on the next iteration. For the PSO and GA optimisation algorithms the population size was set to 30. For CS and MCS a population of 25 was used as this is what is used in [50]. The GA parameters were previously discussed and the probability of crossover is set at 0.7 and the mutation probability at 0.05. In CS and MCS the percentage of nests to be abandoned in each iteration was set equal to 25% and 75%, as suggested in [34] and [35]. For CS the characteristic step length was set to 1/100 of the variable search range as shown in Table 1 as it is in [50]. For MCS the characteristic length is the same as proposed in [35]. The parameters for the various PSOs is typically inherited from the papers they were proposed in, and shown in Table 2. In the table ξ is the search iteration number and Ξ is the maximum number of iterations which was set at 5000. f is defined in [46, 49] is the evolutionary factor that defines the spread of particles relative to the current optimum. For CEPSO the sinusoidal map which performed best in the benchmarks in [47] is used for the chaotic map. The probabilities for the artificial bee colony and GA operators used by HEPSO are the same as those in [49]. For GPSO and LPSO better results were obtained by using the parameter values suggested in [51] rather than the typically used $c_1 = c_2 = 2.0$.

Table 2: PSO parameters

Variant	c_1	c_2	ω	Reference
APSO-09	ESE	ESE	$\frac{1}{1+1.5e^{-2.6f}}$	[46]
APSO-12	$(1.2-2.8)\frac{\xi}{\Xi}+2.8$	$(2.8-1.2)\frac{\xi}{\Xi}+1.2$	$(0.75-0.35)e^{0.001\xi}+0.35$	[33]
APSO-14	2.0	2.0	Based on solution diversity	[32]
HEPSO	$(0.5-2.5)\frac{\xi}{\Xi}+2.5$	$(2.5-0.5)\frac{\xi}{\Xi}+0.5$	$\frac{1}{1+1.5e^{-2.6f}}$	[49]
CEPSO-12	N/A	N/A	N/A	[47]
GPSO	$\frac{1}{2}+\ln(2)$	$\frac{1}{2}+\ln(2)$	$\frac{1}{2\ln(2)}$	[51]
LPSO	$\frac{1}{2}+\ln(2)$	$\frac{1}{2}+\ln(2)$	$\frac{1}{2\ln(2)}$	[51]

Table 3: Summary of best objective value for each algorithm (H stands for Heathrow and S for Sheffield)

	H 8^{th}	H 15^{th}	H 22^{nd}	S 1^{st}	S 8^{th}	S 15^{th}
GA	5384538	5858308	6739297	6595545	7024323	7619637
GPSO	3792653	3142996	2466972	4387156	5644626	3913927
LPSO	3785607	**1997553**	1893783	3956224	**4399481**	**3902504**
APSO-09	5108443	2613920	**1882825**	4530514	5542426	4414949
APSO-12	6541304	2934669	3478906	4841882	7178995	5257622
APSO-14	4357332	5823782	4222196	4366692	7530088	5295245
HEPSO	**2585451**	2389230	1928796	**3797889**	4453520	4061613
CEPSO	3801993	3215278	2092989	4567079	5299468	4881406
CS	5733080	3806055	6468042	6273483	6293645	6649381
MCS	5440922	3988003	6483786	7278220	6840414	7581395

6. Results

6.1. Algorithmic Performance

As mentioned in section 5, ten different algorithms have been tested for solving optimisation problem (15)–(17), (21), (27) aiming at calibrating the METANET simulator for the Heathrow and Sheffield sites. Although this paper aims at highlighting the capability of automatically assigning fundamental diagrams along the site by solving the optimisation problem formulated, an important aspect of the approach is the search algorithm's performance. Selecting an algorithm to use and setting it up optimising its behaviour is a whole different task altogether. Here, a comparison is performed for a number of algorithms with the parameters given in the previous section.

Table 3 provides the best objective function values (including the penalty terms) found when performing the optimisation three times from randomly selected initial populations. The table contains results from both sites and all calibration dates. For brevity the table columns are labelled as H*date* and S*date* for Heathrow and Sheffield, respectively; thus, H8^{th} refers to the data of the Heathrow site collected on the 8^{th} of February 2010 whereas S8^{th} refers to the data collected from the Sheffield site on the 8^{th} of June 2009. Each row gives the minimum found by each of the ten algorithms using the data from the corresponding site and date indicated by the column.

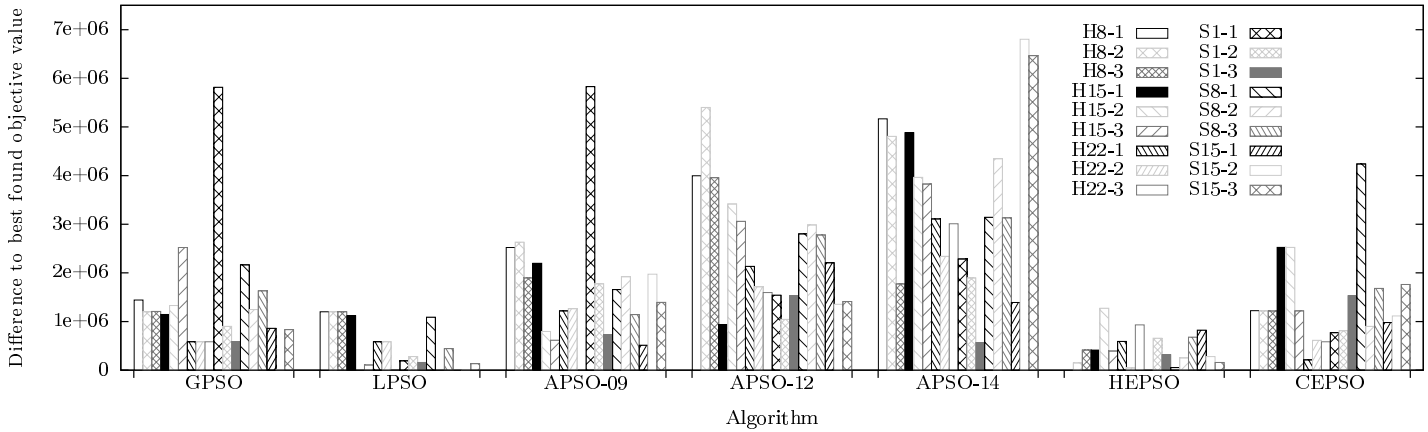
It can be seen that the PSO family of algorithms outperforms both the simple GA and the two CS variants. Comparing CS and GA, CS outperforms the latter for the problem generated using the H15^{th} and H22^{nd} data. The GA solution is superior only for the H8^{th} set of data. For the Sheffield site, CS consistently performs better than both GA and MCS. Interestingly, the

unmodified CS performs better than the MCS for all Sheffield and for two of the three Heathrow sets of data. Hence, it can be concluded that the original CS is more efficient from the modified one and the baseline GA.

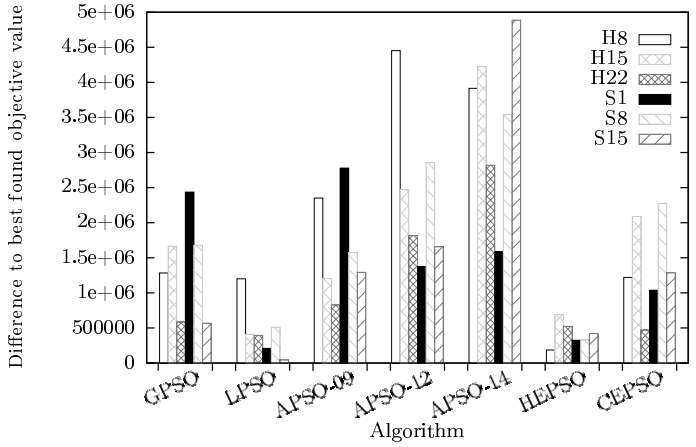
The PSO family of algorithms is generally the most efficient one. Comparing with the GA, the worst performance by any PSO algorithm is that of APSO-12 for the H8$^{\text{th}}$ and S8$^{\text{th}}$ sets of data, where the solution is 21.48% and 7.2%, respectively, worse than the corresponding GA solution. For the rest data sets, the worst PSO outperforms the GA, with a marginal improvement for the H15$^{\text{th}}$. For the rest, the PSOs show at least a 25% improvement over the GA. Hence, it is clear that the PSO algorithms provide the best performance.

Narrowing our discussion around PSO, Table 4 provides the full set of results from all repeated runs for the PSO algorithms. These data are represented in Fig. 8, where all results are normalised by subtracting the best objective function value achieved for the corresponding dataset (the highlighted values in Table 3). In Fig. 8 the same convention for labelling is used as in Table 4 with an extension to include the run number; e.g. H*date-run* refers to the Heathrow calibration of *date* and the repeat number is *run*, and so on. Each bar in Fig. 8 represents the difference to the best objective function value obtained by any algorithm for that dataset. The bars are grouped by algorithm, with each group containing 18 bars (3 datasets from 2 sites with 3 repeated runs); a gap (bar of zero height) represents the best solution found for that particular dataset. Different datasets and sites are represented in Fig. 8 by the bar style. The best results in Fig. 8 are represented by small bars as these are closest to the best objective value found.

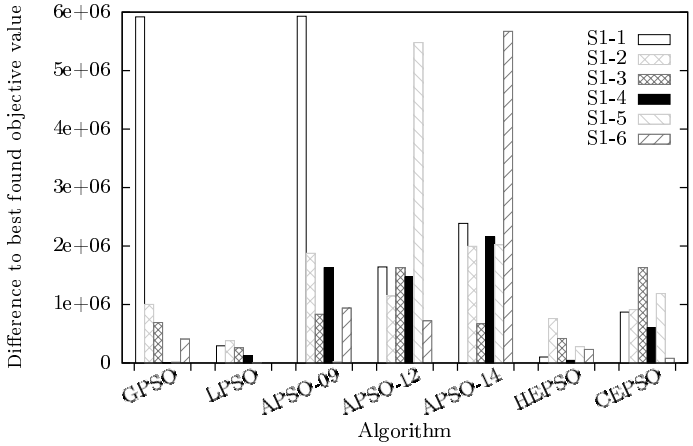
Fig. 8a and Table 4 indicate that the worst performing PSO variant is APSO-14. Apart from a single reasonable result found using S1 in the third run, which is close to solutions found by the other variants, there are substantial differences to the rest. APSO-12 performs slightly better for this problem but is the second worst of the PSOs tested. GPSO, APSO-09 and CEPSO show similar performance characteristics. On occasion these algorithms find very good solutions that are close to the best optimum found. In fact APSO-09 found the optimal solution for the Heathrow site for the dataset of the 22$^{\text{nd}}$ (H22-3). However, these algorithms lack consistency. The standard deviation of the bar heights in Fig 8a for GPSO, APSO-09 and CEPSO are 1.74e+06, 1.78e+06, and 1.53e+06, respectively. This compares to 1.01e+06 for LPSO and 0.99e+06 for HEPSO. Not only are LPSO and HEPSO the most consistent optimisers tested but they also find the



(a) Difference to best objective value found by any calibrator for every dataset.



(b) Average difference to best objective value found for each dataset.



(c) Difference to best objective value found for S1st.

Figure 8: Algorithm performance data.

Table 4: Objective function values for each run of the PSO algorithms: (a) Heathrow site (b) Sheffield site

(a)

	H 8^{th}			H 15^{th}			H 22^{nd}		
GPSO	4028385	3785726	3792653	3142996	3327233	4519245	2467040	2466881	2466972
LPSO	3785607	3785613	3785625	3115172	1997553	2104353	2466580	2466638	1893783
APSO-09	5108443	5218619	4481143	4195910	2793755	2613920	3100260	3145655	1882825
APSO-12	6581623	7987618	6541304	2934669	5412505	5058233	4017077	3596675	3478906
APSO-14	7753486	7392129	4357332	6879063	5962253	5823782	4990201	4222196	4891807
HEPSO	2585451	2732771	2997313	2409576	3270954	2389230	2474337	1928796	2814655
CEPSO	3807770	3804409	3801993	4520900	4521772	3215278	2092989	2496649	2467525

(b)

	S 1^{st}			S 8^{th}			S 15^{th}		
GPSO	9615250	4697883	4387156	6564409	5644626	6029178	4762856	3913927	4735089
LPSO	3990011	4077664	3956122	5488522	4399481	4843245	3902504	3904656	4034818
APSO-09	9627157	5572121	4530514	6056519	6320857	5542426	4414950	5874789	5296917
APSO-12	5339753	4841882	5325566	7201900	7385578	7178995	6109701	5257622	5313983
APSO-14	6084220	5692574	4366692	7544100	8745415	7530088	5295245	10706332	10368168
HEPSO	3797889	4455714	4115981	4453520	4651302	5078914	4726428	4181375	4061613
CEPSO	4567079	4609269	5329183	8637544	5299468	6081886	4881406	5017779	5664702

best solutions. Apart from the previously discussed H22 (optimum found by APSO-09) the best solution for all the remaining datasets is found by either one of these algorithms. Even for the H22 dataset LPSO finds a solution that is only 0.01e+06 worse than the optimal found by APSO-09.

Figure 8b, depicts the average best point achieved for the three runs, normalised as in Fig. 8a where each bar now represents a single dataset. Again, it can be seen that APSO-12 and APSO-14 are the worst performing PSOs and LPSO and HEPSO are still the best. Another interesting point from Fig. 8b is the performance of HEPSO for the H8 dataset. LPSO, GPSO and CEPSO all found similar objective function values on average for this dataset but HEPSO was able to find better solutions. Apart from this dataset, LPSO and HEPSO show a similar performance.

Figs. 8a and 8b provide a general overview of the PSO algorithms across all datasets and sites. To further assess their performance, the S1 dataset, which shows good generalisation properties (see section 6.3), was used to run three more randomly initialised searches. The extra results obtained for each algorithm combined with those from the initial three runs are shown in and Fig. 8c. Figure 8c shows the result of the six repeated runs normalised as before to the best solution found. LPSO and HEPSO are still the most con-

sistent and best performing algorithms. The best result overall was achieved by LPSO but similar objective values were obtained by GPSO and APSO-09, showing that these two algorithms are good candidates worth considering for solving the model validation problem.

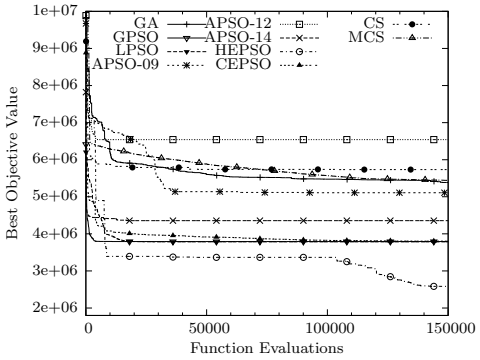
Figures 9 and 10 depict the convergence profiles, i.e. the optimum value over the number of function evaluations, with a function evaluation requiring a separate METANET model run. Function evaluations were explicitly counted as calls to the model to obtain a more comparable set of convergence results, see also [52]. After a short initial period of rapid decrease the GA, CS and MCS settle on a bad minimum, whereas the PSO algorithms bundle around smaller values.

LPSO's behaviour is consistent in the sense that it tends to settle to the minimum quite early on and then improve it. HEPSO manages to find similar points but displays slower convergence. PSO algorithms have shown superior behaviour to GA and CS for this particular problem. The best performing PSO variants are LPSO and HEPSO. The rest of the more recent modifications suggested in the literature of PSO do not provide strong evidence of substantial improvement over the earlier LPSO variant.

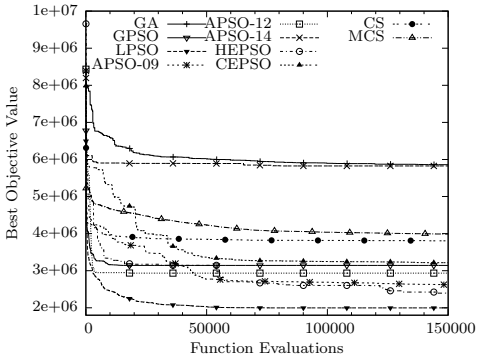
The results shown in Table 3 and Figure 8 are somewhat surprising, since it is the basic variant of the PSO using local information that performs best in terms of the calibration problem. The modifications suggested for the more recent variants are not able to provide the expected level of performance, with the exception of HEPSO. This observation calls for a more focused investigation on the efficiency of the more recent algorithms. Such an investigation would aim at finding ways to set them up and select parameters, which improve convergence properties for the problem formulated here. For a more detailed discussion on the issue of comparing different population-based derivative-free stochastic search algorithms, see [53, 54]. Developing a procedure for selecting and setting up the most appropriate optimisation algorithm is beyond the scope of this paper. It would prove, though, a very important system component in a traffic control centre's information infrastructure. Since our emphasis is on the solution properties in terms of the traffic flow modelling problem, the site specific optimal solutions achieved are discussed next.

6.2. Heathrow Site

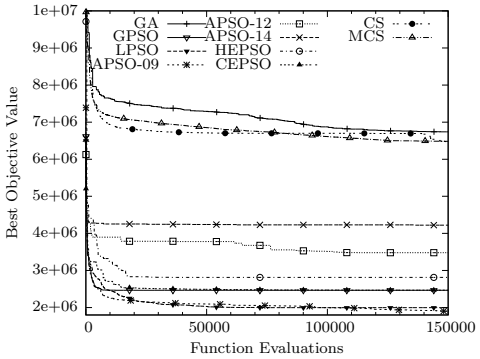
Focusing on the results obtained for the Heathrow site, Table 5 gives the optimal values of the global model parameters and Table 6 the used FD



(a)

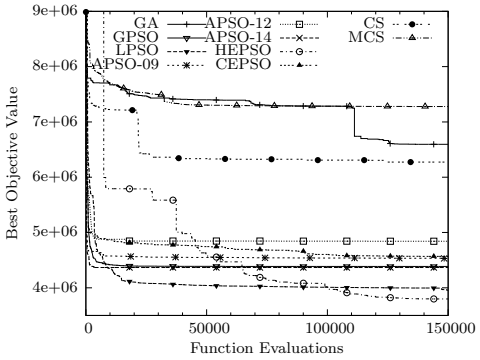


(b)

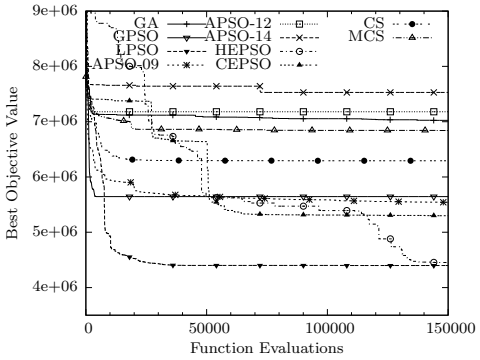


(c)

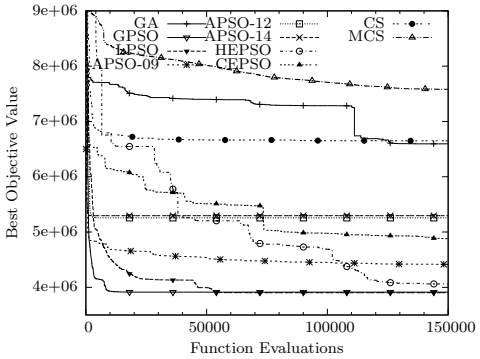
Figure 9: Convergence profiles for calibration of Heathrow model: (a) H8$^{\text{th}}$ (b) H15$^{\text{th}}$ (c) H22$^{\text{nd}}$.



(a)



(b)



(c)

Figure 10: Convergence profiles for calibration of Sheffield model: (a) S1st (b) S8th (c) S15th.

Table 5: Optimal solutions found for Heathrow site model calibration.

	τ	κ	ν	$v_{\min}$	$\rho_{\max}$	δ	ϕ
8^{th}	39.74	5.12	70.26	7.56	160.00	1.06	0.01
15^{th}	14.42	5.04	43.49	8.00	161.77	4.00	0.17
22^{nd}	31.078	5.00	48.59	6.39	189.75	0.002	0.44

Table 6: Heathrow optimal solutions of FD parameters.

	ρ_{cr}	v_f	α	Start link	End link
8^{th}					
FD 1	19.65	126.42	1.5192	1	1
FD 2	34.36	126.36	1.4474	2	4
FD 3	20.76	122.35	2.2059	5	5
15^{th}					
FD 1	22.26	122.65	1.4733	1	1
FD 2	31.96	125.17	1.4244	2	4
FD 3	23.75	111.87	1.8305	5	5
22^{nd}					
FD 1	21.79	129.72	1.2579	1	1
FD 2	30.07	129.20	1.3122	2	4
FD 3	20.06	126.33	1.1600	5	5

parameters and spatial extension. The calibration process identified for each of the three days sets of data that three different FD should be used. These are shown in Fig. 11, where it can be seen that they are arranged into two bundles. The motorway capacity pattern identified in all three cases is that of a low-high-low level, with the area of links 2–4 having high capacity.

The extension of the FD is also the same for all solutions. The reason for this kind of assignment is the fact that link 1 (see Fig. 4) experiences a spill back effect from junction's J4b off-ramp. Congestion spilling back from an off-ramp, either due to reduced capacity or traffic lights at the surface network further downstream, results in an aggregate local behaviour different from the rest of the motorway, hence the need for a dedicated FD, covering the affected area. The critical density of this FD is significantly lower than that of links 2–4 FD, reflecting the different and more cautious behaviour of drivers leaving the system via J4b off-ramp. The spill back effect is interpreted by the optimisation as a reduced capacity area, although geometrically the road is fairly similar.

The second region's FD covering a length of 4.4 km (links 2–4) has higher critical density reflecting the fact that the flow continuing past the first off-ramp is no longer impeded by any spill back.

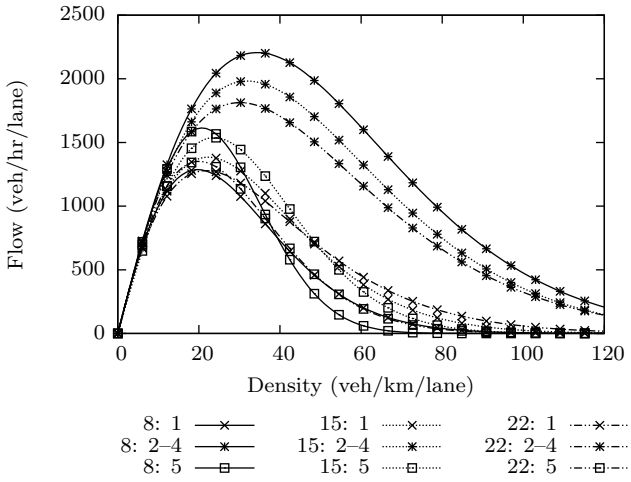


Figure 11: Optimal FD for the Heathrow site model; in the legend, the first number indicates the date of data and the second the link number it applies to.

The final link's FD has also small capacity following the trend of the first FD. At the end of this final link the main boundary condition of the motorway exit is applied, which is a main influence on the system behaviour. However, there is no congestion there during the entire time horizon, hence the full range of possible traffic conditions are not provided as local information to the optimisation problem. Furthermore, link 5 has an on-ramp at its beginning, which means there are inflows to the mainstream and therefore a reduction of the mean speed due to merging. This should be viewed in conjunction with the use of the merging term added explicitly to the speed equation (6). The optimisation just considers the measurements and converges to a solution where a change in the FD parameters for reproducing the drop and recovery of speed is preferred to a solution that changes one of the global model parameters, i.e. δ. This can be seen from the solutions shown in Tables 5 and 6. For the two data sets H8$^{\text{th}}$ and H22$^{\text{nd}}$ where the optimal critical density of link 5 FD is about 20 veh/km/lane, the optimal δ has a low value; for the solution based on H15$^{\text{th}}$ the critical density is 23.75 veh/km/lane but the drop of speed is compensated by assigning δ the maximum value allowed.

The overall result, however, is the desired one, as shown in Fig. 12 where the model flow and speed trajectories are compared to the corresponding MIDAS measurements. The model outputs are obtained by using the optimal

Table 7: Heathrow verification total square error.

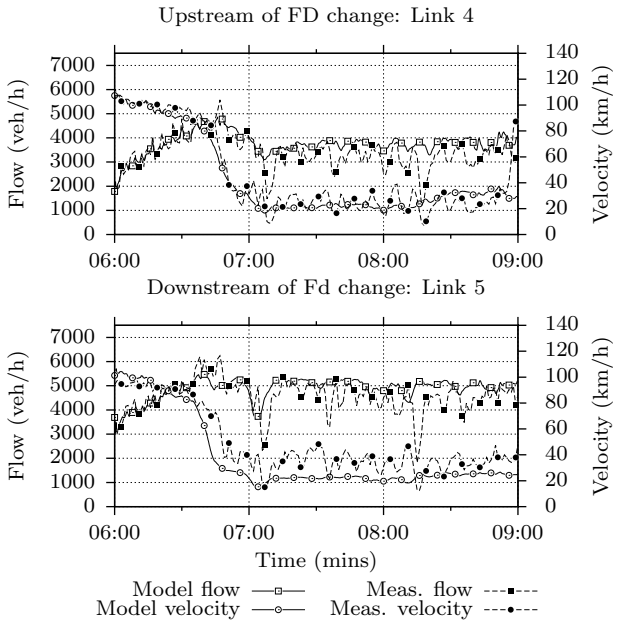
	8	15	22
Calibrated 8	2539830	3060580	4057189
Calibrated 15	7404680	1967917	3080164
Calibrated 22	6614281	3517627	1867561
22 adjusted	4023508	2395942	2421725

parameter set determined by either HEPSO or LPSO as indicated in Table 3. Analysis is based on the results of these algorithms as they show the most consistent behaviour. Fig. 12 depicts the calibrated model output for the data set used to calibrate it.

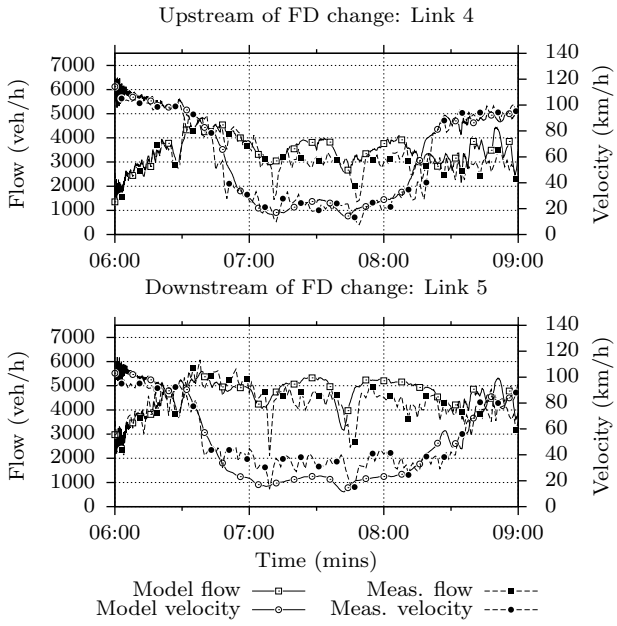
The flow and speed trajectories shown are from the last segment of link 4 and the first segment of link 5, where there is a change in the FD. When the optimal parameter sets are used, the model is able to capture accurately the dynamics of congestion in terms of predicting its onset and the subsequent speed recovery. The resulting total square error, i.e. the values of the objective function without the penalty terms, for the calibrated models are given by the diagonal elements of Table 7.

The off-diagonal elements of the table are the total square error when the optimal parameter set determined using the data from the date indicated at the row (found by the corresponding algorithm mentioned in Table 3) is applied to the METANET model of Heathrow using input boundary conditions from the date indicated by the columns. In other words, Table 7 provides a measure of how the calibrated model based on a particular data set is generalised to the rest of the available data. Reading the table row-wise the quality of the corresponding optimal parameter set can be evaluated.

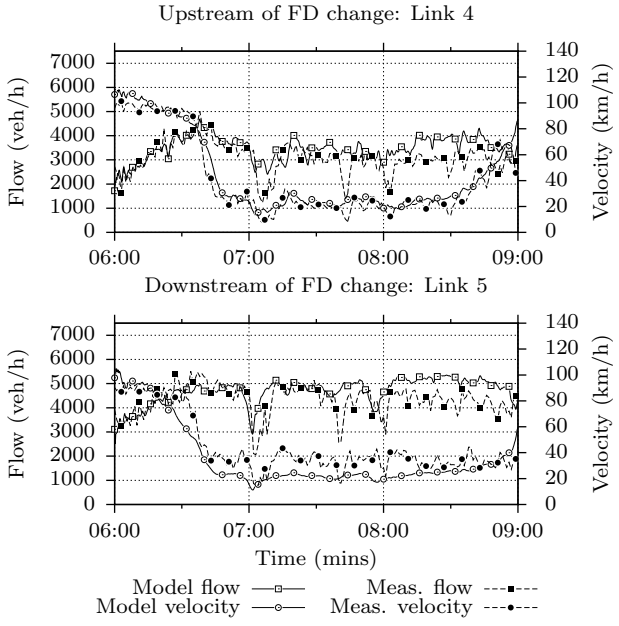
It can be seen there that best and most consistent set of parameters is the one identified by HEPSO with the calibration running with data from H8$^{\text{th}}$. The second best in that respect is the solution based on H22$^{\text{nd}}$. Figure 13 compares the model flow and speed trajectories to measurements when the model inputs are from H15$^{\text{th}}$ and the model parameters used are the optimal solutions identified using data sets H8$^{\text{th}}$ and H22$^{\text{nd}}$. These should also be compared with Fig. 12(b), i.e. the calibration case of H15$^{\text{th}}$. As expected, the METANET model with model parameters based on H15$^{\text{th}}$ is the best fit, since it is calibrated with them, however, performance of the other two parameter sets is quite good and the main elements of congestion dynamics are captured, as can be visually confirmed from Fig. 13. This is also reflected



(a)



(b)



(c)

Figure 12: Sample of time profiles for the calibrated Heathrow model: (a) H8$^{\text{th}}$ (b) H15$^{\text{th}}$ (c) H22$^{\text{nd}}$.

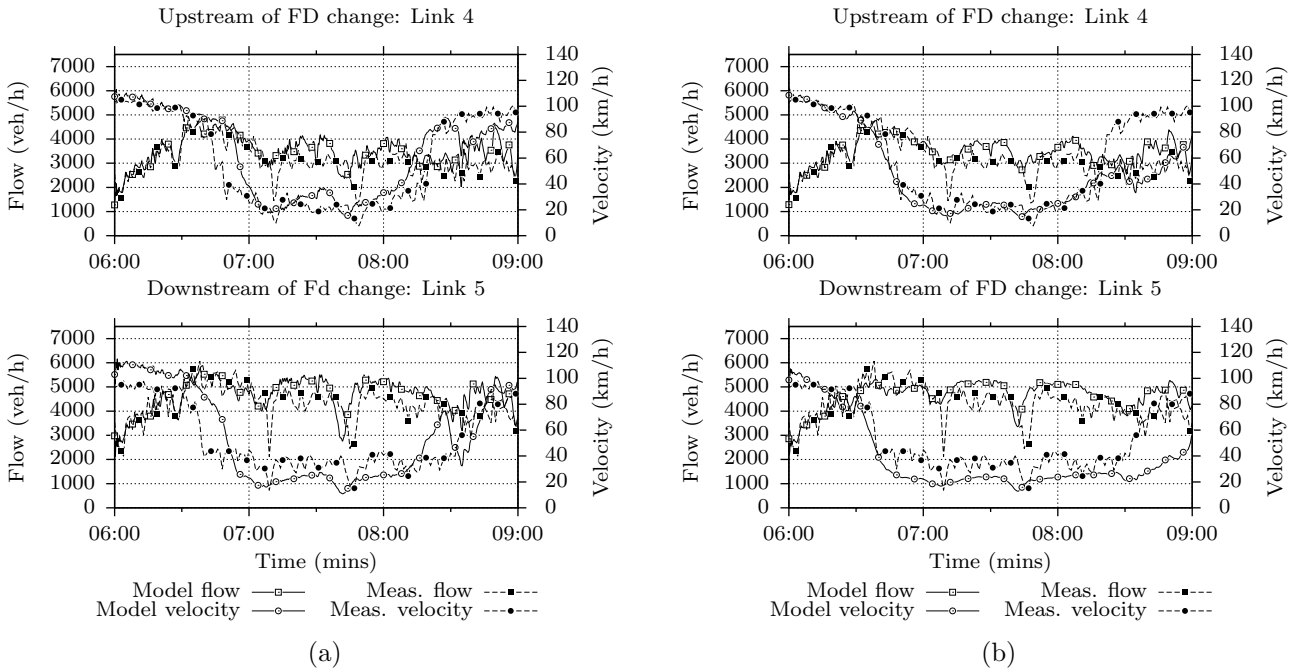


Figure 13: Verification of the Heathrow model using input data from set H15$^{\text{th}}$ and using the optimal parameter set based on (a) the H8$^{\text{th}}$ data set and (b) the H22$^{\text{nd}}$ data set.

in the corresponding error values in Table 7.

A further issue Table 7 raises is the fact that the best calibration result does not provide the most general solution. The solution identified by LPSO using the H22$^{\text{nd}}$ data set for calibration is outperformed by the solution found by HEPSO using the H8$^{\text{th}}$ data set. The H22$^{\text{nd}}$ based optimal parameter set provides a good accuracy, with the exception of the latency of recovering the speed at links 4 and 5 at the last half hour of the time horizon, as shown in Fig. 13(b).

Investigating further, the critical density of link 5 in the parameter set of the H22$^{\text{nd}}$ was manually increased from 20.06 to 20.56 veh/km/lane. The outcome of this adjustment is summarised at the last row of Table 7. When the new parameter set is again applied to the three data sets the adjusted solution is more relevant. This improvement can also be seen in Fig. 14, where now the speed recovery in link 5 is restored for the last half hour. This result demonstrates the increased model sensitivity with respect to the critical density, something which is reported in the literature as well, [18]. It

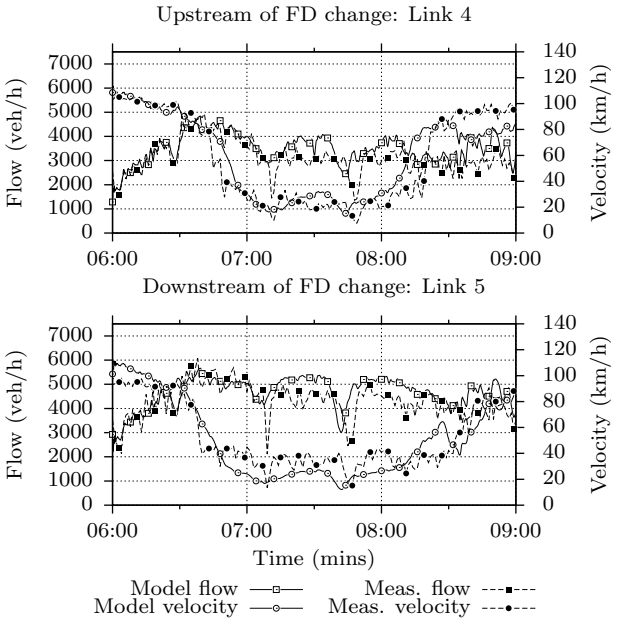


Figure 14: Verification of Heathrow 15$^{\text{th}}$ using manually adjusted model from the 22$^{\text{nd}}$

also shows that there is scope for revising the form of the objective function (27) to an expression that considers the generalisation property of a solution more explicitly, something that is noted as future research. The current method calibrates a specific data set so it is possible as seen by the manual adjustment here to compromise the models accuracy to the calibration data for increased generality. Finally, it is interesting to note that it is HEPSO that provides better quality solutions, rather than LPSO.

Part of future work is the development of a method for averaging and fine tuning the resulting optimal parameter sets based on different calibration data sets, considering the model's sensitivity to its parameters.

6.3. Sheffield Site

The optimal parameter sets identified from the calibration process using the three data sets collected from the Sheffield site are given in Tables 8 and 9. The more extended Sheffield site allows for a larger variety of FD assignments. Out of the possible ten different FDs that can be used, the optimisation algorithms converge to solutions that make use of only three. However, their spatial extension is different. As can be seen from Table 9

Table 8: Optimal solutions found for Sheffield site model calibration.

	τ	κ	ν	$v_{\min}$	$\rho_{\max}$	δ	ϕ
1^{st}	32.34	28.63	55.90	7.48	183.13	0.844	0.038
8^{th}	10.22	20.06	20.00	7.99	184.24	0.001	0.360
15^{th}	19.09	5.89	23.39	8.00	173.38	0.001	0.122

there is always a separate FD for the final link 10. The other two FDs are applied from link 1 to 7 and 8 to 9 for the calibration solutions based on $S1^{st}$ and $S15^{th}$; the solution based on $S8^{th}$ assigns a single FD from link 1 to 3 and then a different one from link 4 to 9. The resulting optimal FDs are shown in Figure 15. It can be seen that the site is split into high and low capacity areas. However, the calibration results follow different patterns. Based on $S1^{st}$ data the site split follows a low-high-low pattern; using $S8^{th}$ data results to a high-low-high pattern whereas $S15^{th}$ to a low-high-high.

This is a problematic result, since there is a fundamental disagreement between the capacity patterns emerging from the $S8^{th}$ data set and the other two. The difference between the capacity pattern in the results based on $S1^{st}$ and $S15^{th}$ is last link's capacity. Just as in the case of Heathrow, there is no congestion in the measured speed trajectories at link 10, hence, the discrepancy is attributed to the lack of information of the full spectrum of traffic conditions, critical and congested.

The low-high-low and low-high-high capacity patterns are not very different as suggested by Table 10, where the total square error is shown for all the calibration and verification cases. The calibration error for the $S1^{st}$ and $S15^{th}$ is comparable and when the calibrated parameters based on $S1^{st}$ are applied using $S15^{th}$ input data and vice-versa, the results are again similar. This is not the case for the $S8^{th}$ data set, which generally shows larger error. The results based on $S8^{th}$ are not consistent with the rest of the data and do not generalise well.

Fig. 16 provides samples of the calibration results comparing model outputs with measurements. It allows for the accuracy of the solutions reported in Tables 8 and 9 to be appreciated, including the good behaviour of the solution based on $S8^{th}$. However, the verification results shown in Fig. 17 show its poor quality generalisation.

Fig. 17 shows the model output and corresponding measurements, when $S15^{th}$ is used for providing the model's input data and the optimal solutions obtained from $S1^{st}$ and $S8^{th}$ are used as model parameters. The model with

Table 9: Sheffield optimal solutions for FD parameters.

	ρ_{cr}	v_f	α	Start link	End link
1^{st}					
FD 1	27.23	122.36	2.6760	1	7
FD 2	30.19	105.35	2.3494	8	9
FD 3	26.68	109.75	1.1386	10	10
8^{th}					
FD 1	31.53	122.22	2.5587	1	3
FD 2	28.50	113.77	1.8865	4	9
FD 3	38.02	104.72	1.0724	10	10
15^{th}					
FD 1	28.43	115.96	2.1077	1	7
FD 2	31.88	103.43	2.0904	8	9
FD 3	35.17	104.73	1.1459	10	10

Table 10: Sheffield verification total square error.

	$S1^{st}$	$S8^{th}$	$S15^{th}$
Calibrated 1	3782550	9741511	7764030
Calibrated 8	14388131	4382314	20064014
Calibrated 15	7578581	10574672	3858505

the parameter set based on $S1^{st}$ is able to correctly reproduce the traffic conditions of the 15^{th}. This is not true when the optimal parameter set based on $S8^{th}$ is used by the model. The mean speed shown in Fig. 17(b) does not follow measurements in a satisfactory way. This result shows the difficult nature of the calibration problem. It is difficult, if not impossible, in view of the noisy nature of the available measurements, to formulate an optimisation problem that does not suffer from local minima. Hence, a learning mechanism, specific to the macroscopic traffic flow modelling context, needs to be in place in order to appreciate the suitability of data sets and relevance of solutions identified.

As a final comment, it should be noted that it is the solution provided by HEPSO, using the $S1^{st}$ data set, just as in the case of the Heathrow site provides the best generalised model. This calls for further investigation researching this particular algorithm's behaviour for the specific calibration problem.

7. Conclusions and Future Work

This paper has presented a study on the application of population based optimisation methods to the problem of macroscopic traffic flow model cali-

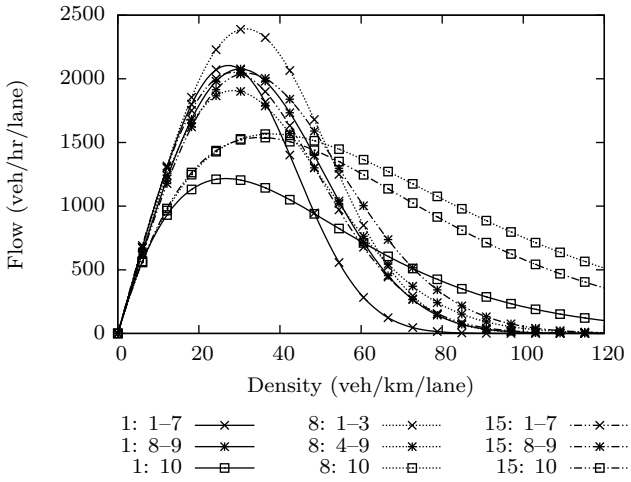
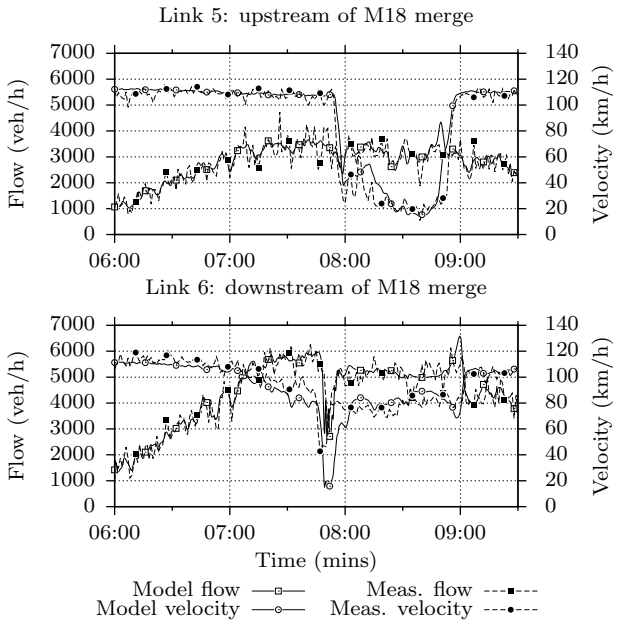


Figure 15: Optimal FDs for the Sheffield site model; in the legend, the first number indicates the date of data and the second the link number it applies to.

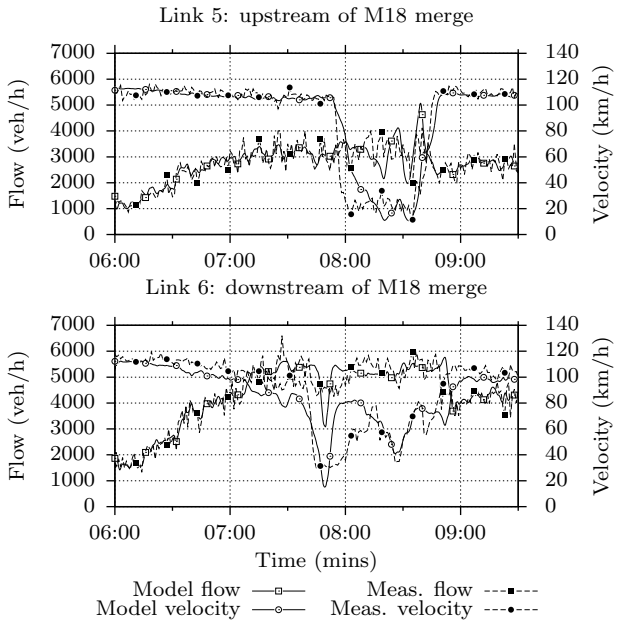
bration and verification. The suggested problem formulation treats the traffic simulator as a black box and therefore the METANET program used can easily be replaced by another executable. The network's spatial and temporal discretisation is model dependent and does not depend on the location of the sensors. A simple geographic mapping is sufficient for the followed approach to deliver excellent solutions given an appropriate algorithm.

Furthermore, the results demonstrate the feasibility of achieving AAFD through the suggested problem formulation. This way, expert engineering opinion is removed from the calibration effort. It has been shown that the derived solutions are able to detect the capacity pattern of a particular network, although some cautious analysis needs to take place. These benefits have become possible to be realised thanks to the nature of the optimisation algorithms used.

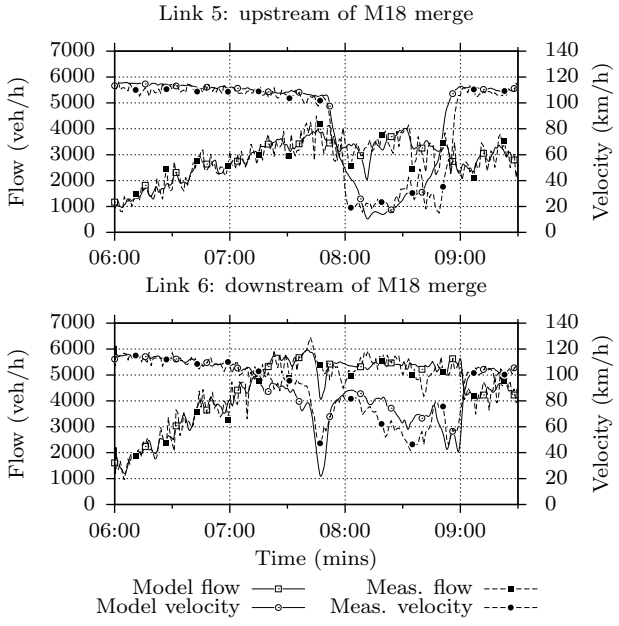
Ten different algorithms have been implemented and evaluated in this paper. A simple GA acting as baseline, two Cuckoo Search variants and seven PSO algorithms. It is clear that the PSO family outperforms the rest. Surprisingly, one of the most efficient PSO algorithms is LPSO, which manages to converge to very good solutions for the stand alone calibration problem faster than more recently proposed PSO variants. However, it is the solutions provided by HEPSO that generalise better for the particular



(a)



(b)



(c)

Figure 16: Sample of time profiles for the calibrated Sheffield model: (a) S1st (b) S8th (c) S15th.

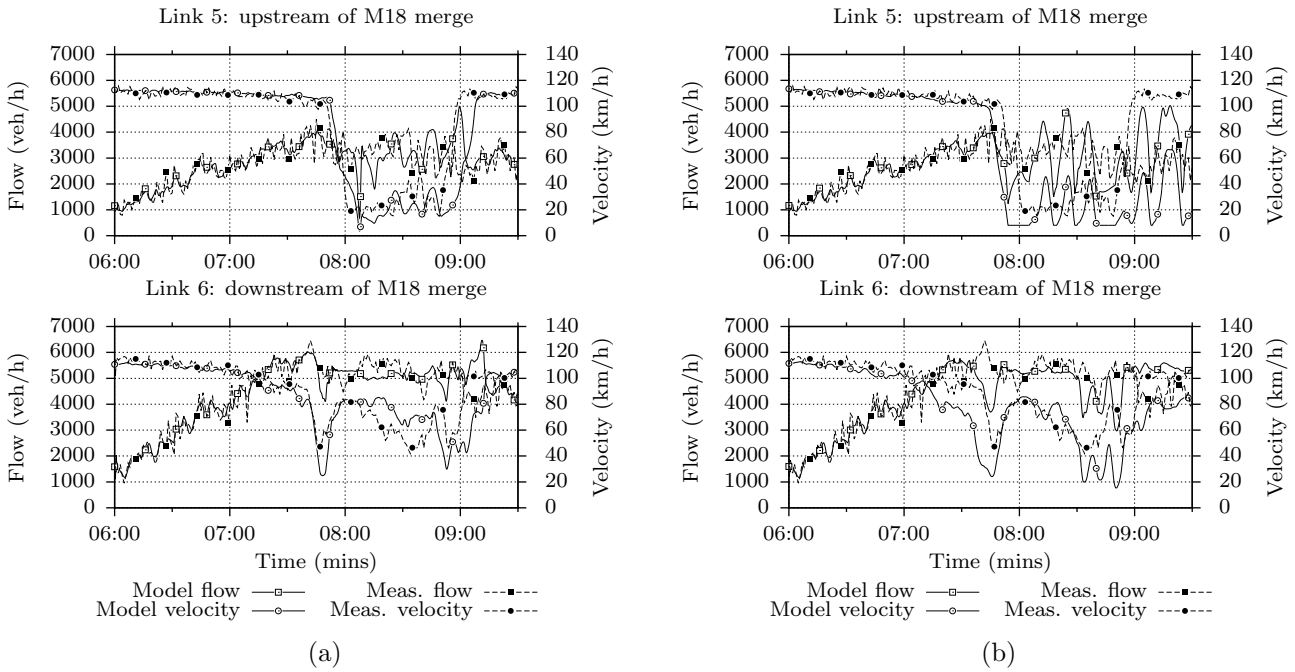


Figure 17: Verification of the Sheffield model using input data from set S15$^{\text{th}}$ and using the optimal parameter set based on (a) the S1$^{\text{st}}$ data set and (b) the S8$^{\text{th}}$ data set.

verification problem. The reasons for this result are unclear and are subject to further investigation.

Future research will also focus on investigating different ways of incorporating the generalisation property of a solution directly into the problem formulation. An automatic way of identifying appropriate values for the weighting parameters in the objective function is also something that needs further development. In fact, the suggested problem formulation implies a multi-objective problem, aiming concurrently at error minimisation, FD number and parameter variance minimisation, and maximum optimal solution generalisation. Explicit multiobjective optimisation methods can provide an alternative approach to the one adopted here. A learning module able to reason about the quality of data used for calibration and the solution obtained is another viable research direction.

The overall objective is a system that combines modelling, optimisation and learning capabilities to automate the process of delivering relevant and up to date model parameter sets with minimum interference from the operator.

Acknowledgement

The authors would like to thank the Highways Agency for the use of MIDAS data, EPSRC for providing funding for this work and Ms. Katherine Halfpenny for her initial work on the CS algorithm. Furthermore we wish to thank the reviewers for their detailed and very constructive comments and suggestions.

References

[1] S. P. Hoogendoorn, P. H. 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) (2001) 283–303.

[2] M. Lighthill, G. Whitham, On kinematic waves II: a traffic flow theory on long crowded roads, Proc. Roy. Soc. London Series A 229 (1955) 317–345.

[3] P. I. Richards, Shock waves on the highway, Operations Research 4 (1956) 42–51.

[4] A. May, Traffic Flow Fundamentals, Prentice Hall, 1990.

[5] J. Del Castillo, F. Benitez, On the functional form of the speed-density relationship–I: general theory, Transportation Research Part B 29 (5) (1995) 373–389.

[6] J. D. Castillo, F. Benitez, On the functional form of the speed-density relationship–II: Empirical investigation, Transportation Research Part B 29 (5) (1995) 391–406.

[7] C. Daganzo, The Cell Transmission Model: A dynamic representation of highway traffic consistent with the hydrodynamic theory, Transportation Research Part B 28 (1994) 269–287.

[8] C. F. Daganzo, The Cell Transmission Model, Part II: Network Traffic, Transportation Research Part B 29 (1995) 79–93.

[9] G. Wong, S. Wong, A multi-class traffic flow model–an extension of LWR model with heterogeneous drivers, Transportation Research Part A 36 (9) (2002) 827–841.

[10] J. Lebacque, J. Lesort, F. Giorgi, Introducing buses into first-order macroscopic traffic flow models, Transportation Research Record 1644 (1998) 70–79.

[11] J.-P. Lebacque, First-order macroscopic traffic flow models: Intersection modeling, network modeling, in: Proc. 16th Int. Symp. on Transportation and Traffic Theory, 2005, pp. 365–386.

[12] H. J. Payne, Models of freeway traffic and control, Proc. Simulation Council 28 (1971) 51–61.

[13] D. Helbing, A. Hennecke, V. Shvetsov, M. Treiber, Micro- and macrosimulation of freeway traffic, Mathematical and Computer Modelling 35 (5–6) (2002) 517–547.

[14] M. Cremer, M. Papageorgiou, Parameter identification for a traffic flow model, Automatica 17 (6) (1981) 837–843.

[15] D. Ngoduy, M. Maher, Calibration of second order traffic models using continuous cross entropy method, Transportation Research Part C 24 (2012) 102–121.

[16] A. Kotsialos, M. Papageorgiou, C. Diakaki, Y. Pavlis, F. Middelham, Traffic flow modeling of large-scale motorway networks using the macroscopic modeling tool METANET, IEEE Trans. Intell. Transp. Syst. 3 (4) (2002) 282–292.

[17] A. Messmer, M. Papageorgiou, METANET: A macroscopic simulation program for motorway networks, Traffic Engineering and Control 31 (1990) 466–470; 549.

[18] M. Papageorgiou, J.-M. Blosseville, H. Hadj-Salem, Modelling and real-time control of traffic flow on the southern part of Boulevard Peripherique in Paris: Part I: Modelling, Transportation Research Part A 24 (5) (1990) 345–359.

[19] M. J. Box, A new method of constrained optimization and a comparison with other method, The Computer Journal 8 (1) (1965) 42–52.

[20] D. Ngoduy, S. Hoogendoorn, H. Van Zuylen, Comparison of numerical schemes for macroscopic traffic flow models, Transportation Research Record 1876 (1) (2004) 52–61.

[21] M. Treiber, A. Kesting, Validation of traffic flow models with respect to the spatiotemporal evolution of congested traffic patterns, Transportation Research Part C 21 (1) (2012) 31–41.

[22] J. Nelder, R. Mead, A simplex method for function minimization, The Computer Journal 7 (4) (1965) 308–313.

[23] A. Spiliopoulou, M. Kontorinaki, M. Papageorgiou, P. Kopelias, Macroscopic traffic flow model validation at congested freeway off-ramp areas, Transportation Research Part C 41 (2014) 18–29.

[24] A. Poole, A. Kotsialos, METANET model validation using a genetic algorithm, in: Proc. of the 13th IFAC Symp. on Control in Transportation Systems, 2012, pp. 7–12.

[25] J. R. D. Frejo, E. F. Camacho, R. Horowitz, A parameter identification algorithm for the metanet model with a limited number of loop detectors, in: Proc. 51st IEEE Conference on Decision and Control, 2012, pp. 6983–6988.

[26] A. Kotsialos, A. Poole, Autonomic systems design for ITS applications, in: Intelligent Transportation Systems (ITSC), 2013 16th International IEEE Conference on, 2013, pp. 178–183.

[27] L. Munoz, X. Sun, D. Sun, G. Gomes, R. Horowitz, Methodological calibration of the cell transmission model, in: Proc. of the 2004 American Control Conf., Boston, MA, USA, 2004, pp. 798–803.

[28] L. Munoz, X. Sun, R. Horowitz, L. Alvarez, A piecewise-linearized cell transmission model and parameter calibration methodology, in: Proc. of the 85th Transportation Research Board Annual Meeting, Washington D.C., USA, 2006, pp. 183–191.

[29] M. Randall, Differential evolution for a constrained combinatorial optimisation problem, International Journal of Metaheuristics 1 (4) (2011) 279–297.

[30] A.-C. Zăvoianu, G. Bramerdorfer, E. Lughofer, S. Silber, W. Amrhein, E. P. Klement, Hybridization of multi-objective evolutionary algorithms and artificial neural networks for optimizing the performance of electrical

drives, Engineering Applications of Artificial Intelligence 26 (8) (2013) 1781–1794.

[31] R.-E. Precup, R.-C. David, E. M. Petriu, S. Preitl, M.-B. Radac, Novel adaptive gravitational search algorithm for fuzzy controlled servo systems, Industrial Informatics, IEEE Transactions on 8 (4) (2012) 791–800.

[32] Z. Zhang, Y. Jiang, S. Zhang, S. Geng, H. Wang, G. Sang, An adaptive particle swarm optimization algorithm for reservoir operation optimization, Applied Soft Computing 18 (2014) 167–177.

[33] Y. Wang, J. Zhou, C. Zhou, Y. Wang, H. Qin, Y. Lu, An improved self-adaptive PSO technique for short-term hydrothermal scheduling, Expert Systems with Applications 39 (3) (2012) 2288–2295.

[34] X.-S. Yang, S. Deb, Engineering optimisation by cuckoo search, Int. J. of Mathematical Modelling and Numerical Optimisation 1 (4) (2010) 330–343.

[35] S. Walton, O. Hassan, K. Morgan, M. Brown, Modified cuckoo search: A new gradient free optimisation algorithm, Chaos, Solitons & Fractals 44 (9) (2011) 710–718.

[36] Dynamic Systems and Simulation Laboratory, Technical University of Crete, Chania, Crete, Greece, "METANET Documentation".

[37] D. Goldberg, Real-coded genetic algorithms, virtual alphabets, and blocking, Complex Syst. 5 (1991) 129–167.

[38] Z. Michalewicz, Genetic algorithms + data structures = evolution programs, Springer, 1998.

[39] D. Goldberg, Genetic algorithms in search, optimization, and machine learning, Addison-Wesley, 1989.

[40] C.-T. Su, C.-L. Chiang, An incorporated algorithm for combined heat and power economic dispatch, Electric Power Systems Research 69 (2) (2004) 187–195.

[41] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Proc. 6th Inter. Symp. on Micro-machine and Human Science, IEEE, 1995, pp. 39–43.

[42] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proc. IEEE Int. Conf. on Neural Networks, Vol. 4, IEEE, 1995, pp. 1942–1948.

[43] Y. Shi, R. Eberhart, A modified particle swarm optimizer, in: Proc. IEEE World Congress on Comput. Intell., IEEE, 1998, pp. 69–73.

[44] Y. Shi, R. Eberhart, A modified particle swarm optimizer, in: Proc. IEEE World Congress on Computational Intelligence, IEEE, 1998, pp. 69–73.

[45] J. Kennedy, R. Mendes, Population structure and particle swarm performance, in: Proc. of the World Congress on Computational Intelligence, Vol. 2, IEEE, 2002, pp. 1671–1676.

[46] Z. Zhan, J. Zhang, Y. Li, H. Chung, Adaptive particle swarm optimization, IEEE Trans. Syst. Man Cybern. B 39 (6) (2009) 1362–1381.

[47] A. H. Gandomi, G. J. Yun, X.-S. Yang, S. Talatahari, Chaos-enhanced accelerated particle swarm optimization, Communications in Nonlinear Science and Numerical Simulation 18 (2) (2013) 327–340.

[48] A. H. Gandomi, A. H. Alavi, Krill herd: a new bio-inspired optimization algorithm, Communications in Nonlinear Science and Numerical Simulation 17 (12) (2012) 4831–4845.

[49] M. J. Mahmoodabadi, Z. Salahshoor Mottaghi, A. Bagheri, HEPSO: High exploration particle swarm optimization, Information Sciences 273 (2014) 101–111.

[50] X.-S. Yang, S. Deb, Cuckoo search via Lévy flights, in: Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on, IEEE, 2009, pp. 210–214.

[51] M. Zambrano-Bigiarini, M. Clerc, R. Rojas, Standard particle swarm optimisation 2011 at cec-2013: A baseline for future PSO improvements, in: Evolutionary Computation (CEC), 2013 IEEE Congress on, IEEE, 2013, pp. 2337–2344.

[52] M. Mernik, S.-H. Liu, D. Karaboga, M. Črepinšek, On clarifying misconceptions when comparing variants of the artificial bee colony algorithm by offering a new implementation, Information Sciences 291 (2015) 115–127.

[53] A. P. Piotrowski, Regarding the rankings of optimization heuristics based on artificially-constructed benchmark functions, Information Sciences 297 (2015) 191–201.

[54] A. V. Kononova, D. W. Corne, P. De Wilde, V. Shneer, F. Caraffini, Structural bias in population-based algorithms, Information Sciences 298 (2015) 468–490.