Contents lists available at SciVerse ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Coupling traffic models on networks and urban dispersion models for simulating sustainable mobility strategies[☆]

Stefano Berrone[a,∗], Francesca De Santi[b], Sandra Pieraccini[a], Massimo Marro[c]

[a] *Dipartimento di Scienze Matematiche, Politecnico di Torino, corso Duca degli Abruzzi 24 - 10129 Torino, Italy*

[b] *Dipartimento di Ingegneria Meccanica e Aerospaziale, Politecnico di Torino, corso Duca degli Abruzzi 24 - 10129 Torino, Italy*

[c] *Laboratoire de Mécanique des Fluides et d'Acoustique, Université de Lyon, CNRS, Ecole Centrale de Lyon, INSA Lyon, Université Claude Bernard, 36, avenue Guy de Collongue, 69134 Ecully, France*

ARTICLE INFO

Article history:
Received 7 September 2011
Received in revised form 23 December 2011
Accepted 15 March 2012

Keywords:
Urban air quality
Macroscopic traffic models
Road networks
Pollutant dispersion models
Traffic emissions control

ABSTRACT

The aim of the present paper is to investigate the viability of macroscopic traffic models for modeling and testing different traffic scenarios, in order to define the impact on air quality of different strategies for the reduction of traffic emissions. To this aim, we complement a well assessed traffic model on networks (Garavello and Piccoli (2006) [1]) with a strategy for estimating data needed from the model and we couple it with the urban dispersion model Sirane (Soulhac (2000) [2]).

1. Introduction

The aim of sustainable mobility is to reduce negative impacts caused by transport on everyday life of citizens: air pollution, acoustic pollution, traffic jams and car accidents. These impacts can vary in intensity in different urban areas, depending on geographical and meteorological conditions, urban planning, social, economics and cultural factors.

Reduction of these impacts can be achieved by local authorities in several ways by promoting public transport and by restricting the use of private cars (limited traffic areas, alternate circulation, car sharing, road pricing, traffic blocks).

However, it is well known that the planning and the realization of such measures require a considerable amount of organizing, administrative and economic resources. Furthermore, such actions can have a noticeable social and economical impact on the population. Hence, it is worthwhile to predict the effectiveness of the solutions which will be applied, in order to define, case by case, which are the optimal ones.

Focusing on air quality, the aim of the present paper is to propose an approach for evaluating the effect of different (modeled) traffic scenarios on the distribution of traffic air pollutants in urban areas, in order to study the impact of different strategies on the reduction of the traffic emissions and, consequently, on the air quality. To achieve this aim, we couple a well assessed traffic model on networks [1] with a pollutant dispersion model [2].

Concerning traffic simulations, several approaches can be used in order to evaluate traffic behavior. Traffic models proposed in literature include microscopic models, in which the behavior of each traveling vehicle is described through its position and velocity as a function of time, mesoscopic (or kinetic) models, in which traffic is analyzed through a distribution function $g(x, v, t)$ giving the number of vehicles that at time t are located in the position x and travel with velocity v, and macroscopic models, which deal only with averaged quantities such as density and mean velocity. We refer the reader to

[☆] The paper has been evaluated according to old Aims and Scope of the journal.

[∗] Corresponding author. Tel.: +39 0110907503; fax: +39 0110907599.
E-mail addresses: stefano.berrone@polito.it (S. Berrone), francesca.desanti@polito.it (F. De Santi), sandra.pieraccini@polito.it (S. Pieraccini), marro.massimo@ec-lyon.fr (M. Marro).

0898-1221/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2012.03.054

Ref. [3] for a recent review on traffic models. We will focus here on macroscopic traffic models, specifically on the traffic models on networks described in [1].

The pollutant dispersion model is implemented in the Sirane code [2] and is used to estimate and predict the downwind concentration of air pollutants produced by emissions of vehicles traveling in urban networks or other factors. Since the model requires the pollutant emissions as input data, the used traffic model aims at modeling and providing these emissions by a direct network traffic simulation without a detailed knowledge of origin–destination data of vehicles. In particular, we propose here some rules that allow us to estimate some parameters needed for computations which should depend on driver behavior, namely the traffic distribution arrays, see Section 2.3. In fact, these arrays have to be defined at each junction of the road network, thus requiring experimental data on all roads involved. When large scale computations (large districts or even whole cities) are performed, this would require a huge amount of data. Lack of data on some edges or nodes, will prevent from the use of the traffic models here used, unless some rules are defined for creating the traffic distribution arrays. We remark that we use here a model city, but after a suitable validation the suggested procedure may be used to assign such parameters whenever/wherever real data are not available, so that they can be used to complement experimental measurements.

The paper is organized as follows. In Section 2 we recall some basic definitions and properties about macroscopic traffic models and road networks. The dispersion model in urban areas implemented in the Sirane code is briefly described in Section 3. Finally in Section 4 we discuss our numerical experiments.

2. The traffic model

We will consider here macroscopic traffic models for the simulation of vehicular traffic. Following [4,1], we will consider a fluid-dynamic model for traffic flow on a road network by means of the conservation law formulation proposed by Lighthill, Whitham and Richards [5,6].

2.1. Generalities on macroscopic traffic models

Macroscopic models for traffic flow were first introduced by Lighthill and Whitham in 1955 and, independently, by Richards in 1956 in the pioneer works [5,6], by comparing vehicle traffic flow on “long crowded roads” and “highways” to fluid flows.

The Lighthill–Whitham–Richards (LWR) models are built prescribing conservation of the number of vehicles. Conservation of cars in each road is described by the nonlinear partial differential equation

$$\partial_t \rho + \partial_x f(\rho) = 0 \tag{1}$$

where $\rho = \rho(x, t) \in [0, \rho^{\max}]$ is the density of cars (number of vehicles for length unit), $(x, t) \in \mathbf{R}^2$, x is the space coordinate along the road, t is time and $\rho^{\max}$ is the maximum density of cars on the road; $f(\rho)$ is the flux (or traffic flow), given by $f(\rho) = \rho v(\rho)$ being v the velocity. Indeed, in LWR models v is assumed to depend only on ρ, typically being a smooth decreasing function of ρ, as it is clear that the velocity of cars diminishes as density increases.

Several models are proposed in literature, corresponding to different expressions of the flux f as a function of ρ. Furthermore, second order models have also been proposed, i.e. models in which the average velocity v of vehicles is no more assumed to depend only on ρ, and a second equation is added for the evolution of v, see for example the Aw–Rascle model [7].

We do not make any attempt here to compare different models. Furthermore, we remark that we aim at designing a general tool matching a rather general macroscopic traffic model with urban dispersion models. To attain this target, the choice of the specific model is not crucial, as the same settlement can be made with any model satisfying some rather general assumptions (see later in Section 2.3). For the sake of simplicity we will focus here on first order models: we will not consider here viscosity terms or second (or higher) order models. Indeed, since the matching between the macroscopic traffic model and Sirane does not depend in a crucial way on the specific traffic model used, the choice of a simple model allows us to easily describe this coupling.

Furthermore, we will not consider multilane models, but circulation on parallel lanes in some larger streets is in fact taken into account by considering different values of $\rho^{\max}$ in such roads.

Finally, we will not consider multipopulation models. Although these models would fit very well our context, as vehicles with different characteristics (e.g. cars, buses, trucks) may travel with different behaviors and pollute in different ways, they would introduce *systems* of conservation laws, increasing computational cost. Nevertheless, we want to stress that possible improvements in the traffic model can be taken into account, as the adopted approach can be extended to richer macroscopic models, as for example the recent phase transition model on networks [8]. We leave to future work the implementation of more complex traffic models.

2.2. Road networks

In order to model traffic in a urban framework, networks of roads must be taken into account. In [9] and the recent works [10,11,4,1] traffic on road networks is considered. We follow here the approach of the latter works, in which a network is described as a directed graph, with a collection of directed arcs (edges) meeting at some vertices.

In this subsection and in the following one we recall several results from [1] and we address the interested reader to this reference for a deeper comprehension. We start recalling the following definitions.

Definition 1 (*Network*)**.** A *network* is a couple $(\mathcal{I}, \mathcal{J})$ such that:

$\mathcal{I}$ is a finite collection of edges $I_k,\ k = 1, \dots, N_E$, described as intervals in $\bar{\mathbb{R}} = \mathbb{R} \cup \{\pm\infty\}$: $I_k = [a_k, b_k] \subseteq \bar{\mathbb{R}}$;
$\mathcal{J}$ is a finite collection of vertices $J_k,\ k = 1, \dots, N_V$. Each vertex J_k is given by the union of two non empty subsets of $\{1, \dots, N_E\}$ denoted by $\mathrm{Inc}(J_k)$ and $\mathrm{Out}(J_k)$.

The following is assumed:

1. For every $k' \neq k''$ we have $\mathrm{Inc}(J_{k'}) \cap \mathrm{Inc}(J_{k''}) = \emptyset$ and $\mathrm{Out}(J_{k'}) \cap \mathrm{Out}(J_{k''}) = \emptyset$.
2. If $k \notin \cup_{J \in \mathcal{J}} \mathrm{Inc}(J)$ then $b_k = +\infty$; if $k \notin \cup_{J \in \mathcal{J}} \mathrm{Out}(J)$ then $a_k = -\infty$. The two cases are mutually exclusive.

According to the previous definition, each vertex J is characterized by the indices corresponding to its incoming and outgoing edges (i.e. indices of the sets $\mathrm{Inc}(J)$ and $\mathrm{Out}(J)$, respectively).

Condition 1 clearly requires that each edge incomes in at most one vertex and outgoes from at most one vertex. Furthermore, according to condition 2 some edges may be infinite on one side if they are not incoming or are not outgoing for any vertex. Obviously, no edge extends to infinity on both sides because in this case it would not be connected with the remaining network.

An edge will be called related to a vertex J if it is either incoming in J or outgoing from J.

Definition 2 (*Road Network*)**.** A *road network* is a network in which edges represent unidirectional roads, with traffic flowing from a_k to b_k, and vertices represent junctions.

2.3. *A fluid-dynamic model on road networks*

The definition of the fluid-dynamic model on a road network is made by prescribing traffic behavior both on edges and at junctions.

In this subsection, the following assumptions are made:

(A1) $\rho^{\max} = 1$,
(A2) v depends only on the density ρ,
(A3) f is a strictly concave C^2 function,
(A4) $f(0) = f(1) = 0$.

Assumptions A1–A4 are mild hypotheses which can be partly relaxed (see [1]). In particular, A1 is assumed for the ease of description, but any value of $\rho^{\max}$ can actually be considered and on each road e, for $e = 1, \dots, N_E$, different values of $\rho_e^{\max}$ will be considered in Section 4.

On each road, the traffic is assumed to be modeled by a hyperbolic system of conservation laws:

$$\partial_t \rho_e + \partial_x f_e(\rho_e), \quad \rho_e \in \mathbb{R}^p,\ e = 1, \dots, N_E. \tag{2}$$

In (2) we can have $p > 1$ for example for multipopulation models. From now on, we will mostly assume $p = 1$.

The crucial point of modeling traffic behavior at junctions is accomplished by introducing Riemann solvers at vertices. We sketch here the main features of the model, referring the reader to [1] for all the details.

We briefly recall that, in the context of conservation laws, a Riemann problem is a Cauchy problem equipped with a piecewise constant initial datum having a single discontinuity. Letting ρ_L and ρ_R be the constant values on the left and on the right of the discontinuity, respectively, the Riemann problem will be denoted by the couple (ρ_L, ρ_R).

Let J be a vertex of the network and assume $\mathrm{Inc}(J) = \{1, \dots, n\}$ and $\mathrm{Out}(J) = \{n+1, \dots, n+m\}$. Let $\rho_0 = (\rho_{0,1}, \dots, \rho_{0,n+m})$ be a $n+m$-uple of constant initial data given on edges related to J.

Definition 3 (*Riemann Solver at Vertices*)**.** A Riemann solver for the vertex J is a function $RS : \mathbb{R}^{n+m} \to \mathbb{R}^{n+m}$ that associates to every initial datum $\rho_0 \in \mathbb{R}^{n+m}$ a vector $\hat{\rho} \in \mathbb{R}^{n+m}$ such that on each edge $I_k,\ k = 1, \dots, n+m$, related to J, the solution to (2) is given by the waves produced by the Riemann problem $(\rho_{0,i}, \hat{\rho}_i),\ i = 1, \dots, n$, for incoming edges and $(\hat{\rho}_j, \rho_{0,j}),\ j = n+1, \dots, n+m$, for outgoing edges. The Riemann solver is also asked to satisfy the consistency condition $RS(RS(\rho_0)) = RS(\rho_0)$.

To determine a Riemann solver at vertices, the following is assumed:

(A) at each junction some fixed coefficients express preferences of drivers, prescribing how incoming traffic flows into the outgoing roads;
(B) respecting (A), drivers choices are made in order to maximize the flux.

Rule (A) is imposed introducing for each junction a matrix

$$A = \{a_{ji}\} \in \mathbf{R}^{m \times n}, \quad j = n+1, \dots, n+m,\ i = 1, \dots, n$$

called *traffic distribution matrix*, whose element a_{ji} gives the percentage of drivers that, arriving from the i-th incoming road, take the j-th outgoing road. As a consequence we have for $i = 1, \dots, n$ and $j = n+1, \dots, n+m$

$$0 \le a_{ji} \le 1, \quad \sum_{j=n+1}^{n+m} a_{ji} = 1.$$

In order to ensure uniqueness of the Riemann solver at vertices, the matrix A has to satisfy the following technical condition.

(C) Let $\{e_1, \dots, e_n\}$ be the canonical basis of $\mathbb{R}^n$ and let $e = (1, \dots, 1) \in \mathbb{R}^n$. For every subset $V \subset \mathbb{R}^n$ let $V^\perp$ be its orthogonal. For every $i = 1, \dots, n$ let $H_i = \{e_i\}^\perp$, i.e. the coordinate hyperplane orthogonal to e_i and, for every $j = n+1, \dots, n+m$ let $\alpha_j = (a_{j1}, \dots, a_{jn}) \in \mathbb{R}^n$ and $H_j = \{\alpha_j\}^\perp$. Let $\mathcal{K}$ be the set of indices $k = (k_1, \dots, k_l)$, $1 \le l \le n-1$ such that $0 \le k_1 < k_2 < \dots < k_l \le n+m$ and for every $k \in \mathcal{K}$ set $H_k = \cap_{h=1}^{l} H_{k_l}$. Then, for every $k \in \mathcal{K}$ we ask

$$e \notin H_k^\perp.$$

As remarked in [1], condition (C) is needed to isolate a unique Riemann solver at junctions. A crucial point is that condition (C) implies $m \ge n$, so that the case $m < n$ will be treated in a different way.

Existence and uniqueness of a Riemann solver, producing admissible weak solutions to Riemann problems satisfying rules (A) and (B), as well as conservation of ρ at junctions, is stated for the case $m \ge n$ in [1, Theorem 5.2.1], whose proof gives all the details for building $\hat{\rho}$. Firstly, the maximum fluxes attainable by a single wave on each road, corresponding to a given initial datum ρ_0, are computed. Let us denote by $\gamma_i^{\max}(\rho_{0,i})$, for $i = 1, \dots, n$, such maximum fluxes for incoming roads, and by $\gamma_j^{\max}(\rho_{0,j})$ those for outgoing roads, for $j = n+1, \dots, n+m$. Defining

$$\Omega_i = [0, \gamma_i^{\max}(\rho_{0,i})], \quad i = 1, \dots, n \qquad \Omega_j = [0, \gamma_j^{\max}(\rho_{0,j})], \quad j = n+1, \dots, n+m$$

and

$$\Omega = \{\gamma \in \mathbb{R}^n : \gamma \in \Omega_1 \times \dots \times \Omega_n, A\gamma \in \Omega_{n+1} \times \dots \times \Omega_{n+m}\},$$

all feasible fluxes clearly belong to the closed, convex and not empty set Ω.

It turns out that the incoming fluxes $\hat{\gamma} = (\hat{\gamma}_1, \dots, \hat{\gamma}_n)$ which maximize the total flux, subject to rule (A), are given by the solution of the Linear Programming (LP) problem

$$\max_{\gamma \in \Omega} e^T \gamma. \tag{3}$$

Thanks to assumption (C), the solution to problem (3) is unique.

Once $\hat{\gamma}$ is obtained, for every $i = 1, \dots, n$, we choose $\hat{\rho}_i \in [0, 1]$ such that $f(\hat{\rho}_i) = \hat{\gamma}_i$ as in [1, Section 5.2.1]. Afterward we set

$$\hat{\gamma}_j = \sum_{i=1}^{n} a_{ji} \hat{\gamma}_i, \quad j = n+1, \dots, n+m$$

and we choose $\hat{\rho}_j \in [0, 1]$ such that $f(\hat{\rho}_j) = \hat{\gamma}_j$. This way, the whole vector $\hat{\rho}$ is computed and all the needed data for solving the conservation law on the edges related to vertex J are available.

In the case $m < n$, as condition (C) is not satisfied, a different approach has to be used. The approach followed here is sketched in [1] for the special case $m = 1$, and the more general case of arbitrary m is described in [12], in the context of packet flows on telecommunication networks. We introduce the vector $q \in \mathbb{R}^n$, with $\sum q_i = 1$, which is a vector of *right of way* parameters [1], and the traffic distribution vector $a \in \mathbb{R}^m$, with $\sum_{j=n+1}^{n+m} a_j = 1$. The meaning of the vectors is explained in the following further rules.

(D1) Let F be the amount of cars arriving at the given junction that actually enter the outgoing roads: then, $q_i F$ is the amount of cars arriving from the ith incoming road.

(D2) Let F be as in (D1): then, $a_j F$ is the amount of cars entering outgoing road j.

Rules (D1) and (D2) allow the computation of the states $\hat{\rho}$ as follows. Firstly, compute the maximal flux for the junction as

$$\gamma^* = \min \left\{ \sum_{i=1}^{n} \gamma_i^{\max}(\rho_{0,i}), \sum_{j=n+1}^{n+m} \gamma_j^{\max}(\rho_{0,j}) \right\}.$$

Next, let us consider the line r in $\mathbb{R}^n$ given by

$$\begin{cases} \gamma_n = \dfrac{q_n}{q_1} \gamma_1 \\ \vdots \\ \gamma_n = \dfrac{q_n}{q_{n-1}} \gamma_{n-1} \end{cases}$$

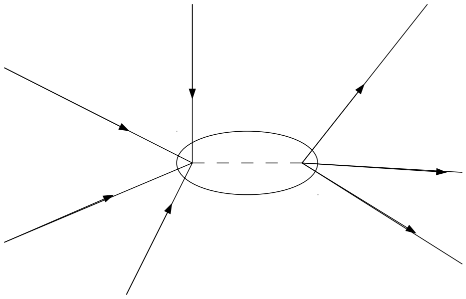

Fig. 1. Case $n > m$: fictitious road.

representing the set of points corresponding to fluxes with ratios respecting rule (D1). Define the closed and convex set $\mathcal{K}$ in $\mathbb{R}^n$

$$\mathcal{K} = \{\gamma \in \mathbb{R}^n : \gamma \in \Omega_1 \times \cdots \times \Omega_n,\ \gamma_1 + \cdots + \gamma_n = \gamma^*\}$$

giving the set of feasible fluxes on incoming edges, and let P be the intersection between the line r and the hyperplane

$$\gamma_1 + \cdots + \gamma_n = \gamma^*.$$

If $P \in \mathcal{K}$, the incoming fluxes $\hat{\gamma} = (\hat{\gamma}_1, \dots, \hat{\gamma}_n) \in K$ are then given by P; otherwise, $\hat{\gamma}$ is given by the unique point in $\mathcal{K}$ which minimizes the distance from P. This point is easily found solving a convex Quadratic Programming (QP) problem.

Next, traffic is distributed among the m outgoing roads in a similar manner: recalling rule (D2), we compute the point $Q \in \mathbb{R}^m$ whose coordinates are given by $\gamma_j = a_j\gamma^*,\ j = n+1, \dots, n+m$. Then, we set

$$\mathcal{H} = \left\{(\gamma_{n+1}, \dots, \gamma_{n+m}) \in \mathbb{R}^m : \sum_{j=n+1}^{n+m} \gamma_j = \gamma^*,\ \gamma_j \in \Omega_j,\ j = n+1, \dots, n+m\right\}.$$

If $Q \in \mathcal{H}$, then the outgoing fluxes are given by Q; otherwise $\hat{\gamma}$ is given by the unique point in $\mathcal{H}$ which minimizes the distance from Q.

Similarly to the case $n \leq m$, we will sometimes refer to the two vectors q and a as to *traffic distribution vectors*. We will use the term *traffic distribution arrays* for referring both to matrices and to vectors without the need of distinguishing the two cases.

As in the case $n \leq m$, once the fluxes $\hat{\gamma}_i$ and $\hat{\gamma}_j$ are computed, inversion of f leads to the corresponding densities $\hat{\rho}_i$ and $\hat{\rho}_j$.

Remark 1. The approach followed in the case $m < n$ can be viewed as originated by the introduction of a fictitious road in the junction, see Fig. 1. In practice, one assumes that all the n incoming roads enter a single fictitious road, and this is accomplished, using rule (D1), as in the case $m = 1$ sketched in [1]; then, vehicles coming from the fictitious road enter the m outgoing roads following (D2).

2.4. Numerical approximation of conservation laws on edges

In this section we describe the numerical scheme used for solving the conservation law (1) on each edge. In the next subsection we will end summing up, and describing the whole process on all the network.

We choose a global space discretization size Δx and timestep Δt. For each edge $e = 1, \dots, N_E$ we set $n_e = \lfloor \frac{L_e}{\Delta x} \rfloor$ and $\Delta x_e = \frac{L_e}{n_e}$, being L_e the length of edge e. This way we have on each edge a uniform spacing Δx_e which is nearly the same for all edges. Then, we let $\lambda_e = \Delta x_e/\Delta t$. A space–time grid is hence formed on each edge by points $(x_j, t^n) = (j\Delta x_e, n\Delta t),\ j \in \mathbb{Z}$ and $n \in \mathbb{N}$. Let u_e be any function defined on the space–time grid on the edge e. For the sake of simplicity, we skip the edge index e in the function u; we set $u_j^n = u(x_j, t^n)$ and we let $\bar{u}_j^n$ denote the cell average over the cell $I_j = \{x : |x - x_j| \leq \frac{\Delta x_e}{2}\}$ at time t^n.

We assume that the initial datum is assigned on each road of the network, and boundary conditions are assigned on road extrema which do not enter in/go out of a junction.

We consider here the second order non oscillatory central scheme proposed in [13]. The scheme is based on the staggered form of Lax–Friederichs scheme. Central differencing provided by this scheme allows a great deal of simplicity in approximating the conservation law as, unlike upwind schemes, it does not require information about characteristic speeds. The excessive numerical viscosity introduced by the Lax–Friederichs scheme is compensated by high resolution interpolants. Accurate choices of numerical slopes in the reconstruction will provide us with second order accuracy, whereas slope limiting will prevent from spurious oscillations.

In details, let us consider the following piecewise linear approximation of the solution:

$$u(x,t^n)=\sum_j\left[\bar{u}_j^n+u_j'\left(x-x_j\right)\right]\chi_j(x), \tag{4}$$

where $\chi_j(x)$ is the characteristic function of the cell I_j and u_j' is a suitable numerical approximation of the space derivative at x_j (see later for details).

Evolving in time (4) and then projecting on the space of staggered cell averages, we obtain

$$\bar{u}_{j+\frac{1}{2}}^{n+1}=\frac{1}{2}(\bar{u}_j^n+\bar{u}_{j+1}^n)+\frac{1}{8}(u_j'-u_{j+1}')\Delta x_e-\frac{1}{\Delta x_e}\int_{t^n}^{t^{n+1}}\left(f(u(x_{j+1},t))-f(u(x_j,t))\right)\,\mathrm{d}t. \tag{5}$$

Under the CFL condition

$$\lambda_e\max|f'(u)|<\frac{1}{2}$$

waves emanating by discontinuities in $u(x,t^n)$ do not reach interfaces between the staggered control volumes; hence, the solution remains smooth at such interfaces and the integral in (5) can be approximated by midpoint rule. Thus the scheme reads

$$u_{j+\frac{1}{2}}^{n+1}=\frac{1}{2}(u_j^n+u_{j+1}^n)+\frac{1}{8}(u_j'-u_{j+1}')\Delta x_e+\lambda_e\left(f\left(u_j^{n+\frac{1}{2}}\right)-f\left(u_{j+1}^{n+\frac{1}{2}}\right)\right). \tag{6}$$

The intermediate value $u_j^{n+\frac{1}{2}}$ is computed by Taylor expansion:

$$\bar{u}_j^{n+\frac{1}{2}}=\bar{u}_j^n-\frac{\lambda_e}{2}f_j'\Delta x_e, \tag{7}$$

where again f_j' is a first order non-oscillatory approximation of the space derivative of f at x_j.

As a whole, the numerical method is a predictor–corrector scheme in which (7) is the predictor and (6) is the corrector.

Concerning slopes, we recall that u_j' in (4)–(6) and f_j' in (7) provide first order approximations of the space derivatives at x_j. Non oscillatory behavior is obtained with the use of slope limiting. In details, we set

$$u_j'=\frac{1}{\Delta x_e}\mathrm{MM}(\theta(u_{j+1}-u_j),u_{j+1}-u_{j-1},\theta(u_{j+1}-u_j))$$

where MM denotes the tree terms MinMod nonlinear limiter

$$\mathrm{MM}(x_1,x_2,x_3)=\begin{cases}\min|x_i| & \text{if } x_i>0\ \forall i \text{ or } x_i<0\ \forall i\\ 0 & \text{otherwise}\end{cases}$$

and $\theta\in[1,2]$. A similar definition is adopted for f_j'.

Remark 2. The scheme here described proposed in [13], can be slightly modified in such a way that staggered cells are not needed. This new variant of the scheme, described in [14], is actually implemented in our scheme. For the sake of simplicity, here we limit ourselves to the description of the original staggered scheme.

2.5. Numerical approximation on the network

Once Riemann solvers at junctions are available, a solution of the overall problem is computed as follows.

We start each time step with a cell averaged data on the network. Then, we build Riemann solvers at junctions thus obtaining boundary conditions for endpoints of edges related to the junctions. Boundary data will be assigned at endpoints that do not meet at any junction. We remark that the infinite edges described in Definition 1 will be obviously finite in actual computations, thus such edges will not end at junctions and will require boundary conditions.

Then, on each road, one step of the numerical method is performed on the conservation Eq. (2). After this step, new cell averages are obtained on the edges, including the first and the last cell of the edges. Thus, at junctions, new Riemann solvers have to be applied in order to obtain the new boundary conditions for the next time step. The whole procedure is then repeated until the final time is reached.

3. Atmospheric dispersion modeling

Air dispersion models are useful tools for important and social targets such as air quality and traffic management, urban planning, data monitoring, pollution forecast, risk analysis, public safety purposes (as for example emergency planning for accidental chemical releases). Several models and codes were developed for these targets; most of them are simplified models in order to require a moderate computational effort and to provide realtime solutions.

In this paper we are interested in the pollutant dispersion in an urban district. In the last twenty years few models have been developed for this target. SBLINE [15] assumes that the pollutant in each street is due to the contribution of sources located in the street itself and those located in the surrounding streets. This is taken into account by a Gaussian model, assuming that the plume of pollutants is transported as if there were no buildings. Similarly, ADMS-Urban [16] provides a module to compute mean concentration in the regions of the domain where the street canyon effect arises. For each street canyon, the concentration is computed as the sum of two components: the background concentration due to street canyon trapping effect and the concentration related to the direct contribution of vehicle emissions within the street. The dispersion of pollutant emitted within the street is modeled by a Gaussian plume.

In this paper we use the urban dispersion model Sirane, that adopts a different approach with respect to ADMS-Urban and SBLINE models. It is based on the street network concept and on a decomposition of the whole flow into external atmospheric and urban canopy sub-flows. The Sirane code has been used to compute street level concentration in several large European cities: Paris, Grenoble, Le Havre, Rouen, Chambery, Lyon [17], Turin, Milan, Florence [18–20]. In the following we briefly describe the parametrization adopted in Sirane. For a more detailed description of the model we refer the reader to [21].

Sirane simulates pollutant dispersion emitted from line sources (e.g. traffic emissions) and point sources (e.g. chimneys) at the district scale, i.e. for length scales ranging from few hundred meters to few kilometers. Therefore, Sirane neglects the influence of the topography (length scale >100 m) and it models the effects of the details of the building geometry (doors, chimneys, windows) as a uniformly distributed wall roughness. The building scale (length scale $\sim$10 m) is the only explicitly represented scale.

The code adopts a quasi-steady approximation. The time step length is assumed to be equal to one hour. This choice is motivated by the spectral gap between the time scale associated to the dynamic of atmospheric turbulence and the time scale related to the variation of the synoptic meteorological conditions. For each time step, pollutant dispersion is computed assuming steady conditions and concentration is estimated assuming no contribution of the pollutant emissions emitted previously. This approach reveals some limits in case of calm wind conditions persisting over several hours, that can induce a significant accumulation of pollutants over the urban area which is not taken into account by the model.

The model is made up of two independent modules to compute flow and dispersion within the urban canopy and in the overlying atmospheric boundary layer. In order to compute the mean concentration within each street, Sirane accounts for three transport mechanisms within the canopy:

- convective mass transfer along the street due to the mean wind along their axis [22];
- convective transport at street intersections [23];
- turbulent transfer across the interface between the street and the overlying atmospheric boundary layer [24].

We describe the first two mechanisms in the next subsection. The third one is described in the Section 3.2.

3.1. Flow and dispersion within the urban canopy

The urban canopy is modeled as a simplified network of connected street segments represented by boxes. The box dimensions are the street length L, the width W and the height H. The flow within each street is driven by the parallel component of the external wind and the pollutant is assumed to be uniformly mixed over the volume of the street. The spatially averaged velocity along the street axis, U_{street}, is evaluated by a balance between the turbulent entrainment at roof level and the drag at the building walls [22]. This velocity is used in order to estimate the mass balance within each street and it is computed as function of different parameters:

$$U_{\text{street}} = U_{\text{street}}(u_*, \varphi, z_{0,\text{build}}, H, W)$$

where u_* is the friction velocity of the overlying boundary layer flow, φ is the external wind direction with respect to the street axis, $z_{0,\text{build}}$ is the aerodynamic roughness of canyon walls.

The flow patterns within a street intersection are strongly related to the size, orientation and relative distance of the buildings bordering the streets that form the intersection. Two kinds of intersections are identified. When the street aspect ratios W/H are sufficiently high, the intersection is considered a 'simple' one and the flow within the intersection is almost decoupled from that in the external flow. Conversely, for low aspect ratios, in 'large squares', the external flow penetrates deeply into the urban canopy. In order to distinguish these two cases a threshold value for the street aspect ratio is needed: if $W \leq 3H$ the intersection is 'simple', otherwise it is considered as 'open terrain region'.

In an intersection the flux is modeled by taking into account the horizontal air flux from one street to another, and the vertical air flux between the urban canopy and the external atmosphere. The horizontal air flux is estimated assuming that the flow dynamics within the intersection is two-dimensional, i.e. that the topology of the streamlines does not depend on the vertical distance from the ground. The vertical air flux is modeled by a simple balance of the air volumes entering and leaving the intersection. In particular, vertical flux is nonzero when the air flow rate through the cross sections of the upwind and downwind streets are different.

The spatially averaged concentration in each street of the district is obtained by a box model computing a balance of the pollutant fluxes entering and leaving the street volume. Assuming steady state conditions, this mass balance takes into account several terms. The most important are the pollutant mass rate emitted within the street (line and point sources),

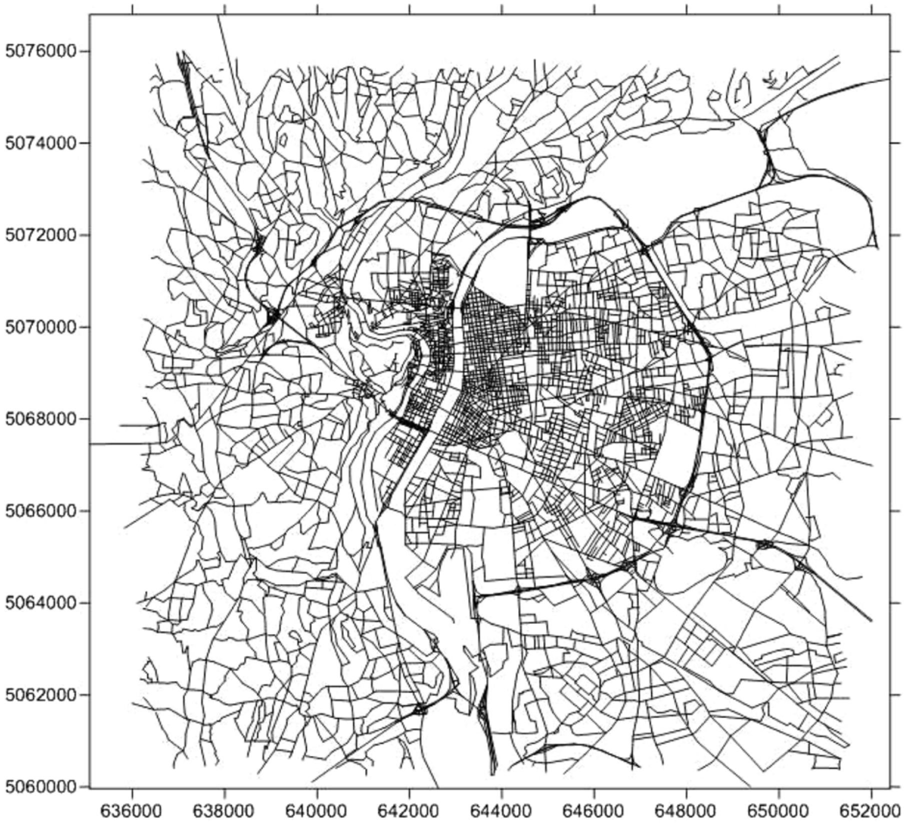

Fig. 2. City map.

the entering flux of pollutants advected by the mean flow along the street axis due to the transfer at the street intersections, the leaving flux of pollutants advected by the mean flow along the street axis, the turbulent transfer between the street and the atmosphere. This last flux is assumed to be proportional to the concentration difference

$$Q_{\text{turb,street}} = u_d WL(C_{\text{ext}} - C_{\text{street}}),$$

where C_{street} and C_{ext} are the mean concentrations within and above the street, respectively, and u_d is a transfer velocity based on the standard deviation of the turbulent vertical velocity following the Monin–Obukhov similarity theory.

3.2. Flow and dispersion above the roof level

A reliable description of the atmospheric boundary layer in Sirane is required to determine the characteristics of the atmospheric flow and turbulence, that are needed to estimate the pollutant dispersion in the external atmospheric flow. Moreover, they allow us to evaluate the turbulent pollutant fluxes between the urban canopy and overlying atmosphere and to compute the parameters describing the flow within the canopy. The external boundary layer is modeled using Monin–Obukhov similarity theory. The flow is assumed homogeneous in the horizontal plane and all the dynamical parameters are assumed to depend on the vertical coordinate only. The external boundary layer flow model is mainly inspired by some well assessed meteorological pre-processors [25,26].

The pollutant dispersion phenomena taking place above roof level are simulated by a Gaussian plume model. Plume reflections at the top of the boundary layer and at roof level are simulated by the introduction of image sources. This technique requires to locate fictitious sources at the mirror images of the real sources in order to impose no-flux conditions at the reflecting surfaces [27].

4. Numerical experiments

In this section we present our numerical experiments, obtained (on different traffic scenarios) by coupling network traffic simulations, performed implementing the traffic model described in Section 2, and the Sirane code.

We have made some experiments on a city model whose map is depicted in Fig. 2 and for which experimental data on traffic are not available to us. This topology corresponds to a large part of Lyon, France. It consists of 19 224 edges and 7587 junctions; length of edges ranges from few meters to 1960 m.

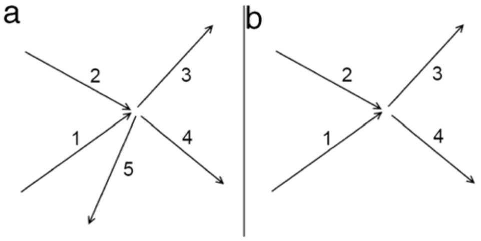

Fig. 3. Junction example.

Since real data are not available, some strategies were used in order to fix the traffic distribution arrays and some parameters needed from the models. For a practical and reliable application of the model, tuning of some input parameters will be needed.

4.1. Setting traffic distribution arrays

Among all possible LWR models, we focused here on the Greenshield model [1], in which one has

$$v(\rho) = v_{\max}\left(1 - \frac{\rho}{\rho^{\max}}\right). \tag{8}$$

Here $v(\rho)$ decreases linearly with ρ from its maximum value $v_{\max}$ (attained at $\rho = 0$) to its lowest value 0 attained when density assumes its larger value $\rho^{\max}$. The flux function trivially satisfies assumptions A2–A4 of Section 2.3.

The traffic model is linked to the code Sirane, which takes advantage from a Geographic Information System (GIS) to obtain coordinates, length and width of roads of real cities. Besides these parameters, other information about the traffic situation in cities under consideration should be prescribed, namely the traffic distribution arrays for all the junctions.

In the following we will describe some rules for assigning traffic distribution arrays. We want to point out that these rules appear as a suitable way for assigning traffic distribution arrays for tests where experimental data are not known. Few traffic data are presently available to us for the city of Torino. A preliminary comparison of the results obtained by our rules with these traffic data gives quite satisfactory results at least for fluxes corresponding to peak hours. A deeper validation of this model, which is beyond the scope of the present paper and is left to future work, will allow us to finely tune this model in order to use it for filling possible gaps in available real data, i.e. in real life computations the model will be possibly used to assign traffic distribution arrays for those junctions for which real data are not available.

At first, let us define the traffic distribution matrix for junctions with $n \leq m$. We recall that element a_{ji} represents the percentage of drivers coming from road i who enter road j after crossing the junction. We assume that these percentages are ruled by the width of outgoing roads, assuming that width gives, at least to drivers' perception, an indication of road capacity and, as a consequence, of drivers' interest in the road.

Furthermore, provided roads' direction, we want traffic distribution arrays to model a traffic behavior in which most drivers privilege their present direction of movement, i.e. after the junction they prefer to enter the road with minimal angle with respect to the incoming road. At the same time, U-turns (or almost-U-turns) are still allowed. Let us consider, for example, the junction shown in Fig. 3(a): roads 1 and 2 are incoming, whereas roads 3–5 are outgoing. Supposing that all the roads have the same width, we assume that a large part of drivers arriving from road 1 will proceed entering road 3, whereas just a few drivers will enter road 5.

Furthermore, in case (b), if all the roads have the same width and if we do not consider at all the angles between the streets, using only information from road width, we would obtain a matrix A having the form

$$\begin{bmatrix} \alpha & \alpha \\ 1-\alpha & 1-\alpha \end{bmatrix}, \qquad \alpha \in [0, 1]$$

which does not satisfy condition (C) of Section 2.3.

More in detail, we define A as follows: for $i = 1, \dots, n$ and $j = n + 1, \dots, n + m$ the elements a_{ji} are given by

$$a_{ji} = \frac{w_j p_{ij}}{\sum\limits_{k=n+1}^{n+m} w_k p_{ik}},$$

where w_j is the width of the outgoing road j and p_{ij} is defined as

$$p_{ij} = \left(\frac{1}{2} - \varepsilon\right)(\cos\beta_{ij} + 1) + \varepsilon$$

where β_{ij} is the angle underlying roads i and j and ε is a small nonzero parameter. The rationale behind this formula is the following: the more the outgoing road j has the same direction of incoming road i, the more road j is appealing for drivers

Table 1
Traffic composition (% of vehicles).

	Cars	Trucks	Motorbikes	Buses
Conventional/ECE 1504	10.22	1.79	4.49	0.10
Euro 1	5.44	0.96	2.07	0.04
Euro 2	21.18	2.04	1.82	0.10
Euro 3	19.74	2.77	0.89	0.05
Euro 4	25.40	0.88	–	0.003
Euro 5	–	–	–	0.02
Total	81.99	8.44	9.27	0.31

Table 2
Weighted pollutant emissions factors (g/km).

NOX	VOC	CO	PM
0.711	1.108	8.630	0.079

coming from road i, whereas if β_{ij} is almost π, i.e. essentially an U-turn is made when going from road i to road j, then p_{ij} is small. The nonzero parameter ε still allows some U-turns to be made.

In the case $n > m$, priorities among incoming roads are set as

$$q_i = \frac{w_i}{\sum\limits_{k=1}^{n} w_k} \quad i = 1, \dots, n$$

and, similarly, the traffic distribution vector for outgoing roads is defined by:

$$a_j = \frac{w_j}{\sum\limits_{k=n+1}^{n+m} w_k}, \quad j = n+1, \dots, n+m.$$

We remark that in this case information concerning origin of drivers is lost when crossing the junction. This is the reason why quantities relating incoming and outgoing roads (namely p_{ij}) are not used in the definition of q and a.

We also remark that time-dependent traffic distribution matrices can also be considered in order to model, for example, traffic lights and/or different driver preferences with daily hours.

4.2. Setting vehicle parameters

For the evaluation of pollutants, we used data provided by `Copert` [28,29], a software for computing air pollutant emissions from road transport. From this software it is possible to obtain, for each class of vehicles, pollutant emissions factors for some given pollutants, measured in grams per kilometer of distance covered, (g/km).

Concerning traffic compositions, in our computations we used the data reported in Table 1, corresponding to the situation in Torino in 2007. For each type of vehicle, the table report the percentage on the total traffic, distinguishing among different European emissions standards. The latest European standard emissions available are Euro 4 for cars and trucks, Euro 3 for motorbikes and Euro 5 for buses.

Since we do not use a multipopulation model, our computations actually do not distinguish among different vehicle types as far as density computations are concerned. Different emissions are accomplished by first computing the *weighted* emissions factors. Let C be the number of different vehicles classes ($C = 20$ in Table 1) and let p_i be the percentage of vehicles of class i, for $i = 1, \dots, C$; for a given pollutant P let ε_i^P be the pollutant emission factor of vehicles of class i: then, the weighted emission factor for P is

$$\varepsilon^P = \sum_{i=1}^{C} p_i \varepsilon_i^P .$$

In Table 2 we report the weighted emissions factors computed with our data. After our computations are performed on the whole network, on each edge we average density both in space and in time. Since the traffic model links density to flux velocity, we easily obtain the averaged number of kilometers covered by vehicles traveling on the given edge, and, as a consequence, total emissions for the edge which are then processed by Sirane.

Fixing a mean length for vehicles (we used here: cars 4.0 m, trucks 8.0 m, motorbikes 2.2 m, buses 8.5 m) and following traffic composition given in Table 1, we may compute the length of the *mean weighted vehicle* (here $L_v = 4.1821$ m) and $\rho^{\max}$ can then be computed as $\rho^{\max} = d/L_v$ vehicles/m, where d is a positive integer giving the number of lanes present in the road.

The maximum velocity $v_{\max}$ is set to $v_{\max} = 50$ km/h$\approx$14 m/s.

Please cite this article in press as: S. Berrone, et al., Coupling traffic models on networks and urban dispersion models for simulating sustainable mobility strategies, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.03.054

As initial condition, we set on each edge a constant density, whose value ρ_e^0 depends, on each edge $e = 1, \ldots, N_E$, on $\rho_e^{\max}$ (namely $\rho_e^0 = 0.15\rho_e^{\max}$).

We recall that boundary conditions are needed for edge endpoints which do not correspond to junctions. These may be left endpoints (i.e. the edge *enters* in the network) or right endpoints (i.e. the edge *exits* from the network). In the first case, we set as boundary condition Dirichlet conditions, with values constant in time still depending on $\rho_e^{\max}$, and also distinguishing the cases in which these edges correspond to main roads entering the town or to small blind alleys. In the second case (edge exiting from the network) we set free-flow boundary conditions.

4.3. Implementation details

The algorithm described is easily implementable in a parallel way, both on shared memory architectures (OpenMP) and on distributed memory architectures (MPI). Indeed, a large part of the computations can be easily identified in two distinct blocks: computations performed to build Riemann solvers at vertices (`NodeSolve` procedure) and solution of conservation law on edges (`EdgeSolve` procedure). The data exchange between processors/threads only concerns data corresponding to the first and the last cell on each edge, and we can easily assign some junctions to each processor in `NodeSolve`, and some edges to each processor in `EdgeSolve`. This is the choice we adopted in our implementation in a C code, in which we used a OpenMP parallelization, achieved by splitting the edges and the junctions among the threads used for computations.

In our implementation LP problems are solved by the `GLPK` (GNU Linear Programming Kit) package [30] (version 4.47), a library of ANSI C routines implementing a revised simplex method. QP problems have been solved by `QuadProg++` [31] (version 1.2), a C++ library for Quadratic Programming which implements the Goldfarb–Idnani active-set dual method [32].

4.4. Results

We implemented in our code the following traffic scenarios:

- normal traffic (NT): no restrictions are considered;
- local environmental traffic block (EB): a very large amount of cars is not allowed to enter in a prescribed zone (typically a central zone of the town);
- cars and trucks are all conforming to Euro 4 specifications (E4).

A few remarks on previous scenarios. For the sake of simplicity, the EB zone is here a ball with a prescribed center and radius; of course any kind of shape can be assumed. The EB scenario has been implemented by penalizing vehicles from entering the EB zone through a proper definition of traffic distribution arrays, whereas vehicles already in the zone may freely travel both inside and outside the zone.

In Euro 4 scenario we assume that all circulating vehicles are conforming to Euro 4 specifications, i.e. all the non conforming vehicles are replaced with conforming ones, so that the total number of vehicles circulating does not change with respect to the normal traffic.

For a fair comparison, we started our computations with the same initial and boundary conditions for vehicle densities in all the three scenarios, except for the EB scenario where, inside the EB zone, we obviously used smaller initial and Dirichlet boundary conditions. That is coherent with a small amount of cars traveling inside the block zone.

In our simulations we have run the traffic model for a fixed time and the dispersion model for a comparable time. We did not use background pollution distribution in the dispersion model, thus highlighting the pollutant production due to traffic.

In Figs. 4–9 we report the results obtained for the pollutant dispersion computed by Sirane on the basis of the performed traffic simulations in the three different scenarios, which have been obtained with a space meshsize $\Delta x = 0.5$. All figures refer to PM concentrations, and values in output are measured in $\mu g/m^3$. In particular, Figs. 4–6 show pollutant dispersion over almost the whole city. In such figures pollutant dispersion is superimposed on the city map. In Figs. 7–9 details are given on a part of the city located northeast of the city center. This zone includes both a portion of the city involved by the environmental traffic block and a portion outside this region.

Comparing Fig. 8 with Figs. 7 and 9 we can clearly see the effect of the Environmental Block measure from the low levels of PM concentration in the bottom-left corner. Also the maximum values of PM concentrations corresponding to EB and E4 measures are lower than the values reached considering normal traffic. We notice that even if EB clearly yields very small values of pollutants inside the block, as a whole the E4 measure seems to be more effective with respect to the reduction of maximum values. In Tables 3–5 we report, for the three traffic scenarios here considered, some punctual mean concentration values computed by the dispersion model at given points called *receptors*. These tables report, for nine fixed receptors, the mean values computed for all the pollutants considered in our simulations (NO, NO_2, O_3 and PM, all measured in $\mu g/m^3$). For each receptor we also report *x* and *y* UTM coordinates inside the city. The points marked with Id $= 0, \ldots, 5$ are located on the same horizontal axis, starting from the center of the city and moving toward East; the receptor with Id $= 0$ is close both to the center of the city and to the center of the EB region, receptor 1 is inside the EB but not far from its boundary, receptor 2 is very close to the boundary, receptor 3 is immediately outside the EB region, receptors 4 and 5 are far from the EB region. Receptors 6–8 are located northeast of the center of the EB region, receptor 6 being outside the region, receptor 7 on the boundary and receptor 8 inside the EB zone. Comparing the concentrations in Tables 3 and 4 we can clearly see the reduction due to the traffic block inside the EB region. This reduction seems to have no effect at all on zones outside

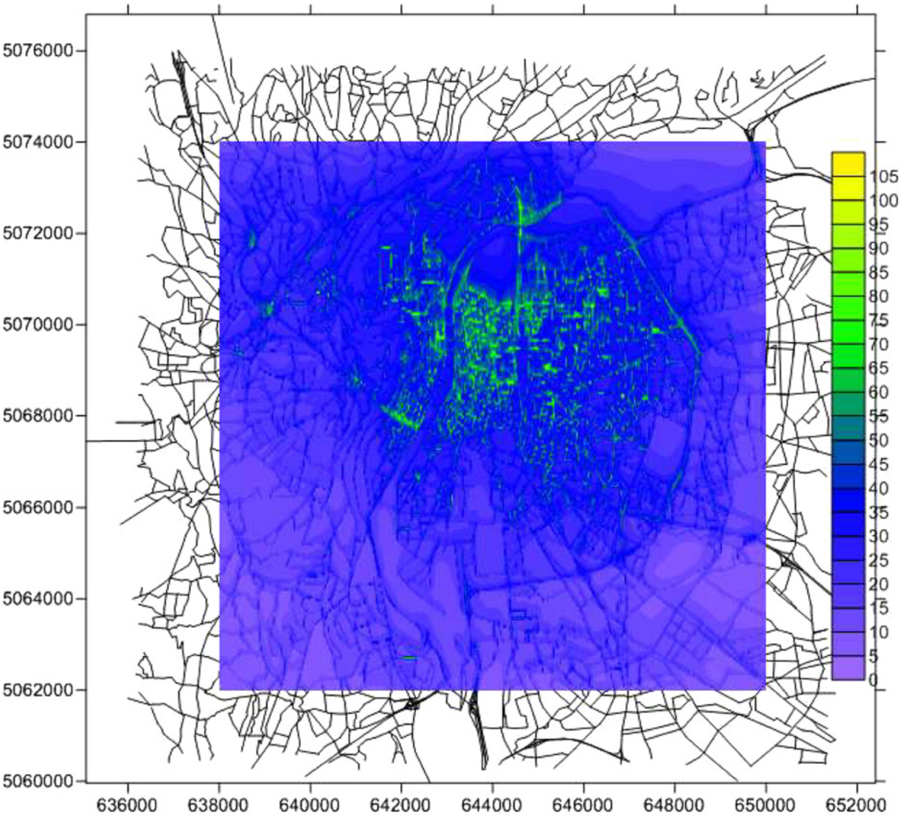

Fig. 4. Normal traffic scenario, PM concentrations ($\mu g/m^3$).

Table 3
Normal traffic scenario: mean concentrations ($\mu g/m^3$).

Id	x	y	$\bar{C}_{NO_2}$	$\bar{C}_{NO}$	$\bar{C}_{O_3}$	$\bar{C}_{PM}$
0	643 580.0	5 068 563.0	209.45	52.21	11.07	36.42
1	645 200.0	5 068 563.0	417.28	74.85	15.38	61.16
2	645 800.0	5 068 563.0	249.41	49.68	13.85	38.04
3	646 000.0	5 068 563.0	334.20	60.28	15.30	48.72
4	647 000.0	5 068 563.0	216.55	43.75	13.66	33.15
5	649 000.0	5 068 563.0	144.98	29.89	13.38	24.24
6	646 393.0	5 070 622.0	254.46	48.53	14.47	38.25
7	645 463.0	5 069 000.0	268.67	52.64	14.08	40.80
8	642 268.0	5 070 180.0	327.80	58.22	15.54	47.73

Table 4
Environmental block scenario: mean concentrations ($\mu g/m^3$).

Id	x	y	$\bar{C}_{NO_2}$	$\bar{C}_{NO}$	$\bar{C}_{O_3}$	$\bar{C}_{PM}$
0	643 580.0	5 068 563.0	68.21	32.78	5.74	18.48
1	645 200.0	5 068 563.0	129.68	38.73	9.24	25.17
2	645 800.0	5 068 563.0	231.15	47.38	13.46	35.75
3	646 000.0	5 068 563.0	327.69	59.49	15.20	47.91
4	647 000.0	5 068 563.0	216.53	43.75	13.66	33.14
5	649 000.0	5 068 563.0	144.98	29.89	13.38	24.24
6	646 393.0	5 070 622.0	252.94	48.34	14.44	38.06
7	645 463.0	5 069 000.0	159.63	38.67	11.39	27.11
8	642 268.0	5 070 180.0	247.09	48.30	14.12	37.66

the region. Finally, it is not surprising that E4 measure yields a good reduction of the pollutant in all the city without the need of traffic reduction, but we want to stress that, if real data are available, the procedure here adopted may be used to quantify the effective impact on air quality of this traffic control strategy.

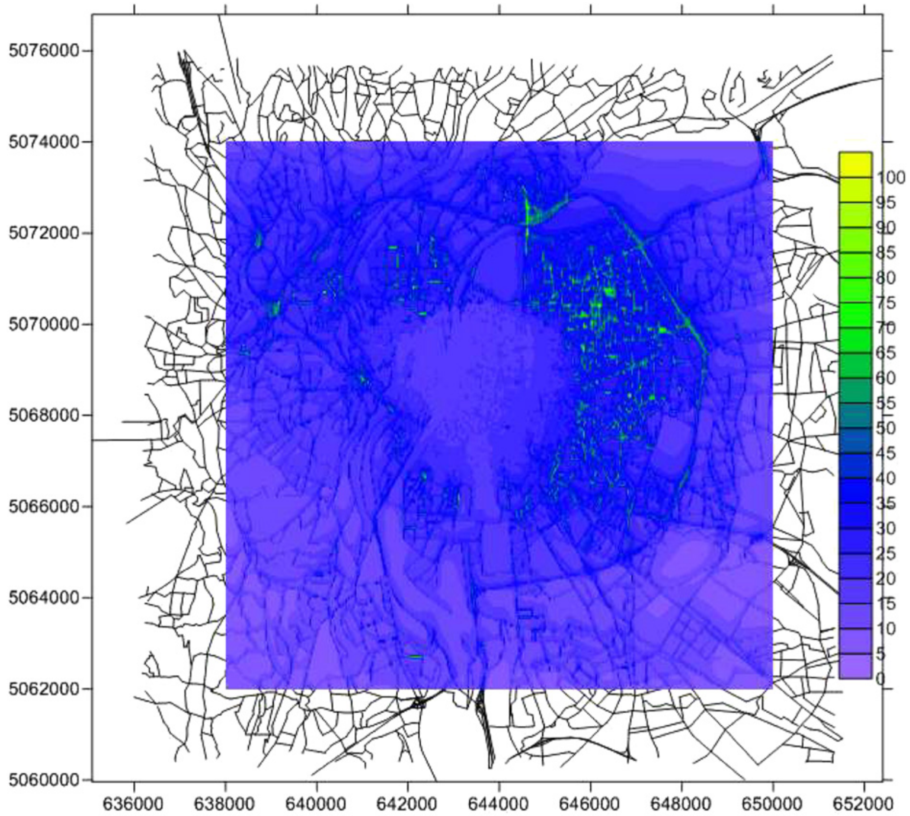

Fig. 5. Environmental block scenario, PM concentrations (μg/m^3).

Table 5
Euro 4 scenario: mean concentrations (μg/m^3).

Id	x	y	$\bar{C}_{NO_2}$	$\bar{C}_{NO}$	$\bar{C}_{O_3}$	$\bar{C}_{PM}$
0	643 580.0	5 068 563.0	98.49	37.25	7.30	31.35
1	645 200.0	5 068 563.0	194.93	47.44	11.34	51.15
2	645 800.0	5 068 563.0	115.68	32.11	9.94	31.95
3	646 000.0	5 068 563.0	155.19	37.70	11.36	40.63
4	647 000.0	5 068 563.0	100.01	28.13	9.81	27.82
5	649 000.0	5 068 563.0	65.67	18.53	9.78	20.57
6	646 393.0	5 070 622.0	117.61	30.64	10.59	32.02
7	645 463.0	5 069 000.0	124.75	33.93	10.15	34.26
8	642 268.0	5 070 180.0	151.92	36.02	11.64	39.77

Table 6
Normal traffic scenario, mean concentrations (μg/m^3), different mesh sizes.

Id	$\Delta x = 1$				$\Delta x = 2$			
	$\bar{C}_{NO_2}$	$\bar{C}_{NO}$	$\bar{C}_{O_3}$	$\bar{C}_{PM}$	$\bar{C}_{NO_2}$	$\bar{C}_{NO}$	$\bar{C}_{O_3}$	$\bar{C}_{PM}$
0	210.33	52.32	11.09	36.54	212.47	52.60	11.15	36.80
1	418.47	74.99	15.40	61.31	421.12	75.31	15.43	61.64
2	250.30	49.79	13.87	38.15	252.51	50.07	13.92	38.42
3	335.36	60.41	15.32	48.87	338.13	60.75	15.36	49.21
4	217.53	43.88	13.68	33.27	220.48	44.25	13.75	33.64
5	145.20	29.92	13.39	24.27	145.71	29.99	13.41	24.33
6	255.37	48.64	14.49	38.36	257.50	48.90	14.53	38.63
7	269.59	52.75	14.10	40.92	271.88	53.03	14.15	41.20
8	329.20	58.39	15.56	47.90	334.44	59.02	15.64	48.55

In Tables 6 and 7 we report, for normal traffic and environmental block scenarios, the mean concentration of pollutants predicted at the same receptors of the previous tables, but with traffic flow simulations performed with a larger space meshsize, namely $\Delta x = 1$ and $\Delta x = 2$ instead of $\Delta x = 0.5$. A similar behavior is obtained for the E4 scenario and the

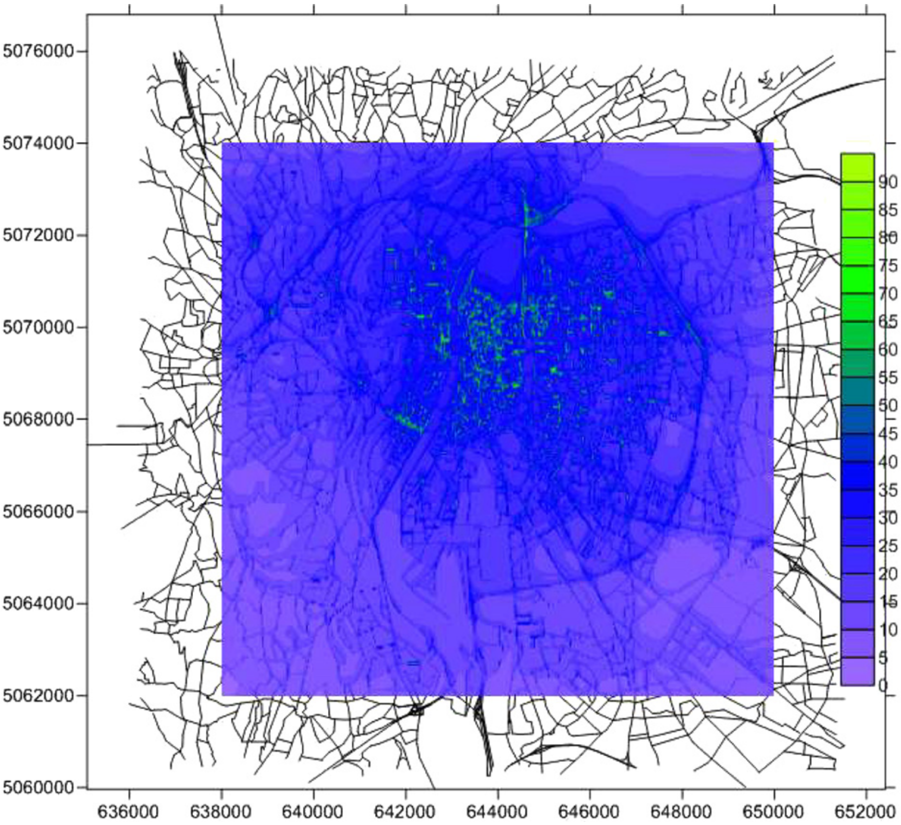

Fig. 6. Euro 4 scenario, PM concentrations ($\mu g/m^3$).

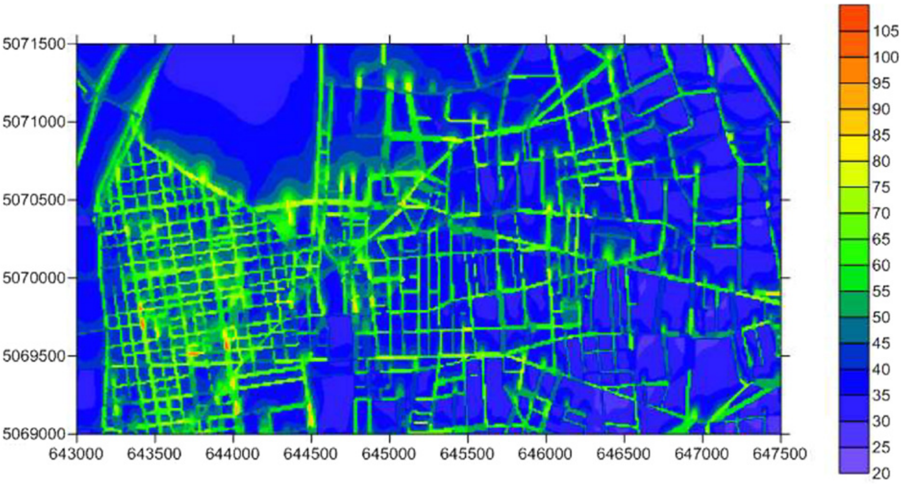

Fig. 7. Normal traffic scenario, detail of PM concentrations ($\mu g/m^3$).

corresponding table is not reported here. As can be seen, the overall computation is moderately affected by this change in the meshsize, as the differences in general are smaller than 1%, except for just one receptor in the BA case, for which differences reach the order of 5%.

We close this section with a brief analysis of the computational costs of our code. In Table 8 we report, for different thread numbers, the computational times spent in the main activities of the code. The computations are performed on a bi-processor computer endowed with two processors Intel Xeon E5520, 2.27 GHz (2009), 4 cores each. Information in Table 8 refers to the simulation of traffic flow (normal traffic scenario) spanning 3600 s, obtained with $\Delta x = 2$, followed by a pollutant dispersion stage.

Please cite this article in press as: S. Berrone, et al., Coupling traffic models on networks and urban dispersion models for simulating sustainable mobility strategies, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.03.054

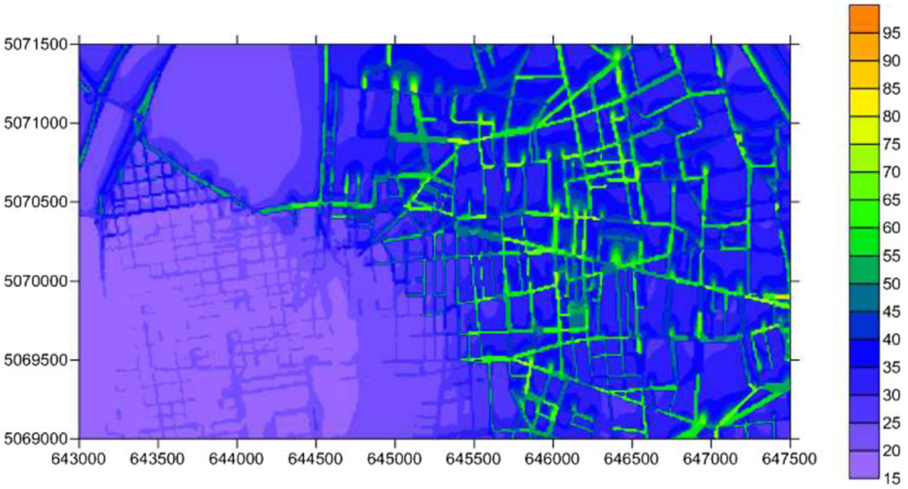

Fig. 8. Environmental block scenario, detail of PM concentrations ($\mu g/m^3$).

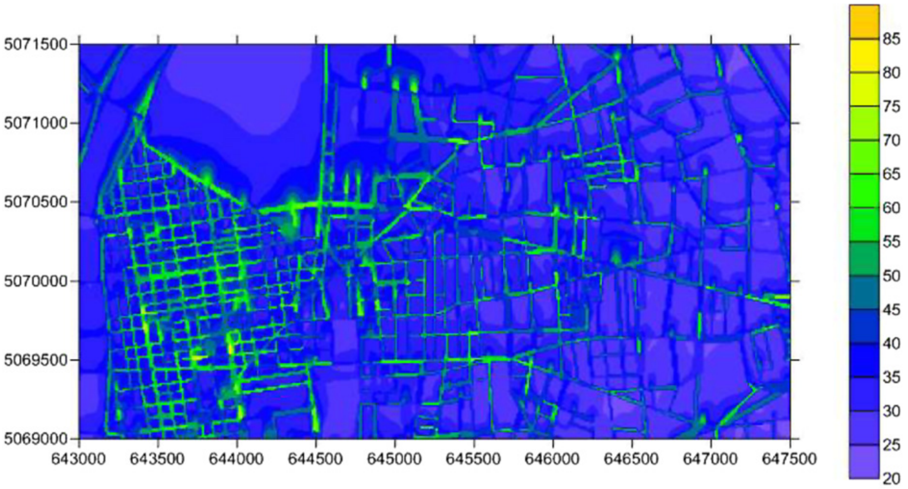

Fig. 9. Euro 4 scenario, detail of PM concentrations ($\mu g/m^3$).

Table 7
Environmental block scenario, mean concentrations ($\mu g/m^3$), different mesh sizes.

Id	$\Delta x = 1$				$\Delta x = 2$			
	$\bar{C}_{NO_2}$	$\bar{C}_{NO}$	$\bar{C}_{O_3}$	$\bar{C}_{PM}$	$\bar{C}_{NO_2}$	$\bar{C}_{NO}$	$\bar{C}_{O_3}$	$\bar{C}_{PM}$
0	68.47	32.82	5.76	18.51	68.99	32.90	5.79	18.58
1	128.85	38.61	9.21	25.07	129.77	38.74	9.24	25.18
2	243.73	48.97	13.73	37.33	246.14	49.27	13.79	37.63
3	334.29	60.29	15.30	48.73	337.16	60.63	15.34	49.09
4	217.51	43.88	13.68	33.27	220.46	44.25	13.75	33.64
5	145.20	29.92	13.39	24.27	145.71	29.99	13.41	24.33
6	253.95	48.46	14.46	38.18	256.09	48.73	14.50	38.45
7	157.79	38.43	11.33	26.88	159.22	38.62	11.38	27.06
8	244.32	47.96	14.06	37.32	246.52	48.23	14.10	37.59

We isolated the following main activities:

- junctions solve (`NodeSolve`): solution of the LP or QP problems at the junctions;
- edges solve (`EdgeSolve`): solution of the conservation law on the edges;
- dispersion (`Sirane`): all the activities concerning the dispersion process over the urban canopy.

In Table 8 we also report the total execution time, including time spent in the initialization phase.

As can be seen from Table 8, in our code, except for the case of just one thread, with $\Delta x = 2$ approximately 63% of the computing time required by traffic simulations is spent in `NodeSolve`, whereas approximately 37% is spent in `EdgeSolve`.

Table 8
Computational times (s) of the main blocks of the code.

# threads	NodeSolve	EdgeSolve	Sirane	Total
1	5920	4540	34	10 496
2	4217	2532	23	6 773
4	2616	1506	15	4 138
6	1758	1010	13	2 782
8	1692	942	12	2 647
12	1481	900	12	2 394

As a consequence, the use of more complex traffic models, such as for example second order models or phase transition models [8], would affect only to a moderate extent the computing time and we do not expect the overall computation suffers too much from the use of such models.

Let us also point out that `NodeSolve` computing time depends on Δx only via the number of time steps performed (Δt is linked to Δx via the CFL condition), whereas in `EdgeSolve` also the complexity of each step depends on Δx. So, if Δx is increased, `EdgeSolve` computing time decreases faster than `NodeSolve`, and the percentage of computing time spent in `EdgeSolve` becomes smaller.

We remark that no effort has been spent to optimize parallelization of the code, so computing times reported in Table 8 have to be intended as a qualitative information, nevertheless confirming that the required computing time on a network involving a whole city may easily be lower than the actual time interval simulated. A hybrid MPI-OpenMP parallelization can largely take advantage from the increased number of processors involved in the computation and reduce the computing times.

5. Conclusions and perspectives

We have performed traffic simulations on a traffic network topology corresponding to a large part of a model city. We have coupled these simulations with a dispersion model implemented in the code Sirane. These results prove that a real traffic simulation with the approach described here is possible, and that computing time on a recent desktop computer may be smaller than actual time simulated. Once different traffic scenarios are modeled, this coupling allows to predict their effects on air quality.

The simulations here presented do not aim at providing quantitative results for the different traffic countermeasures, but they aim at proving the viability of the model here proposed to provide quantitative data if a suitable amount of real traffic and geometry data are available.

In this paper we have introduced some assumptions that should be validated with real local traffic data and network topology. Many of the strategies introduced here to compute some needed parameters can be used to fulfill unavailable data after a suitable tuning process. For a reliable quantitative application of these models, we are planning to perform a validation and a tuning of the model for approximating traffic distribution arrays. To this aim, a large amount of traffic data is needed and its collection is in progress.

Acknowledgments

The authors were supported by Regione Piemonte (Italy) via the project "AirToLyMi: Modeling and simulating sustainable mobility strategies. A study of three real test cases: Turin, Lyon, Milan", (CIPE grant 2006).

References

[1] M. Garavello, B. Piccoli, Traffic Flow on Networks, Springer, 2006.
[2] L. Soulhac, Modélisation de la dispersion atmosphérique à l'intérieur de la canopée urbaine, Ph.D. Thesis, École Centrale de Lyon, 2000.
[3] N. Bellomo, C. Dogbe, On the modelling of traffic and crowds: a survey of models, speculations, and perspectives, SIAM Rev. 53 (2011) 409–463.
[4] G.M. Coclite, M. Garavello, B. Piccoli, Traffic flow on a road network, SIAM J. Math. Anal. (2005).
[5] M.H. Lighthill, G.B. Whitham, On kinematic waves II: a theory of traffic flow on long, crowded roads, Proc. R. Soc. Lond. Ser. A 229 (1955) 317–345.
[6] P.J. Richards, Shock waves on the highway, Oper. Res. 4 (1956) 42–51.
[7] A. Aw, M. Rascle, Resurrection of "second order" models of traffic flow? SIAM J. Appl. Math. 60 (2000) 916–938.
[8] R.M. Colombo, P. Goatin, B. Piccoli, Road networks with phase transitions, J. Hyperbolic Differ. Equ. 7 (2010) 85–106.
[9] H. Holden, N.H. Risebro, A mathematical model of traffic flow on a network of undirectional roads, SIAM J. Math. Anal. 26 (1995) 999–1017.
[10] G. Bretti, R. Natalini, B. Piccoli, Fast algorithms for the approximation of a traffic flow model on networks, Discrete Contin. Dyn. Syst. Ser. B 6 (2006) 427–448.
[11] G. Bretti, R. Natalini, B. Piccoli, Numerical approximations of a traffic flow model on networks, Netw. Heterog. Media 1 (2006) 57–84.
[12] C. D'Apice, R. Manzo, B. Piccoli, Packet flow on telecommunication networks, SIAM J. Math. Anal. 38 (2007) 717–740.
[13] H. Nessyahu, E. Tadmor, Non-oscillatory central differencing for hyperbolic conservation laws, J. Comput. Phys. 87 (1990) 408–463.
[14] G.S. Jiang, D. Levy, C.T. Lin, S. Osher, E. Tadmor, High-resolution nonoscillatory central schemes with nonstaggered grids for hyperbolic conservation laws, SIAM J. Numer. Anal. 35 (1998) 2147–2168.
[15] A.K. Namdeo, J.J. Colls, Development and evaluation of SBLINE, a suite of models for the prediction of pollution concentrations from vehicles in urban areas, Sci. Total Environ. 189–190 (1996) 311–320.
[16] C.A. McHugh, D.J. Carruthers, H.A. Edmunds, ADMS-Urban: an air quality management system for traffic, domestic and industrial pollution, Int. J. Environ. Pollut. 8 (1997) 666–674.

[17] L. Soulhac, C. Puel, O. Duclaux, R.J. Perkins, Simulation of atmospheric dispersion in Greater Lyon: an example of the use of nested models, Atmos. Environ. 37 (2003) 5147–5156.

[18] V. Garbero, P. Salizzoni, S. Berrone, L. Soulhac, Air pollution modelling at the urban scale and population exposure: a case study in turin, in: 31st NATO/SPS International Technical Meeting on Air Pollution Modelling and its Application, 2010.

[19] S. Biemmi, R. Gaveglio, P. Salizzoni, M. Boffadossi, S. Casadei, M. Bedogni, V. Garbero, L. Soulhac, Estimate of boundary layer parameters and background concentrations for pollutant dispersion modeling in urban areas, in: 31st NATO/SPS International Technical Meeting on Air Pollution Modelling and its Application, 2010.

[20] P. Giambini, P. Salizzoni, L. Soulhac, A. Corti, Influence of meteorological input parameters on urban dispersion modeling for traffic scenario analysis, in: 31st NATO/SPS International Technical Meeting on Air Pollution Modelling and its Application, 2010.

[21] L. Soulhac, P. Salizzoni, F.-X. Cierco, R. Perkins, The model SIRANE for atmospheric urban pollutant dispersion: part I, presentation of the model, Atmos. Environ. 45 (2011) 7379–7395.

[22] L. Soulhac, R. Perkins, P. Salizzoni, Flow in a street canyon for any external wind direction, Bound.-Layer Meteorol. 126 (2008) 365–388.

[23] L. Soulhac, V. Garbero, P. Salizzoni, P. Mejean, R. Perkins, Flow and dispersion in street intersections, Atmos. Environ. 43 (2009) 2981–2996.

[24] P. Salizzoni, L. Soulhac, P. Mejean, Street canyon ventilation and atmospheric turbulence, Atmos. Environ. 43 (2009) 5056–5067.

[25] A.A.M. Holtslag, A.P.V. Hulden, A simple scheme for daytime estimates of the surface fluxes from routine weather data, J. Clim. Appl. Meteorol. 22 (1983) 517–529.

[26] B.E.A. Fischer, J.J. Erbrink, S. Finardi, P. Jeannet, S. Joffre, M.G. Morselli, U. Pechinger, P. Seibert, D.J. Thomson, Harmonisation of the preprocessing of meteorological data for atmospheric dispersion models, COST Action 710—Final Report, 1998.

[27] S.P. Arya, Air Pollution Meteorology and Dispersion, Oxford University Press, 1999.

[28] D. Gkatzoflias, C. Kouridis, L. Ntziachristos, Z. Samaras, Copert 4: computer programme to calculate emissions from road transport, Tech. Rep., ETC-ACC, European Topic Centre on Air Pollution and Climate Change, 2007.

[29] Istituto Superiore per la Protezione e la Ricerca Ambientale. http://www.sinanet.isprambiente.it/en (last visited: 14.12.11).

[30] A. Makhorin, GLPK, Free software foundation. http://www.gnu.org/s/glpk/ (last visited: 14.12.11).

[31] L. Di Gaspero, Quadprog++, Project web hosted by Sourceforge.Net. http://sourceforge.net/projects/quadprog/ (last visited: 14.12.11).

[32] D. Goldfarb, A. Idnani, A numerically stable dual method for solving strictly convex quadratic programs, Math. Program. 27 (1983) 1–33.

Please cite this article in press as: S. Berrone, et al., Coupling traffic models on networks and urban dispersion models for simulating sustainable mobility strategies, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.03.054