



**DOTTORATO DI RICERCA IN INGEGNERIA CIVILE PER
L'AMBIENTE ED IL TERRITORIO
XIV Ciclo - Nuova Serie (2013-2015)**

DIPARTIMENTO DI INGEGNERIA CIVILE, UNIVERSITÀ DEGLI STUDI DI SALERNO

**METAHEURISTIC APPROACHES
FOR COMPLETE NETWORK SIGNAL SETTING DESIGN
(CNSSD)**

**Approcci metaeuristici
per la progettazione completa di reti di intersezioni semaforizzate**

CANDIDATO: ING. SILVIO MEMOLI

Relatore:
PROF. ING. G. E. CANTARELLA

Coordinatore:
PROF. ING. V. BELGIORNO

Correlatore:
PROF. ING. S. DE LUCA

Metaheuristic approaches for Complete Network Signal Setting Design (CNSSD)

Copyright © 2016 Università degli Studi di Salerno – via Giovanni Paolo II, 132 – 84084 Fisciano (SA), Italy – web: www.unisa.it

Proprietà letteraria, tutti i diritti riservati. La struttura ed il contenuto del presente volume non possono essere riprodotti, neppure parzialmente, salvo espressa autorizzazione. Non ne è altresì consentita la memorizzazione su qualsiasi supporto (magnetico, magnetico-ottico, ottico, cartaceo, etc.).

Benché l'autore abbia curato con la massima attenzione la preparazione del presente volume, Egli declina ogni responsabilità per possibili errori ed omissioni, nonché per eventuali danni dall'uso delle informazione ivi contenute.

<i>Preface</i>	1
1. <i>Introduction</i>	2
2. <i>Research goals</i>	15
3. <i>Outline and contributions of this thesis</i>	16
4. <i>Signal setting design at a single junction through the application of Genetic Algorithms</i>	21
5. <i>A Hybrid Optimisation Method for Network Signal Setting Design</i>	32
6. <i>Macroscopic vs. Mesoscopic traffic flow models in Signal Setting Design</i> ...	43
7. <i>Network Traffic Control based on a Mesoscopic Dynamic Flow Model</i>	56
8. <i>Network Signal Setting Design: metaheuristic optimisation methods</i>	99
9. <i>Network Signal Setting Design with Stage Sequence Optimisation</i>	139

Preface

This dissertation is submitted for the degree of Doctor of Philosophy at the Department of Civil Engineering (DICIV), University of Salerno. The research described herein is the product of three years of study (2013-2015) under the supervision of Professor Giulio Erberto Cantarella.

Before going through the Chapters that make up this thesis, it should first be noted here that most Chapters consist of papers that have been published, are forthcoming, or have been submitted for publication in a scientific peer reviewed journal or book. Intrinsicly, this format leads to considerable overlap between some of the Chapters, especially regarding parts of their introductions, although every attempt has been made to write each paper in such a way that enables the transition from one chapter to another to be as smooth as possible.

The majority of the research presented here is a result of a close interaction with my supervisor and my research group whose members appear therefore as co-authors of the various papers that form the Chapters of this book.

1. Introduction

1.1 Problem statements (WWWWWHH)

Traffic lights are one of the most common ways to control a road junction network. The design of control variables can be formulated as an optimisation problem, often named Network Signal Setting Design (NSSD).

This thesis predominantly focuses on increasing our understanding on the whole optimisation process which yield to the best Network Signal Settings via a top level (WWWWWHH) principle which organizes this issue into *objectives* (Why is the problem being developed?); *milestones and schedules* (What have be done by When?) *responsibilities* (Who is responsible (variables) for a function? Where are they organizationally located?); *approach* (How will the optimisation be done, technically (algorithms) and managerially (strategies?); *resources* (How much of each resource (time of computation) is necessary?).

“The most important thing to achieve for the Life Cycle Objectives milestone is the conceptual integrity and compatibility of its components above. The element which assures this is the “Feasibility rationale”. It uses an appropriate combination of analysis, measurement, prototyping, simulation, benchmarking, or other techniques, to establish that a system built to the life cycle architecture (LCA) and plans would support the system’s operational concept.” (Boehm, 1996).

1.2 Objectives (Why?)

The continuous challenge for a sustainable and eco-rational transportation system go through demand based strategies or supply based strategies. The former consists in the well-known travel demand management policies, that have been implemented worldwide and have allowed significant results in terms of direct and indirect externalities. Supply based strategies aim to increase the supplied transport capacity, introducing new transport modes, building new road infrastructures, strengthening the transit system and/or optimising the existing transport capacity.

Among all the cited policies, the optimisation of actual transport capacity is the most economical solution, may solve most of the traffic congestion problems and is complementary to any other transportation policy.

At urban level the main framework of the supply design can be defined as Road Network Design Problem (RNDP) and this is addressed to the identification of the optimal configuration of the network in terms of link directions (e.g. road network topology) and to the road traffic control (e.g. Signal Setting Design).

1.3 Milestones and schedules (What? and When?)

Signalised junctions must be distinguished as:

1. Isolated, if the delay can be calculated neglecting the effects of adjacent junctions;
2. interacting, if the delay computation need considering the effects of adjacent junctions that imply the adoption of a traffic flow model to simulate flow propagation.

In terms of signal setting design let further distinguish:

3. single junctions, if considered control variables do not include the offsets;
4. networks of junctions otherwise (in this case also arterial are included).

The signal setting for an isolated junction can be addressed through delay minimisation (Allsop, 1971a) or capacity factor maximisation (Allsop, 1971b, 1976). In particular, in undersaturation conditions, two main problem with a different set of variables are defined: green timing problem and green timing and scheduling problem.

The mathematical programming techniques that can be adopted to evaluate the signal settings optimisation problem for isolated junctions can be grouped into two classes:

- i. Stage based approach: The composition and the sequence of the stages are assumed to be fixed initially and need to be specified in advance while the green times for the stages optimizing a given performance index are calculated. With respect to the graph theory, a signal controlled junction can be represented by a compatibility graph which imposes that no conflicting movements are permitted to move together. Considering a clique as a sub-graph that represents a stage,

the analysis of the sequences of compatibility cliques is conducted. The calculation of the optimal clique sequence allows to enhance the overall efficiency of a junction within a signal cycle (Stoffers, 1968; Zuzarte Tully, 1976; Zuzarte Tully and Murchland, 1978). The computation of the green time for each clique is evaluated by a performance index.

- ii. Phase based (group based) approach: green timing and scheduling can be obtained with respect to the incompatibilities among the streams. The stage structure is variable during optimisation and so these methods can be called group based (phase based in British terminology). Improta and Cantarella (1984) formulate a group based approach for the problem of signal setting optimisation as a Binary Mixed Integer Linear Program (BMILP) which is solved by discrete programming techniques (i.e. a branch and bound technique). Control variables are calculated simultaneously, the number and the composition of stages are an implicit result of the computation procedure. Heydecker and Dudgeon (1987) and Gallivan and Heydecker (1988), starting from Zuzarte's procedure (Zuzarte Tully, 1976) propose a related approach to minimise delay for isolated signal controlled junctions in which the cycle structure is specified by one of the stage sequences (clique sequences). Hence this stage sequence can vary during optimisation process providing much flexibility for the calculation of signal variables. The limit of this method is in adopting Zuzarte's procedure that produces a large number of stage sequences and requires an iterative gradient search method (in which the descent direction at each iteration can be determined) to solve the problem. To overcome this limit Heydecker (1992) introduces a procedure to group all alternative possibilities in a smaller number of equivalent classes with a successor function that represents each of this latter. The green timings and the scheduling are represented by two variables for each group (e.g. start and duration of each group) and the computations can be faster. Silcock (1997) develops a more detailed mathematical model for the group based method and implements the framework in a software called SIGSIGN (Sang and Silcock, 1989) in which double green signals with varying saturation flows are available and the signal timings are designed by minimising critical cycle length, maximising capacity factor or minimising total delay (computed through Webster two terms expression). According to these methods Burrow, 1987, developed Oscady PRO (Phase based Rapid Optimisation), a computer program for optimising phase-based signal timings and calculating capacities, queue lengths and delays

(both queueing and geometric) for isolated traffic signal controlled junctions.

The signal setting optimisation in urban network, is carried out by the coordination of signals and the synchronization of signals. In case of signals coordination, the procedure is based on the optimisation of the offsets (based on delay minimisation), once known the green timings, the cycle length and the scheduling for each junction (based on delay minimisation or maximisation of capacity factor). For traffic signal synchronisation, the procedure is based on the optimisation of the offsets and, at the same time, of the green timings and of the cycle length at each junction (based on delay minimisation). Although it's possible, with the currently available commercial software (such as TRANSYT 14® (Binning et al., 2010) and TRANSYT-7F®), to implement the network delay minimisation by considering as decision variables the green timings at each junction and the offsets among the interacting junctions, literature lacks of methods which include the stages scheduling. This variable makes the problem considerably harder and it is often neglected, assuming the stage sequence as given. As a matter of fact, generally a two-step optimisation structure is adopted: first decisional variables are grouped in two sets, then sequentially optimized. No one-step optimisation method for the simultaneous optimisation of green times, their schedule, and node offsets, the so-called scheduled synchronization, is already available to authors' knowledge.

Formally, existing phase-based methods for single junctions may easily be extended to specify one-step methods for NSSD, since the node offsets may easily be obtained from decision variables, say the start and the end of the green of each approach, and if needed the stage composition and sequence as well. Nonetheless the resulting problem may be hard to solve since several equivalent local optima exist; this condition may quite easily dealt with for a single junction, but it is rather unclear how it can effectively be circumvented for a network (with loops). Thus meta-heuristics, or other optimisation techniques, might perform rather poorly unless the features of the space of solution are further exploited.

Stage-based methods can be used to specify one-step methods for NSSD by explicitly considering the stage composition and sequence as decisions. First, for each junction a set of candidate stages is defined then, the stage sequence can be optimised. Resulting methods are simpler than those derived from phase-based methods, but cannot provide the optimal solution in the general case.

1.4 Responsibilities (Who? and Where?)

This sub-section describes the methodological framework adopted to optimise Network Signal Setting. In following the variables, the constraints and the objective functions (strictly related to traffic flow modelling to consider the interaction among the junctions of the Network) are discussed in more detail.

1.4.1 Variables and Constraints

Assuming that the green scheduling is described by the stage composition and their sequence, let

c be the cycle length, common to all junctions, assumed known or as a decision variable;

for each junction (not explicitly indicated)

t_j be the length of stage j as a decision variable;

t_{ar} be the so-called all red period at the end of each stage to allow the safe clearance of the junction, assumed known (and constant for simplicity's sake);

Δ be the approach-stage incidence matrix (or stage matrix (SM) for short), with entries $\delta_{kj} = 1$ if approach k receives green during stage j and 0 otherwise, assumed known;

l_k be the lost time for approach k , assumed known;

$g_k = \sum_j \delta_{kj} t_j - t_{ar} - l_k$ be the effective green for approach k ;

f_k be the arrival flow for approach k , assumed known;

s_k be the saturation flow for approach k , assumed known;

for each junction i in the network

ϕ_i be the node offset say the time shift between the start of the plan for the junction i and the start of the reference plan, say the plan of the junction 1, with $\phi_1 = 0$. Given such a reference value, all the other $m-1$ node offsets are independent variables where m is the number of junctions in the network.

for each pair of junctions (i, h) in the network

$\phi_{ih} = \phi_h - \phi_i = -\phi_{hi}$ be the link offset between adjacent junctions i and h , needed for computing total delay through a traffic flow model .

Let the junction network be represented by an undirected graph with a node for each junction and an edge for each pair of adjacent junctions (the actually traffic directions are irrelevant). According to this representation if the network is loop less, all the $m-1$ link offsets are independent (as many as the independent node offsets) and may be used as decision variables as

well; arterials are a special case of such a network; if the network contains k independent loops, the number of independent link offsets will be equal to $m-k$; in this case it is better to use the $m-1$ independent node offsets as optimisation variables.

Some constraints were introduced in order to guarantee:

stage lengths being non-negative

$$t_j \geq 0 \quad \forall j$$

consistency among the stage lengths and the cycle length

$$\sum_j t_j = c$$

effective green split being non-negative

$$g_k \geq 0 \quad \forall k$$

this constraint is usually guaranteed by the non-negative stage length, but for a very a short cycle length with regard to the values of all-red period length and lost times, say

$$\sum_j \text{MAX}_k (\delta_{kj} l_k + t_{ar}) \geq c$$

the minimum value of the effective green split

$$g_k \geq g_{min} \quad \forall k$$

Finally let assume

$$c > (\phi_i) \geq 0$$

in order to avoid multiple equivalent solutions e non negative values for node offsets.

1.4.2 Objective Functions

The objective function generally considered in NSSD is the Total Delay (TD) which may be evaluated according to the degree of interaction among the junction composing the network.

For non-interacting approaches (isolated or external junctions) TD may be computed through the two terms Webster's formula (Webster, 1958) as:

$$TD = \sum_k f_k \cdot (0.45 \cdot c \cdot (1 - g_k/c)^2 / (1 - f_k/s_k) + f_k \cdot 0.45 / (s_k \cdot g_k/c) \cdot ((g_k/c) \cdot (s_k/f_k) - 1))$$

For the computation of delay for an interacting approach the cumulated input flows, $Cif_k^j(t)$, and those output, $Cof_k^j(t)$, on the stop line on approach k of junction j, in the subsequent sub-intervals t were compared. The Deterministic Total Delay (DTD_k^j) cumulated in the interval [0, T] for approach k of junction j was then given by the following expression:

$$DTD_k^j = \sum_{t=1..T/\Delta\tau} (Cif_k^j(t \times \Delta\tau) - Cof_k^j(t \times \Delta\tau)) \Delta\tau$$

Thus delay experienced on an interacting approach is a function of the offsets between the timing plans. In fact, such a delay depends on the output flow in the downstream junction which is obtained by starting from the input flow in the upstream junction through the phenomenon of dispersion.

Let s_k^j be the saturation flow on approach k of junction j, the Stochastic and Oversaturation component of Total Delay $SOTD_k^j$ on approach k of junction j is computed using the following expression:

$$SOTD_k^j = \{ [(f_k^j - s_k^j)^2 + (4f_k^j/T)]^{0.5} + (f_k^j - s_k^j) \} T/4$$

and considering the average of the values of the cyclic flow profile along the connecting link arriving at approach k of considered junction j, f_k^j as input flow.

1.4.3 Traffic flow modeling

Whichever optimisation method is employed, Network signal setting design strategies require within-day-dynamic traffic flow modelling. Several approaches can be adopted for within-day dynamics in a transportation network. They may be classified according to the level of detail adopted for representing the traffic systems:

- Microscopic models, which assume disaggregate supply functions and disaggregate flow representation
- Mesoscopic models, which assume aggregate supply functions and disaggregate flow representation
- Macroscopic flow models, which assume traffic flow and supply functions at a high level of aggregation

Macroscopic flow models can be classified according the order of partial differential equations that underlie the model.

Despite the high level of detail reached by using microscopic models, high level of computational performances are requested. Therefore, when simulation is aimed at actual time traffic predictions (relevant to the dynamic description of traffic), macroscopic models are considered as more

appropriate due to the possibility of using mean state variables values (e.g. mean speed, flow rate, etc.) and considering traffic flow as fluid analogy.

At macroscopic level (as described above) the vehicles are not looked at as separate but as aggregate elements.

Generally two main classes of (time continuous) models can be identified:

- Space discrete models (link-based)
- Space continuous models (point-based)

The space discrete models describe the propagation of flows through a link by relationships between whole link variables such as link travel time, link inflows, outflows or link volume (i.e. the number of vehicles on the link) at each point in time. These models are also named as link based. Whole link models (Astarita, 1995; Ran et al., 1997; Wu et al., 1998) are widely used in mathematical programming models for dynamic traffic assignment (DTA) because of their simplicity. However they show several limits: Firstly, as well as the link length increases, they are not able to represent reliable hypocritical congestion conditions (and so the spillback effects) due to the fact that the propagation of flow states along the link is not considered; Furthermore, they cannot be applied at network level because in this case it may be particularly difficult to take out the differences on observed results between the effects of the “behaviours” within individual links and the effects related to the network.

The space continuous models derive from the analogy between vehicular flow and flow of continuous media (e.g. fluids or gasses), yielding flow models with a limited number of partial differential equations that allow to describe the dynamics of variables like the following:

- Density (k): Typical variable from physics adopted by traffic science to express the number of vehicles per kilometer of road.
- Flow rate (q): Represents the number of vehicles that crosses a section per time unit.
- Mean speed (u): Defined as the ratio between the flow rate and the density.

The most elementary continuous traffic flow model is the first order model developed concurrently by Lighthill & Whitham (1955) and Richards (1956), based around the assumption that the number of vehicles is conserved between any two points if there are no entrances (sources) or exits (sinks). This produces a continuous model known as the Lighthill-Whitham-Richards (LWR). This particular model suffer from several limitations. The model does

not contain any inertial effects, which implies that vehicles adjust their speeds instantaneously, nor does it contain any diffusive terms, which would model the ability of drivers to look ahead and adjust to changes in traffic conditions, such as shocks, before they arrive at the vehicle itself. In order to address this limitations Payne (1970) develop a second order continuous model governing traffic flow.

Daganzo (1995) demonstrates that the Payne model, as well as several other second-order models available in the literature, produces false behaviour for some traffic conditions. Specifically, it is noted that traffic arriving at the end of a densely-packed queue would result in vehicles travelling backwards in space, which is physically unreasonable. This is due to the behaviour of vehicles is influenced by vehicles behind them due to diffusive effects. As the differential equations used in LWR model are difficult to solve, especially in situations of high density variations like bottlenecking (in this cases the LWR calls for a shock wave), different approximate techniques have been proposed to solve that equations. Newell (1993), introduces a simplified theory of kinematic waves in which, by using cumulative inflow/outflow curves, the state of flow at an extreme, according to the traffic conditions of another one, can be predicted without considering traffic conditions at intermediate sections. This theory provides a relation between traffic flow q and density k , captured in the triangular shaped fundamental diagram. The author proposes in this way a space discrete model (link based) which provides link travel times complying with the simplified kinematic wave theory.

Consistently with simplified first order kinematic wave theory after Newell, Yperman et.al (2006) present the Link Transmission Model (LTM) in which link volumes and link travel times are derived from cumulative vehicle numbers. Another way to solve the LWR space continuous problem is introduced by Daganzo (1994) through the “Cell Transmission model”, developed as a discrete analogue of the LWR’ differential equations in the form of difference equations which are easy to solve and also take care of high density changes.

In assuming a uniform speed for all the vehicles in a roadway, Daganzo CTM and Yperman LTM cannot fully predict realistic traffic flow behaviour as the platoons keep the same density when moving from the upstream stop-line section to the downstream section, and all vehicles travel at the same free flow speed. For this reason one of the aim of this thesis (as shown in following chapters) was to implement a traffic flow model that took into account a platoon dispersion volume function which describes a more realistic volume from the upstream section to the downstream one when the density is low.

1.5 Approach and Resources (How? and How much?)

An optimisation problem can be dealt with using various approaches, depending both on the specific size of the considered problem and on the real targets which are to be obtained. For this reason, in those cases where it is necessary to reach the optimal solution of a certain problem, an implicit enumeration can be adopted, whether based on a formulation of integer programming, or on the exploitation of combinatorial properties of the problem. A heuristic algorithm (or, simply, a heuristic) must be able to generate a solution in relatively short time. While clearly it is possible to design specific heuristics for any combinatorial optimisation problem, in recent years have gained increasing importance some heuristic approaches of general type, called metaheuristics. The structure and the basic idea of each metaheuristic are essentially fixed, but the implementation of the various components of the algorithm depends on the individual problem. In general, these approaches have evolved through interactions and analogies derived from biological, physical, computer and decision making sciences (Genetic Algorithms, Gas, Hill-Climbing, HC, Simulated Annealing, SA, Ant Colony ACO, etc.). To get a fully operational algorithms from a meta-heuristic requires the specifications of several functions and/or parameters whose meaning depends on the meta-heuristic itself.

In this dissertation we cope with different optimisation strategies applied for Network Signal setting design; for each of them the selection of an algorithm was based on the trade-off between the effectiveness of the algorithm related to the space solution exploration, the parameters to be set and the computational effort depending on the algorithm complexity. To that extent preliminary considerations on algorithm benefits and drawbacks led us to choose the algorithm which best fitted with the optimisation strategy adopted in our several works. As an example of such analysis, in following **Table 1.1** and **Table 1.2** a review on advantages and disadvantages in adopting algorithms like Simulated Annealing (SA) and Genetic Algorithms (Gas) is described.

Table 1.1 - Advantages and limitations in applying SA algorithm.

Simulated Annealing (SA)	
Benefits	Drawbacks
It statistically guarantees finding an optimal solution.	SA is not that much useful when the energy landscape is smooth, or there are few local minima.
It is relatively easy to code, even for complex problems.	There is a trade-off between the quality of the solutions and the time needed to compute them.
SA can deal with nonlinear models, unordered data with many constraints.	More customization work needed for varieties of constraints and have to fine-tune the parameters of the algorithm.
Its main advantages over other local search methods are its flexibility and its ability to approach global optimality.	The precision of the numbers used in implementation have a major effect on the quality of the result.
It is versatile because it does not depend on any restrictive properties of the model.	

Table 1.2 - Advantages and limitations in applying GAs.

Genetic Algorithms (Gas)	
Benefits	Drawbacks
They always give solution and solution gets better with time.	When fitness function is not properly defined, GAs may converge towards local optima.
They support multi-objective optimisation.	Operation on dynamic sets is difficult.
They are more useful and efficient when search space is large, complex and poorly known or no mathematical analysis is available.	GAs is not appropriate choice for constraint based optimisation problems.
The GAs are well suited to and has been extensively applied to solve complex design optimisation problems because they can handle both discrete and continuous variables, and nonlinear objective functions without requiring gradient information.	GAs convergence is very much dependent on initial solution GAs has a "big" stochastic component. This means that many simulations are required to find a statistical convergent solution.

Basically, SA can be thought as GA where the population size is only one. The current solution is the only individual in the population. Since there is only one individual, there is no crossover, but only mutation. This is in fact the key difference between SA and GA. While SA creates a new solution by modifying only one solution with a local move, GA also creates solutions by

combining two different solutions. Whether this actually makes the algorithm better or worse, is not straightforward, but depends on the problem and its representation. It should be noted that both SA and GA share the fundamental assumption that good solutions are more probably found "near" already known good solutions than by randomly selecting from the whole solution space. If this were not the case with a particular problem or representation, they would perform no better than random sampling. What GA does differently here is that it treats combinations of two existing solutions as being "near", making the assumption that such combinations (children) meaningfully share the properties of their parents, so that a child of two good solutions is more probably good than a random solution. Again, if for a particular problem or representation this is not the case, then GA will not provide an advantage over SA. This obviously depends on what the crossover operator is. If the crossover operator is poorly chosen in respect to the problem and its representation, then a recombination will effectively be a random solution. This kind of destructive crossover often results with combinatorial problems if a chromosome directly expresses a solution, and can sometimes be cured by choosing a different representation, where a chromosome is thought of as a "genotype" that only indirectly expresses a solution, a "phenotype". This approach, with two levels of solution representation, has traditionally been specific to GA, but there is no reason why a similar approach could not be applied in SA as well. Also, it should be noted that the relative weight given to mutation and recombination is a crucial parameter affecting what a GA actually does. If mutation is the dominant way of creating new solutions, then the GA in fact acts as a parallelized version of SA, where several solutions are being independently improved.

While it is necessary that a metaheuristic demonstrate good solution quality to be considered viable, having a fast runtime is another critical necessity. If metaheuristics did not run quickly, there would be no reason to choose these approaches over exact algorithms. At the same time, runtime comparisons are some of the most difficult comparisons to make. This is fueled by difficulties in comparing runtimes of algorithms that compiled with different compilers (using different compilation flags) and executed on different computers, potentially on different testbeds.

After all, if there were no limits on execution time, one could always perform a complete search, and get the best possible solution. Most stochastic algorithms can do the same, given unlimited time. In practice, there are always some limits on the execution time.

A key property of stochastic algorithms such as SA and GA is that given more time, they usually provide better solutions, at least up to some limit. If, in an empirical comparison, algorithm A is allowed to use more time than algorithm

B, their solution qualities are no longer comparable, since there is no indication on how good solutions algorithm A would have produced given the same time as algorithm B.

It is, of course, possible that for short time limits, one algorithm outperforms, while for longer time limits, the other one does. There are some hints in literature that this is the case with SA and GA; to be more precise, some comparisons seem to claim that SA is a "quick starter" which obtains good solutions in a short time, but is not able to improve on that, given more time, while GA is a "slow starter" that is able to improve the solution consistently when it is given more time. However, any such claim is in doubt unless it is explicitly stated and tested, by running each algorithm both for a short time and for a long time.

2. Research goals

The acquisition of the state of art has allowed us to identify gaps in the literature on the topic of SSD, in consequence of which it was possible to identify a number of research goals. In particular, the following issues have been identified as object of analysis:

- the effect of using different solution algorithms;
- the effect of using different traffic flow models
- the effect of the stage scheduling as decision variable in the optimisation procedure;

Resulting from the first research goal, i.e. in terms of objective functions, our research has been developed by focusing on:

- the optimisation method as it may be based on mono/multi criteria functions.
- the number of decision variables to be optimised as it depends on the single junction optimisation or on the network optimisation;
- the degree of complexity of the considered layout (i.e. arterial vs. network); in fact in both cases, the number of decision variables is the same even though the network signal setting design is more affected by the interactions of the vehicles as a function of the traffic flow simulation.

In terms of traffic flow modeling several contributions, presented in following chapters, have been proposed which focused on:

- developing a traffic flow model allowing simultaneously for horizontal queuing and platoon dispersion modeling;
- comparing traffic flow models assuming different levels of detail (Macroscopic vs. Mesoscopic).

Finally, complying with the last research objective above listed, the effect of the stage matrix or of the green scheduling on the network optimisation has been addressed through two different strategies as in following described:

1. the explicit enumeration of solutions, in which several feasible stages are first generated then sequenced in different ways for each junction;
2. the implicit enumeration of solutions: all feasible stage matrices are described by duly defined binary variables; the resulting optimisation problem has been solved through a metaheuristic algorithm.

3. Outline and contributions of this thesis

The presented work can be view as a result of a decomposition/recomposition process through which analyzing the complex task concerning the Network Signal Setting Design. To that extent the thesis can be split into three parts which are complementary to each other: Part I, containing Chapters 4 and focusing on the application of the Genetic Algorithms at single junction signal timing with reference to mono-criterion and multi-criteria optimisation; Part II, containing Chapters 5-7 and focusing on the traffic flow modelling, both aiming at developing a traffic flow model allowing simultaneously for modeling different interaction phenomena such as horizontal queuing and platoon dispersion rather than comparing traffic flow models assuming different levels of detail (Macroscopic vs. Mesoscopic Models); Part III, containing Chapters 8-9 and focusing on the efficiency and effectiveness of adopting different types of optimisation strategies: three step optimisation, in which first the stage matrix (stage composition and sequence), the green timings at each single junction are optimised, then the node offsets are computed in three successive steps; (ii) two step optimisation, in which the stage matrix is defined at a first step, then the green timings and the node offsets are computed at a second step (iii) one step optimisation which is a simultaneous optimisation of green times, their schedule, and node offsets, the so-called *scheduled synchronisation*. This chapters could be seen as epilogue of the PhD research project due to the fact that allow to adopt all the previous understanding in the optimisation techniques and traffic flow modeling.

In **Fig.3.1** the outlines of this thesis are shown. Columns represent the three main research goals as previously defined; the arrows imply a chronological order, not an input-output causation.

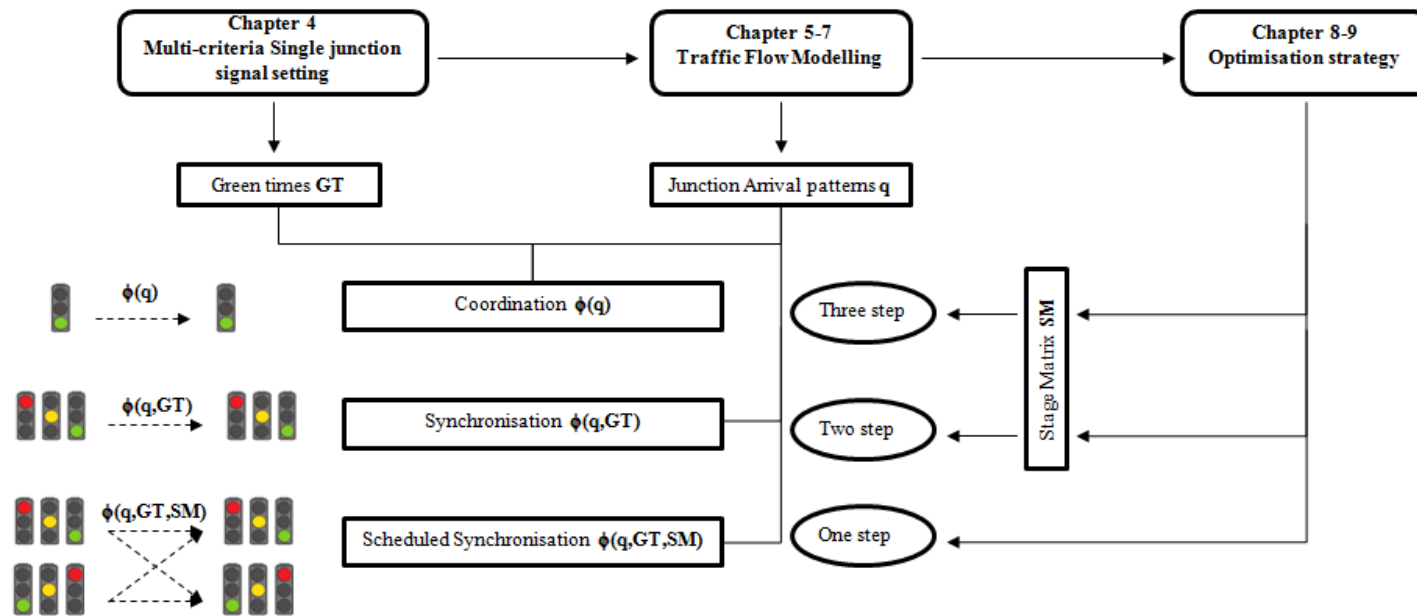


Fig.3.1 - Overview of the thesis

References

- Allsop, R., E., (1971a). Delay-minimizing settings for fixed-time traffic signals at a single road junction, *Journal of the Institute of Mathematics and its Applications*, vol. 8, pp. 164-185.
- Allsop, R., E., (1971b). SIGSET: A computer program for calculating traffic signal settings, *Traffic Engineering and Control*, pp. 58-60.
- Allsop, R.E., (1976). SIGCAP: A computer program for assessing the traffic capacity of signal-controlled road junctions. *Traffic Engineering & Control* 17, pp. 338-341.
- Astarita, V. (1995). Flow propagation description in dynamic network loading models. In *Proceedings of the Fourth International Conference on Application of Advanced Technologies in Transportation Engineering (AATT)*, Capri. American Society of Civil Engineers, pp 599-603.
- Binning, J. C., Crabtree, M. R., & Burtenshaw, G. L. (2010). *Transyt 14 user guide*. Transport Road Laboratory Report nr AG48. APPLICATION GUIDE 65 (Issue F).
- Boehm, B., (1996). Anchoring the software process. *IEEE Software*, vol.13, issue No.04, pp.73-82.
- Burrow, I.J., (1987). OSCADY : a computer program to model capacities, queues and delays at isolated traffic signal junctions: Traffic Management Division, Traffic Group, Transport and Road Research Laboratory, Crowthorne, Berkshire.
- Daganzo, C.F. (1994). The cell-transmission model. Part 2: Network traffic, University of California, Berkeley, California.
- Daganzo, C.F., (1995). Requiem for second-order fluid approximations of traffic flow. *Transport. Res.B*, 29B pp 277–286.
- Gallivan, S., Heydecker, B., G., (1988). Optimising the control performance of traffic signals at a single junction, *Transportation Research B*, pp.357-370.
- Heydecker, B., G., Dudgeon, I., W., (1987). Calculation of signal settings to minimise delay at a junction, *Proceedings of 10th International Symposium on Transportation and Traffic theory*, MIT, Elsevier, New York.
- Heydecker, B., G., (1992). Sequencing of traffic signals. In Griffiths, J.D. (Ed.) *Mathematics in Transport and Planning and Control*, Clarendon Press, Oxford, pp.57-67.
- Improta, G. & Cantarella, G. E., (1984). Control system design for an individual signalized junction. *Transportation Research Part B: Methodological*, 18(2), 147-167.
- Lighthill, M. J., Whitham, G. B. (1955) On kinematic waves. II. A theory of traffic flow on long crowded roads, *Proc. Roy. Soc. London.*, pp. 317–345.
- Lo, H.K., (1999). A novel traffic signal control formulation. *Transportation Research* 33A (6), 433-448.
- Newell, G.F. (1993). A simplified theory of kinematic waves in highway traffic, Part I: General theory, Part II: Queuing at freeway bottlenecks, Part III: Multi-destination flows, *Transportation Research* 27B, pp 281-313.
- Payne, H.J., (1971). Models of Freeway Traffic and Control. In *Simulation Council Proceedings*, 1, pp 51–61.
- Richards, P.I., (1956). Shockwaves on the highway, *Operations Research*, Vol. 4, pp.42-51.
- Sang, A., P., Silcock, J., P., (1989). SIGSIGN User Manual. Steer Davies and Gleave Ltd and Transport Studies Group, University College London.
- Silcock, J., P., (1997). Designing signal controlled junctions for group based operation, *Transportation Research A*, pp.157-173.
- Stoffers, K., E., (1968). Scheduling of Traffic Lights – A new approach, *Transportation Research*, 2, pp.199-234.
- Webster, F. V., (1958). *Traffic Signal Settings*, Road Research Technical Paper, 39, HMSO, London.
- Wu, J.H., Chen, Y., Florian, M., (1998). The continuous dynamic network loading problem: a mathematical formulation and solution method. *Transportation Research B* 32, pp 173-187.
- Yperman, I., Logghe, S., Immers, B., (2005). The Link Transmission Model: an efficient implementation of the kinematic wave theory in traffic networks, in *Proceedings of 10th EWGT Meeting and 16th Mini-EURO Conference*, Poznan, Poland.

Zuzarte Tully, I., M., (1976). Synthesis of sequences for traffic signal controllers using techniques of the theory of graphs, OUEL Report 1189/77, University of Oxford.

Zuzarte Tully, I.,M., Murchland, J.,D., (1978). Calculation and Use of the Critical Cycle Time for a Single Traffic Controller, PTRC Summer Annual Meet, Proc., PTRC-P152, pp.96-112.

4. Signal setting design at a single junction through the application of Genetic Algorithms

Cantarella, G., E., Di Pace, R., Memoli, S., de Luca, S., (2013c). *Signal Setting Design at a Single Junction through the Application of Genetic Algorithms*, in EWGT 2013, 16th meeting of the EURO Working Group on Transportation , published in a special issue of *Advances in Soft Computing (Springer)*, Volume 262, pp.321-331.

Abstract

The purpose of this paper is the application of Genetic Algorithms to solve the Signal Setting Design at a single junction. Two methods are compared: the monocriteria and the multicriteria optimisations. In the former case, three different objectives functions were considered: the capacity factor maximisation, the total delay minimisation and the total number of stops minimisation; in the latter case, two combinations of criteria were investigated: the total delay minimisation and the capacity factor maximisation, the total delay minimisation and the total number of stops minimisation. Furthermore, two multicriteria genetic algorithms were compared: the Goldberg's Pareto Ranking (GPR) and the Non Dominated Sorting Genetic Algorithms (NSGA-II). Conclusions discuss the effectiveness of multicriteria optimisation with respect to monocriteria optimisation, and the effectiveness of NSGA-II with respect to the GPR.

4.1 Problem statement

The Signal Setting Design (SSD) is usually addressed through optimisation models where the decision variables are the signal timings, while the network layout is usually assumed given.

Existing contributions on the SSD, address the following problems: single junction optimisation or network optimisation. Furthermore, SSD can be generally addressed considering the junctions as isolated or interacting within a network. In the former case, the green timings, their scheduling and the cycle length are calculated without considering the influence of upstream on downstream junctions, in the latter case, the interaction between successive junctions must be considered ([8]).

Optimisation models for SSD can be solved through feasible direction algorithms, still these algorithms may fail to find the optimal solution when the objective function is not convex, and /or has several local optimal points. Moreover,

multicriteria optimisation, which is receiving an increasing attention, can hardly be solved through traditional feasible direction algorithms.

On the basis of these points of weakness, in the last years several metaheuristics derived from biological metaphors, such as evolutionary algorithms and swarm intelligence, have been developed first to deal with discrete optimisation, and then extended to cope with continuous optimisation possibly with several optimal points. These methods include genetic algorithms (GAs), differential evolution (DE), ant colony (ACO), bee colony (BCO), bacteria foraging algorithms (BFO) etc.

According to the literature particular attention has been given to the GAs which have been applied to monocriteria network SSD in several cases (see [13]; [21]; [16]) while few contributions can be find with respect to the application of GAs to multicriteria SSD ([24], [5]). Furthermore, in some cases, the multicriteria network SSD has been carried out by using the weighting coefficients in order to combine more objective functions in a unique objective function (see [11]).

On the basis of previous considerations more investigations need to be made on multicriteria optimisation with respect to both the single junction and the network SSD. In particular, this paper aims at investigating the application of genetic algorithms to multicriteria SSD at a single junction (see [9]).

The paper is organised as follows: in Section 4.2, the SSD Background is summarised; in Section 4.3, the Problem is described; in Section 4.4, the Monocriteria and Multicriteria GAs are briefly discussed; in Section 4.5, the Numerical Results are shown and finally in Section 4.6, the Conclusions and research perspectives are summarised.

4.2 Background

In case of single junction SSD in undersaturation conditions, two main problems with different sets of variables may be defined: the green timing and the green timing and scheduling. In the former case, the optimisation variables are the green timings and (probably) the cycle length while the stage matrix is fixed; in the latter case, the stage sequences are also considered as optimization variables.

In accordance with the literature, the green timing problem has been solved by the application of: (i) the Equisaturation principle ([27]); (ii) the total delay minimisation, formulated as a convex programming model ([2] and solved by the program SIGSET described in [4]); (iii) the capacity factor maximisation, formulated as a linear programming (LP) model ([3]; solved by the program SIGCAP described in [6]).

The green timing and scheduling, in undersaturation conditions, can be addressed through the capacity factor maximisation or the total delay minimisation, and several methods based on discrete linear or continuous non-linear optimisation techniques have been proposed ([19]; [6], [7]; [14]). The oversaturation conditions for a single junction can be addressed through total delay minimisation during the entire oversaturated period and not over the cycle length, by looking for the best phase switching strategies, such as the semi-graphical methods ([15]), where the Pontryagin maximum principle is applied to derive analytical solutions of the

optimal trajectories. [20] proposed the bang bang control method which attempted to find an optimal switch over point during.

Finally, most papers address the monocriteria SSD ([26]; [23]), while the multicriteria SSD method has not been discussed in depth ([25]; [22]).

Summing up the aim of this paper is threefold:

- (i) preliminarily investigate the effect of some algorithms parameters on the algorithms effectiveness;
- (ii) compare monocriteria optimisation with multicriteria optimisation;
- (iii) compare two multicriteria optimisation algorithms;

In case (ii), three objectives functions were considered: the capacity factor maximisation, the total delay minimisation and the number of stops minimisation; in case (iii), two combinations of criteria were compared: the total delay minimisation and the capacity factor maximisation, the total delay minimisation and the total number of stops minimisation.

4.3 Problem description

This paper, as stated in the introduction, aims at solving the monocriteria and the multicriteria single junction SSD with given stage matrix through the application of genetic algorithms. The general framework is described below.

4.3.1 Variables

The main definitions and notations are introduced below. Let:

- c be the cycle length, assumed known;
- t_j be the duration of stage j , a decision variable;
- t_{ar} be the so-called all red period at the end of each stage to allow the safe clearance of the junction, assumed known;
- Δ be the approach-stage incidence matrix (or stage matrix for short), with entries $\delta_{kj} = 1$ if approach k receives green during stage j and 0 otherwise, assumed known;
- l_k be the lost time for approach k , assumed known;
- $g_k = \sum_j \delta_{kj} \times t_j - t_{ar} - l_k$ be the effective green for approach k ;
- $r_k = c - g_k$ be the effective red for approach k ;
- q_k be the arrival flow for approach k , assumed known;
- s_k be the saturation flow for approach k , assumed known;
- $(s_k \times g_k) / (c \times q_k)$ be the capacity factor for approach k .

So far, each solution is described by a vector / chromosome with entries / genes given by the stage lengths.

4.3.2 Constraints

Some constraints are introduced in order to guarantee:

stage durations being non-negative

$$t_j \geq 0 \quad \forall j$$

effective green being non-negative

$$g_k \geq 0 \quad \forall k$$

This constraint is usually guaranteed by the stage duration non-negative.

It must be observed that in case of variable cycle length, for too small durations of it, say $c \leq \sum_j \text{MAX}_k(\delta_{kj} \times l_k + t_{ar})$, with regard to the values of all-red period length and lost times.

consistency among the stage durations and the cycle length

$$\sum_j t_j = c$$

the minimum value of the effective green timing

$$g_k \geq g_{min} \quad \forall k$$

A further constraint may be included in order to guarantee the capacity factor must be greater than 1

$$0 \leq (s_k \times g_k) / (c \times q_k) - 1 \quad \forall k$$

Such a constraint may be added only after having checked that the maximum junction capacity factor is greater than one, otherwise a solution may not exist whichever is the objective function/s.

4.3.3 Objective functions

In the monocriteria SSD three objective functions were considered: (i) the Capacity Factor (CF) to be maximised; (ii) the Total Delay (TD) to be minimised; (iii) the Number of stops (NS) to be minimised. In the multicriteria SSD, the optimisation can be carried out with respect to any combination of the above introduced objective functions. In the following, results are described for some combinations of two objective functions only: (v) the Total Delay, to be minimised and the Total Number of stops, to be minimised; (vi) the Total Delay, to be minimised and the Capacity Factor, to be maximised.

All defined multicriteria problems were solved by applying the Goldberg's Pareto ranking method (see [17]) and the NSGA-II (see [12]).

The effectiveness of each optimisation method was evaluated on the base of following performance index: (i) the Capacity Factor (CF);(ii) the Total Delay (TD); (iii) the Number of stops (NS), this indicator was introduced given its effect on the air pollution.

The objective functions in the optimisation problems were computed as follows:

junction capacity factor as

$$CF = \text{MIN}_k (s_k \times g_k) / (c \times q_k) \quad (\text{Eq.4. 1})$$

total delay applying the two terms Webster formula (see [27]) as

$$TD = \sum_k q_k \times (0.45 \times c \times (1 - g_k/c)^2 / (1 - q_k/s_k) + q_k \times 0.45 / (s_k \times g_k/c) \times ((g_k/c) \times (s_k/q_k) - 1)) \quad (\text{Eq.4. 2})$$

total number of stops applying the Akçelik formula ([1]) as

$$NS = \text{MAX}_k 0.9 \times q_k \times (1 - g_k/c) / (1 - q_k/s_k) \quad (\text{Eq.4. 3})$$

It is worth remembering that the delay from the two-terms Webster formula and the number of stops from the Akçelik formula are convex with respect to variables g_k/c , thus ratios t_j/c are the decision variables actually used.

4.4 Monocriteria and multicriteria GAs

GAs search the optimal solution by simulating the evolution of a “population” (of individuals), mimicking the basic principle of bacteria evolution. Each solution is described by a vector of decision variables called a chromosome made up by genes. Usually the most adopted approach for genes generation is based on decodification of binary code for each one, however the degree of accuracy of decodification depends on the landscape of objective functions: in case of stable landscape decodification should be avoided and the direct generation of stages duration should be applied. The optimisation is carried out through an iterative process of random reproduction of the individuals (solutions) in the population based on the fitness function ([18]).

After reproduction each chromosome may be modified by genetic operators, *the crossover* and *the mutation*. This iterative procedure is repeated until some conditions (e.g. number of iterations or of improvement of the best solution) are satisfied. Main parameters to fully specify a GAs are the population size, the crossover probability (or rate), and the mutation probability (or rate).

In case of multicriteria optimisation some major considerations need to be made with respect to procedure described above and regarding the selection criteria based on the fitness function evaluation. In fact, during the iterative procedure, the fitness function can be computed with respect to one or more criteria.

Multicriteria GAs are often called Multiobjective Evolutionary Algorithms (MOEAs), and they can be implemented following one of the below methods: (i) *The Aggregating Functions method*, which defines how all the objectives are combined into a single one by the application of weighting coefficients; (ii) *The Pareto- based methods*, that incorporate the estimation of the Pareto front in the selection mechanism; these methods include the Goldberg's Pareto Ranking, the Non dominated Sorting Genetic Algorithm (NSGA-II), the Niche Pareto Genetic Algorithm etc.

In this paper two multicriteria GAs were implemented: the Goldberg's Pareto Ranking and the NSGA-II. The Goldberg's Pareto Ranking introduces the *rank based* fitness assignment; in particular the fitness function (f) for a given chromosome j is calculated by the *linear ranking* ([4]) approach as follows:

$$f(j) = 2 - SP + 2 (SP - 1) (\text{rank}(j) - 1) / N \quad (\text{Eq.4. 4})$$

where:

SP is the selective pressure fixed to 1.5;

N is the population size;

$\text{rank}(j)$ is computed from the chromosome dominance hierarchy.

This algorithm was compared with the NSGA-II in which an additional criterion is introduced for selecting among solutions with the same ranks. Each solution is attached to a value of *crowding distance* which is computed by the Eulerian distance between the vector of the fitness functions of the given solution and the vector of the best fitness functions (i.e the best value among all solutions); if two or more solutions have the same rank value, selection at successive steps, is based on the best value of the crowding distance.

4.5 Numerical results

This section discusses results obtained for SSD of the T junction with layout described in **Fig.4.1** The stage matrix and the main inputs are shown in **Table 4.1**.

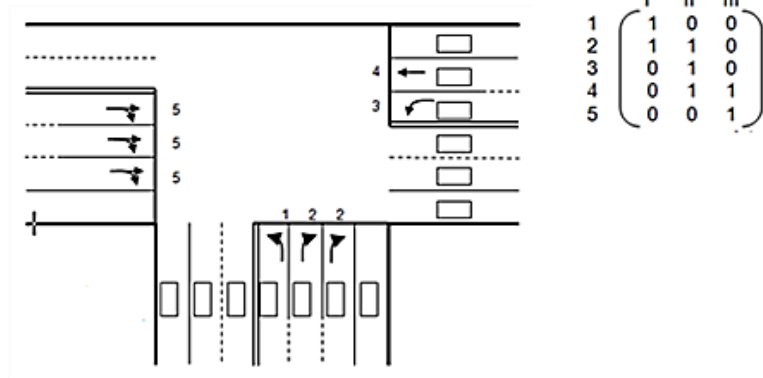


Fig.4.1 - junction layout; stage matrix

Table 4.1 - Junction characteristics.

approach	q_k (pcu/h)	S_k (pcu/h)	C(s)
1	400	1518	
2	350	2122	
3	270	1511	64
4	125	1145	
5	550	4392	

The cycle length was computed in accordance with the Webster indication as in following:

$$c = (1.5 L + 5) / (1 - Y) \quad (Eq.4. 5)$$

where $L = \sum_j l_k(j) \geq 0$, $y_j = q_k(j) / s_k(j) \geq 0$, $Y = \sum_j y_j$, $k(j)$ is the reference approach of stage j .

Before starting the iterative procedure, some parameters that can influence the results of the optimisation, need to be fixed. These are the *population size (pop)*, the *crossover rate and the mutation rate*. Crossover occurs only with a probability **PC** (the crossover rate or crossover probability). When individuals are not subjected to crossover, they remain unmodified. The mutation operator is used to change some elements in selected individuals with a probability **PM** (the mutation rate or mutation probability), leading to additional genetic diversity to help the search process escape from local optimal traps.

First monocriteria SSD problem was addressed. The population size, the crossover rate and the mutation rate were assessed by observing the speed of convergence towards the (sub-)optimal solution. The optimal value of objective function vs. population size (6; 20; 40; 60; 80; 100) were evaluated with regard to each

crossover rate (PC) and/or mutation rate (PM) pair $\{(0.7;0.01), (0.6;0.1), (0.8,0.001), (0.8,0.0001), (0.9,0.001), (0.9,0.0001)\}$ (see for instance in **Fig.4.2**).

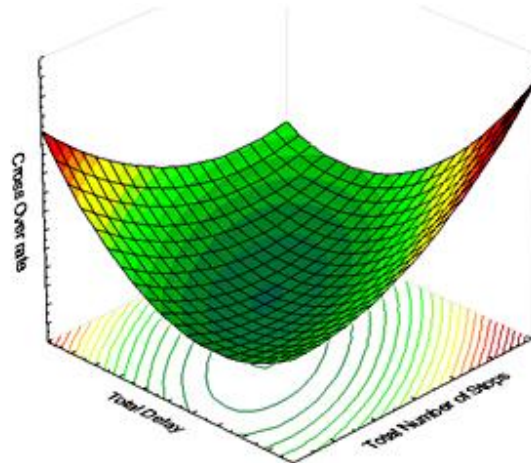


Fig.4.2 - Tridimensional surface of TD&NS w.r.t different values of PC.(for given values of PM and population size).

Numerical results are shown in **Table 4.2** in which each row shows stage durations obtained with one of the objective functions described above, together with values of the other objective functions; best values are in bold. Population size values, the crossover probability/ rate (PC), and the mutation probability/ rate (PM) are given in the table.

Table 4.2 - Results of monocriteria SSD - Population size = 40.

Criterion	PC/PM	TD (veh-h/h)	CF	NS (stops/h)	t ₁ (s)	t ₂ (s)	t ₃ (s)
TD	0.8/0.001	9.14	1.32	459.76	28	20	16
CF	0.9/0.0001	9.15	1.35	459.76	28	20	16
N _{vq}	0.9/0.0001	12.01	1.16	397.87	23	18	23

At first it can be observed that values of stage durations are affected by the criterion considered as objective functions; by comparing obtained results the best value of each indicator is consistent with the optimisation criterion (i.e. in case of CF optimisation, CF = 1.35 and this is the best value of CF among all).

Furthermore, about other indicators, in case of TD minimisation lower value of TD is obtained (TD=9.14 veh); in case of CF maximisation, higher value of capacity factor is reached (CF=1.35); finally, in case of NS minimisation lower value of NS is obtained (NS=397.88 stops/h). Furthermore, the proposed approach was also

validated by comparing GAs results with SIGCAP and SIGSET results. Starting from monocriteria SSD results, multicriteria SSD problem was investigated preliminary assessing the speed of convergence with respect to parameter values (i.e. population size and PC, PM pairs). The GPR and the NSGA-II methods were performed and results are shown in **Table 4.3**.

Table 4.3 - Results of multicriteria SSD (GPR; NSGA-II); Population size = 40.

Opt. Meth.	C* ₁	C* ₂	PC/PM	TD (veh-h/h)	CF	NS (stops/h)	t ₁ (s)	t ₂ (s)	t ₃ (s)
GPR	TD	CF	0.9/0.0001	9.45	1.26	450.93	26	21	17
		NS	0.8/0.001	10.35	1.17	442.08	29	17	18
NSGA-II	TD	CF	0.9/0.0001	9.23	1.33	450.92	27	20	17
		NS	0.8/0.001	10.18	1.23	468.60	31	18	15

*C=criterion considered in the objective function

In particular, three kinds of evaluations can be made: (i) with respect to the effectiveness of each multicriteria optimisation, results obtained by combining TD and CF are better than results obtained by combining TD and NS; (ii) with respect to the comparison between NSGA-II and GPR, more satisfactory results can be obtained by the application of the first one in particular with regard to TD and CF; (iii) finally, with respect to the comparison between monocriteria (based on CF) and multicriteria methods (based on TD&CF), no significant differences can be appreciated between the first one ($TD_{mono} = 9.15 \text{ veh}$; $CF_{mono} = 1.35$; $NS_{mono} = 459.76 \text{ stops/h}$) and the second one (TD&CF: $TD_{multi} = 9.23 \text{ veh}$; $CF_{multi} = 1.33$; $NS_{multi} = 450.92 \text{ stops/h}$); however CF and TD were better in case of monocriteria optimisation than multicriteria optimisation, while a lower (and better) value of total number of stops can be appreciated in case of NSGA- II than monocriteria optimisation.

4.6 Conclusions and research perspectives

This paper is addressed to preliminary investigate the GAs application to the monocriteria or multicriteria SSD for a single junction.

In case of monocriteria optimisation three criteria were considered based on total delay minimisation, capacity factor maximisation and total number of stops minimisation. Monocriteria GAs based on CF maximisation and TD minimisation were compared to SIGCAP and SIGSET methods in order to consolidate their effectiveness of the method. In case of multicriteria optimisation the combination of total delay minimisation and capacity factor maximisation and the combination of total delay minimisation and total number of stops minimisation were considered.

Two algorithms were compared: the GPR and the NSGA-II. On the base of obtained results it can be observed that NSGA-II outperforms GPR. Furthermore, monocriteria GAs were compared to multicriteria GAs (in particular with NSGA-II method). Results show that no significant differences can be appreciated with respect to the values of indicators ($TD_{mono} = 9.14 \text{ veh}$; $CF_{mono} = 1.35$ vs. $TD_{multi} = 9.23$

veh; $CF_{multi}=1.33$) while better results of total number of stops are carried by the application of multicriteria optimisation ($NS_{multi}=450.92$ stops/h) than monocriteria optimisation ($NS_{mono}=459.76$ stops/h).

In future papers, procedures with cycles length optimisation will be addressed. Furthermore obtained results will be compared to other metaheuristics (such as the Simulated Annealing). In particular, the interest in further investigations about application of other metaheuristics to SSD, is related to the complexity of GAs implementation; in fact in this case an high number of parameters (such as crossover rate, mutation rate, population size) must be set preliminarily.

Finally, the integration of multicriteria SSD within the traffic assignment problem with variable demand (see [10]) will be investigated.

References

- [1] Akcelik, R. (1981). *Traffic Signals: Capacity and Timing Analysis*. Research Report ARR No.123. ARRB Transport Research Ltd, Vermont South, Australia.
- [2] Allsop, R., E. (1971a). *Delay minimising settings for fixed time traffic signals at a single junction*. Journal of the Institute of Mathematics and its Applications, 164-185.
- [3] Allsop, R., E. (1972). Estimating the traffic capacity of a signalized road junction. Transportation Research, 245-255.
- [4] Baker, J.E. (1985). *Adaptive selection methods for genetic algorithms*. Proceedings of the 3rd International Conference on Genetic Algorithms and Applications, In: Grefenstette, J.J. (ed.), New Jersey, Lawrence Erlbaum: Hillsdale, 100-111.
- [5] Benekohal, R.F., Waller, S. T. (2003). *Multiobjective traffic signal timing optimization using non-dominated sorting genetic algorithm*. In Intelligent Vehicle Symposium, Proceedings IEEE 9, 198 – 203.
- [6] Cantarella, G., E., Improta, G. (1983). *A nonlinear model for control system design at an individual signalized junction*. Operation Research Italian Society proceedings of the Conference, 709-722.
- [7] Cantarella, G., E., Improta, G. (1988). Capacity Factor or Cycle Time Optimization for Signalized Junctions: A Graph Theory Approach. Transportation Research B.
- [8] Cantarella, G.E.; Di Pace, R.; Memoli, S.; de Luca, S. (2013a). The Network Signal Setting Problem: The Coordination Approach vs. the Synchronisation Approach. Computer Modelling and Simulation (UKSim), 2013 UKSim 15th International Conference, 575-579, doi: 10.1109/UKSim.2013.99.
- [9] Cantarella, G.E.; Di Pace, R.; Memoli, S.; de Luca, S. (2013b). The Application of Multicriteria Genetic Algorithms for Signal Setting Design at a Single junction 8th EUROSIM Congress on Modelling and Simulation, 2013, 472-477, doi 10.1109/Eurosim.2013.85
- [10] Cantarella, G.E., de Luca, S., Di Gangi, M. & Di Pace, R. (2013c). *Stochastic equilibrium assignment with variable demand: Literature review, comparisons and research needs*. WIT Transactions on the Built Environment, Vol. 130, 349-364
- [11] Ceylan, H., Bell, MGH (2005). *Genetic algorithm solution for the stochastic equilibrium transportation networks under congestion*. Transportation Research Part B, 39, 169–185.
- [12] Deb, k., Pratap, A., Agarwal, S., Meyarivan, T. (2002) *A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II*. IEEE Transactions on Evolutionary Computation, vol. 6, NO. 2, 182-197.
- [13] Foy, M. D., Benekohal, R. F. and Goldberg, D. E. (1992). *Signal Timing Determination using genetic algorithms*. Transportation Research Record 1365, 108-115.
- [14] Gallivan, S., Heydecker, B., G. (1988). *Optimising the control performance of traffic signals at a single junction*. Transportation Research B, 357-370.
- [15] Gazis, D., C. (1964). Optimal control of a system of oversaturated intersections. Operations Research, 815-831.

- [16]Girianna, M. and Benekohal, R. F. (2004). *Using Genetic Algorithms to design signal coordination for oversaturated networks*. Intelligent Transportation Systems, 8, 117-129
- [17]Goldberg, D.E. (1989).*Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, MA
- [18]Holland, J.H. (1975). *Adaptation in Natural and Artificial System*. The University of Michigan Press.
- [19]Improta, G., Cantarella, G.E. (1984). Control system design for an individual signalized junction, Transportation Research B, 147-168
- [20]Michalopoulos, P., Stephanopolos, G. (1977). *Oversaturated signal system with queue length constraints*. Transportation Research, 413-421.
- [21]Park, B., Messer, C. J. and Urbanik II, T. (1999). *Traffic Signal Optimization Program for Oversaturated Conditions: Genetic Algorithms Approach*. Transportation Research Record 1683, 133-142
- [22]Putha, R., Quadrifoglio, L. and Zechman (2012). *Comparing Ant Colony Optimization and Genetic Algorithm Approaches for Solving Traffic Signal Coordination under Oversaturation Conditions*. Computer-Aided Civil and Infrastructure Engineering, 27, 14–28.
- [23]Renfrew, D., Xiao-Hua Yu (2012). *Traffic Signal Optimization Using Ant Colony Algorithm*. IEEE, Brisbane, Australia, 1-7.
- [24]Sun, D., Benekohal, R.F., Waller, S. T. (2003a). *Multiobjective traffic signal timing optimization using non-dominated sorting genetic algorithm in Intelligent Vehicle Symposium*, Proceedings IEEE 9, 198 – 203.
- [25]Sun, D., Benekohal, R.F., Waller, S. T. (2003b). *Bi-level Programming Formulation and Heuristic Solution Approach for Dynamic Traffic Signal Optimization*. Computer-Aided Civil and Infrastructure Engineering, 21, 321–333
- [26]Teklu F., Sumalee A., Watling D.P. (2007). *A genetic algorithm approach for optimising traffic control signals considering routing*. Journal of Computer-Aided Civil and Infrastructure Engineering, 22, 31-43.
- [27]Webster, F. V. (1958). *Traffic Signal Settings*. Road Research Technical Paper, 39, HMSO, London.

5. A Hybrid Optimisation Method for Network Signal Setting Design

***Cantarella, G. E., de Luca, S., Di Pace, R., & Memoli, S., (2014). A Hybrid Optimisation Method for Network Signal Setting Design. 11th international congress on Advances in Civil Engineering (ACE), 21-25 October 2014, Istanbul/Turkey.**

* For friendship policy, the authors order is organised in alphabetical order with respect to the affiliation.

Abstract

This paper addresses the Network Signal Setting Design by considering both queue blockage and platoon dispersion in the traffic flow model.

The adopted traffic flow model allows to explicitly representing horizontal queuing phenomena as well as dispersion along a link, outperforming both Cell Transmission model (which is not able to cope with dispersion) and Robertson's dispersion model (where vertical queue formation is considered).

A two-step solution method is adopted: first, signal settings are locally optimised for each junction to get single junction green timing, and then all the absolute offsets are simultaneously optimised to get junction coordination.

Specific meta-heuristic algorithms are investigated to solve the two above problems. On the base of preliminary investigations, the single junction mono-criteria/multi-criteria green timing optimisation is solved by (Pareto based) Genetic Algorithms, whilst the junction coordination by Hill Climbing Algorithm.

In order to evaluate the effectiveness of the proposed method, obtained results are compared with those from a two-step method based on OSCADY PRO® and TRANSYT14® - TRL, both implementing mono-criteria optimisation only.

5.1 Introduction

Traffic models may be classified according to the level of detail adopted for representing the traffic systems:

- Microscopic models, describe both the space-time behaviour of the systems' elements (i.e. vehicles and drivers) as well as their interactions;
- Mesoscopic models, represent traffic by groups of vehicles possibly small, the activities and interactions of which are described at a low detail level;
- Macroscopic flow models, describe traffic at a high level of aggregation as a flow without distinguishing its constituent parts. For instance, the traffic

stream is represented in an aggregate manner using characteristics as flow-rate, density, and velocity.

Macroscopic flow models can be classified according the order of partial differential equations that underlie the model.

Despite the high level of detail reached by using microscopic models, high level of computational performances are requested. Therefore, when simulation is aimed at actual time traffic predictions (relevant to the dynamic description of traffic), macroscopic models are considered as more appropriate due to the possibility of using mean state variables values (e.g. mean speed, flow rate, etc.) and considering traffic flow as fluid analogy.

The main purpose of this paper is to investigate the features of the Macroscopic traffic flow models for obtaining an alternative model in which whether the dispersion along network connecting links and the queues forming phenomena such as spillback can be simulated in a unique framework.

5.2 State of art

At macroscopic level (as described above) the vehicles are not looked at as separate but as aggregate elements.

Generally two main classes of (time continuous)models can be identified:

- Space discrete models (link-based)
- Space continuous models (point-based)

The space discrete models describe the propagation of flows through a link by relationships between whole link variables such as link travel time, link inflows, outflows or link volume (i.e. the number of vehicles on the link) at each point in time. These models are also named as link based. Whole link models (Astarita, 1995; Ran et al., 1997; Wu et al., 1998) are widely used in mathematical programming models for dynamic traffic assignment (DTA) because of their simplicity. However they show several limits: Firstly, as well as the link length increases, they are not able to represent reliable hypocritical congestion conditions (and so the spillback effects) due to the fact that the propagation of flow states along the link is not considered; Furthermore, they cannot be applied at network level because in this case it may be particularly difficult to take out the differences on observed results between the effects of the “behaviours” within individual links and the effects related to the network.

The space continuous models derive from the analogy between vehicular flow and flow of continuous media (e.g. fluids or gasses), yielding flow models with a limited number of partial differential equations that allow to describe the dynamics of variables like the following:

- Density (k): Typical variable from physics adopted by traffic science to express the number of vehicles per kilometer of road.
- Flow rate (q): Represents the number of vehicles that crosses a section per time unit.
- Mean speed (u): Defined as the ratio between the flow rate and the density.

The most elementary continuous traffic flow model is the first order model developed concurrently by Lighthill & Whitham (1955) and Richards (1956), based around the assumption that the number of vehicles is conserved between any two points if there are no entrances (sources) or exits (sinks). This produces a continuous model known as the Lighthill-Whitham-Richards (LWR). This particular model suffer from several limitations. The model does not contain any inertial effects, which implies that vehicles adjust their speeds instantaneously, nor does it contain any diffusive terms, which would model the ability of drivers to look ahead and adjust to changes in traffic conditions, such as shocks, before they arrive at the vehicle itself. In order to address this limitations Payne (1970) develop a second order continuous model governing traffic flow.

Daganzo (1995) demonstrates that the Payne model, as well as several other second-order models available in the literature, produces false behaviour for some traffic conditions. Specifically, it is noted that traffic arriving at the end of a densely-packed queue would result in vehicles travelling backwards in space, which is physically unreasonable. This is due to the behaviour of vehicles is influenced by vehicles behind them due to diffusive effects.

As the differential equations used in LWR model are difficult to solve, especially in situations of high density variations like bottlenecking (in this cases the LWR calls for a shock wave), different approximate techniques have been proposed to solve that equations. Newell (1993), introduces a simplified theory of kinematic waves in which, by using cumulative inflow/outflow curves, the state of flow at an extreme, according to the traffic conditions of another one, can be predicted without considering traffic conditions at intermediate sections. This theory provides a relation between traffic flow q and density k , captured in the triangular shaped fundamental diagram. The author proposes in this way a space discrete model (link based) which provides link travel times complying with the simplified kinematic wave theory.

Consistently with simplified first order kinematic wave theory after Newell, Yperman et.al (2006) present the Link Transmission Model (LTM) in which link volumes and link travel times are derived from cumulative vehicle numbers. Another way to solve the LWR space continuous problem is introduced by Daganzo (1994) through the “Cell Transmission Model”, developed as a discrete analogue of the LWR differential equations in the form of difference equations which are easy to solve and also take care of high density changes.

In assuming a uniform speed for all the vehicles in a road, Daganzo CTM and Yperman LTM cannot fully predict realistic traffic flow behaviour as the platoons keep the same density when moving from the upstream stop-line section to the downstream section, and all vehicles travel at the same free flow speed. For this

reason the aim of this paper is to implement a traffic flow model that takes into account a platoon dispersion volume function which describes a more realistic volume from the upstream section to the downstream one when the density is low.

5.3 Methodology

5.3.1 Traffic flow simulation

In a cell-transmission model a highway is broken down into small sections (cells) and the track of the cell contents (number of vehicles) as time goes on is kept. The record is updated at closely spaced instants (clock ticks) by calculating the number of vehicles that crosses the boundary separating each pair of adjoining cells during the corresponding clock interval. This average flow on the link i from clock tick t to clock tick $t+1$ is given by

$$Y_i(t) = \min \{n_i, \min[Q_i, Q_{i+1}], d_{i+1}[N_{i+1} - n_{i+1}]\} \quad (\text{Eq.5. 1})$$

And it is the result of a comparison between the maximum number of vehicles that can be “sent” by the cell directly upstream of the boundary:

$$S_i(t) = \min\{Q_i, n_i\} \quad (\text{Eq.5. 2})$$

and those that can be “received” by the downstream cell:

$$R_i(t) = \min \{Q_i, d_i[N_i - n_i]\} \quad (\text{Eq.5. 3})$$

n_i : is the number of vehicles on the cell i ;

Q_i : is the maximum flow rate in cell i

d_{i+1} : is the wave speed coefficient of cell $i+1$

N_{i+1} : is the maximum number of vehicles present in the cell $i+1$.

Hence the flow $Y_i(t)$ can be rewritten as:

$$Y_i(t) = \min \{S_i(t), R_{i+1}(t)\} \quad (\text{Eq.5. 4})$$

As shown above, in the CTM equations due to the assumption that all vehicles travel at the same speed (keeping so the same density) for getting in the downstream section, no platoon dispersion might be detected. To overcome this lack, a platoon dispersion volume function is taken into account for describing situations of low densities cell. In particular, by employing the well-known Drake speed-density relationship we introduce a further equation:

$$X_i(t) = k_i(t) * v_0 * \exp[-0.5(k_i(t)/k_m)^2] \quad (\text{Eq.5. 5})$$

$X_i(t)$: is the platoon dispersion volume function;

$k_i(t) = n_i(t)+n_{i+1}(t)/2L$: is the density of cell i and cell i+1 at time t, being L the length of the cell;

v_0 : is the free-flow-speed;

k_m : is the traffic density at maximum flow,

and calculate the flow $Y_i(t)$ as:

$$Y_i(t) = \min \{S_i(t), R_{i+1}(t), X_i(t)\} \quad (\text{Eq.5. 6})$$

5.3.2 Signal Setting Design

With regards to the signal setting design at the single junctions composing the network layout, the multi-criteria (Pareto-based) Genetic Algorithm approach based on total delay minimisation and capacity factor maximisation, has been adopted. The optimisation problems are formulated also taking into account some constraints such as the length of traffic lights (minimum/ maximum values of green/ red lights) and the queue length constraints. The consistency of the stage scheduling in the optimisation process is also included. Once defined the feasible cliques of compatibility i.e. the set of mutually compatible streams, the maximum cliques of compatibility are carried out. These latter represent those cliques to which no further streams can be added (without violating the requirement of the mutual compatibility). Defined, hence, each stage as a maximal clique of compatibility, it is possible to identify a number of possible stage matrices. For each stage matrix, feasible sequences are defined. In general there may be multiple sequences, the choice of the optimum one is carried out from comparison among them, comparing the value of the indicators, once designed control parameters.

The results (signal timing) worked out are then set for each junction and a coordination (minimisation of network total delay reached by absolute offset optimisation) is performed by applying a Hill-Climbing algorithm.

5.4 Optimisation (Algorithms)

5.4.1 Genetic Algorithms (GAs)

In accordance with literature, recently Genetic Algorithms (GAs) have been proposed to solve several kinds of optimisation problems. GAs belong to the Nature inspired meta-heuristics in which an iterative procedure is applied in order to reach at least a good approximation of the optimal solution (such as all heuristics algorithms). These meta-heuristics, based on some biological metaphors such as evolutionary algorithms, have been firstly developed to deal with discrete optimisation and then extended to cope with continuous optimisation and possibly with objective functions with several local optimal points.

GAs seek the optimal solution by simulating the evolution of a “population” of solutions (individuals), in mimicking the basic principle of bacteria evolution. Each solution is described by a vector of decision variables called a chromosome made up of genes. In our case, each gene is representative of the stage duration; the

number of genes in each chromosome depends on the number of stages computed by the use of the relationship between approaches and stages. Optimisation is carried out through an iterative process which is representative of a *reproductive cycle* of the individuals (solutions) in the population. GAs are based on the genetic operators application such as crossover and mutation; while the former promotes the ability of the algorithm to explore the wider areas in order to search for solutions, the latter introduces some (random) diversifications of the same type.

For all these reasons, GAs are characterised by a higher effectiveness in approximating the global solutions in optimisation problems.

GAs for multi-criteria optimisation are often called Multi-objective Evolutionary Algorithms (MOEAs). Among several methods, the most adopted in operative implementations is the *Pareto-based* which incorporates the estimation of the Pareto front in the selection mechanism. This method includes the Goldberg's Pareto Ranking, the Non dominated Sorting Genetic Algorithm (NSGA; Srinivas, 1994), the Non dominated Sorting Genetic Algorithm II (NSGA-II; Deb, 2002) etc. In Goldberg's Pareto Ranking method, the selection (to be parents) of the chromosomes is made on the basis of the ranking of the solutions. In fact, before starting the selection step, for each solution a value of *rank* is associated to the number of times in which the considered solution dominates the others.

In the NSGA-II (Deb and Pratap, 2002) method, an additional criterion is introduced for selecting among solutions with the same ranks. Each solution is attached to a value of *crowding distance* given by the Eulerian distance between the vector of the fitness functions of the solution and the vector of the best fitness function values, each one is defined as the best value among all solutions or in some cases the *reference* fitness function values are defined with regard to the considered criteria. If two or more solutions have the same rank value, selection at successive stages is based on the best value of the crowding distance.

In this paper a multi-criteria approach, performed by NSGA-II, has been applied to solve the signal setting design problem.

5.4.2 Hill Climbing (HC)

The Hill Climbing is a neighborhood-based meta-heuristic algorithm, without memory, which is deterministic in its basic version. The name originates from its ability in generating a succession of solutions exploring the objective function surface which, if plotted, could be thought of as a series of hills and valleys in a multiple-dimensional world.

In this method, starting from an initial solution, successive iterations in the neighborhood are performed until the current solution is not further improved. The algorithm stops when a local minimum/maximum is reached.

Different stochastic variants of this method have been proposed in attempting to endow it with a diversification strategy. For instance, the method can be applied starting from multiple initial solutions randomly generated (as in case of Shot-Gun Hill Climbing), or by varying the structure of the surroundings during the iterations.

In this paper a basic Hill Climbing has been applied for minimising the Total Delay at downstream stopline (as a result of connecting link flow simulation). In order to reduce the risk of being trapped in a poor local optimum, a list of both small and large incremental offset alterations has been set up (such increments are listed as percentages of the cycle time). Thereby, low increments allow to find an approximate local minimum of the Total Delay whilst high increments avoid getting trapped in that minimum.

5.4.3 The numerical application

In this section the performances of the two step method applied on a toy network, build up for simulations, are presented by showing:

- the results obtained for the SSD of the network junctions (the layout together with the stage matrix and the main input data are described in **Fig.5.1**).
- Once defined the connecting link between the two junctions hereunder viewable, their interaction is simulated through the traffic flow model described in **sub-section 5.3.1** (in following the acronym **HOME - CTMD** i.e. Hybrid Optimisation Method - Cell Transmission Model with Dispersion is adopted). The results worked out, in terms of offsets and total delay related to a variable connecting link length, are compared with those obtained by TRANSYT14[®]-PDM (i.e. Platoon Dispersion Model) and then summarised within **Table 5.5**. Graphical outputs are finally shown in **Fig.5.3**.

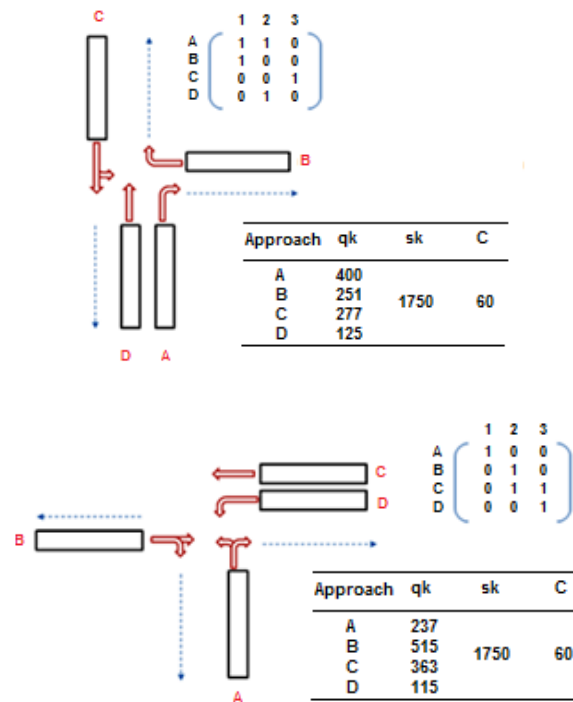


Fig.5.1- Layout, Stage matrix, characteristics for junction 1 (up) and junction 2 (bottom).

All simulations were carried out considering a fixed value of the cycle length computed by the Webster indication. Results are shown in terms of green timing and scheduling and with regard to the optimisation criteria in following tables (from Table 5.1 until Table 5.4).

Table 5.1- Multicriteria optimisation based on Total delay minimisation and Capacity Factor maximisation at junction 1(Pop size = 20; Mutation rate = 0.8; Crossover rate = 0.0001).

Approach	Total Delay [PCU-hr/hr]	Capacity Factor
A	1,32	2,20
B	1,55	2,03
C	1,69	2,11
D	0,99	2,04

Table 5.2- Signal Setting for junction 1.

stage	approach/stage	Scheduling (stage sequence)	Stage duration
			[sec]
1	A,B		23
2	A,D	1,2,3	13
3	C		24

Table 5.3- Multicriteria optimisation based on Total delay minimisation and Capacity Factor maximisation at junction 2(Pop size = 20; Mutation rate = 0.8; Crossover rate = 0.0001).

Approach	Total Delay [PCU-hr/hr]	Capacity Factor
A	1,57	1,81
B	2,89	1,40
C	0,97	2,80
D	2,64	1,48

Table 5.4- Signal Setting for junction 2.

stage	approach/stage	Scheduling (stage sequence)	stage duration[sec]
1	A		19
2	B,C	1,3,2	30
3	C,D		11

Starting from the stages duration obtained by the application of GAs implementations, once fixed five layouts based on connecting link (from junction 1 to junction 2) lengths variability (i.e. 500m; 400m; 300m; 200m; 100m), the Network Total Delay and the absolute offset (**Fig.5.2**) between the signal plans of the interacting junctions are carried out through traffic flow model simulations (**Fig.5.3**).

Table 5.5- Numerical results for TRANSYT14[®] - PDM and HOME - CTMD w.r.t. length variations.

Transyt14 - PDM simulation			HOME - CTMD simulation			Length
Offset [sec]	TD [PCU-hr/hr]	DOS [%]	Offset [sec]	TD [PCU-hr/hr]	DOS [%]	[m]
55	13.83		45	15.12	70	500
44	12.77		34	14.71	78	400
34	11.38	71	22	12.90	92	300
22	9.85		10	23.97	98	200
11	8.57		---	---	oversaturation	100

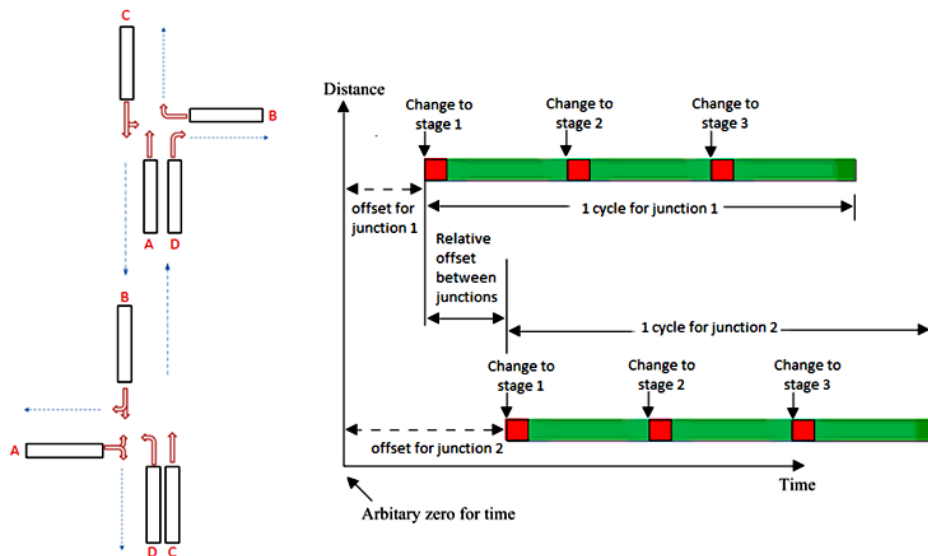


Fig.5.2- Absolute Offsets scheme (green and red bars are not proportional to actual values)

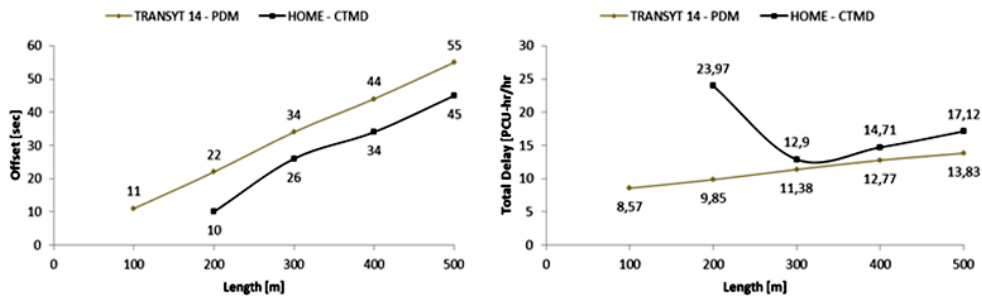


Fig.5.3- Optimal offset (left) and Total Delay (right) w.r.t. Length.

As shown in Fig.5.3 offset increases against link length. This trend may be explained by a decrease of the vehicles cruise time which forces a temporally close start of downstream signal plan respect to the upstream one.

With regards to the Total Delay it is observed that such indicator increases against length for distances greater than 300m. For closer than 300m junctions, TRANSYT14® - PDM underestimates the Total Delay, presumably due to the effect of the horizontal queuing.

5.5 Conclusions and further perspectives

The research presented in this paper aims to provide enhancements for effective traffic flow modelling among interacting junctions in which both dispersion and queue blockage phenomena are represented. An arterial coordination, by adopting a

Cell Transmission Model including a dispersion volume function (presented into this paper with the acronym HOME - CTMD), has been applied.

Satisfactory results, carried out by the application of the model at an arterial level, suggest further advances to network level application and to a complete signal setting design framework where a synchronisation approach, including the simultaneous optimisation of the offsets together with the green timings and scheduling at each junction, is adopted.

With regard to the optimisation algorithm, the Hybrid GAs-Hill Climbing method, applied into this paper, well fit to the above cited complex problem and could be so adopted in attempting to pursue this perspective. However, one of the main problem in GAs application lies in setting the basic parameters of the meta-heuristic, thus in future works the authors will investigate the implementation of self adaptive meta-heuristics.

References

- Astarita, V. (1995). Flow propagation description in dynamic network loading models. In Proceedings of the *Fourth International Conference on Application of Advanced Technologies in Transportation Engineering (AATT)*, Capri. American Society of Civil Engineers, pp 599-603.
- Daganzo, C.F. (1994). The cell-transmission model. Part 2: Network traffic, University of California, Berkeley, California.
- Daganzo, C.F., (1995). Requiem for second-order fluid approximations of traffic flow. *Transportation Research B*, 29B pp 277–286.
- Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., (2002). A Fast and Elitist Multi-Objective Genetic Algorithm : NSGA II. *IEEE Trans. On Evolutionary Computation*, vol.6, pp.182-197.
- Lighthill, M. J., Whitham, G. B. (1955). On kinematic waves. II. A theory of traffic flow on long crowded roads, *Proc. Roy. Soc. London.*, pp. 317–345.
- Newell, G.F. (1993). A simplified theory of kinematic waves in highway traffic, Part I: General theory, Part II: Queuing at freeway bottlenecks, Part III: Multi-destination flows, *Transportation Research 27B*, pp 281-313.
- Payne, H.J., (1971). Models of Freeway Traffic and Control. In *Simulation Council Proceedings*, 1, pp 51–61.
- Ran, B., Boyce, D., (1996). Modelling Dynamic Transportation Networks. *Springer*, Heidelberg.
- Richards, P.I., (1956). Shockwaves on the highway, *Operations Research*, Vol. 4, pp.42-51.
- Srinivas, N., Deb, K., (1994). Multiobjective Optimization Using Non-dominated Sorting in Genetic Algorithms, *Evolutionary Computations*, vol.2, pp.221-248.
- Wu, J.H., Chen, Y., Florian, M., (1998). The continuous dynamic network loading problem: a mathematical formulation and solution method. *Transportation Research B* 32, pp 173-187.
- Yperman, I., Logghe, S., Immers, B., (2005). The Link Transmission Model: an efficient implementation of the kinematic wave theory in traffic networks, in *Proceedings of 10th EWGT Meeting and 16th Mini-EURO Conference*, Poznan, Poland.

6. Macroscopic vs. Mesoscopic traffic flow models in Signal Setting Design

***Cantarella, G. E., de Luca, S., Di Gangi, M., Di Pace, R., Memoli, S., (2014). Macroscopic vs. Mesoscopic traffic flow models in Signal Setting Design. IEEE, International Conference on Intelligent Transportation Systems (ITSC), 8-11 October 2014, Qingdao, China.**

¹Department of Civil Engineering, University of Messina, Italy, EU

²Department of Civil Engineering, University of Salerno Italy, EU

* For friendship policy, the authors order is organised in alphabetical order with respect to the affiliation, then in alphabetical order of the authors name within each affiliation.

Abstract

The simulation of interactions among vehicles approaching signalised junctions, at urban level, is still an open issue. Based on this consideration, this paper aims at comparing the macroscopic and mesoscopic approaches for traffic flow simulation in signalised junctions. In terms of signal setting design, the considered decision variables are the green timings, the scheduling and the offsets thus the three steps optimisation (i.e. each step is referred to the optimisation of each decision variable) has been applied. In particular, the green timings and the scheduling are carried out at single junctions, by multicriteria optimisation, and the offsets are computed by monocriteria optimisation. Finally, the optimisation problems are solved through metaheuristics algorithms.

6.1 Introduction and motivation

The paper focuses on the stochastic traffic control with respect to the dynamic simulation of interactions among signalised junctions. Several researchers have investigated the effectiveness of the on-line methods with respect to the off-line approach (e.g. [1]) and different strategies have been developed as SCOOT (e.g. [2]; [3]), OPAC ([4]), PROLYN ([5]).

Three open issues may be identified in on-line approach: the first one is related to the time periods of the optimisation technique; the second one is related to the optimisation strategies; the third issue is related to the dynamics estimation of traffic flow profiles and queue forming phenomena.

With regards to time-horizon optimisation, different approaches are discussed in literature in order to address the real time applications. They may be classified as:

binary choice approach and sequencing approach. In binary choice approach, time interval is divided into sub-intervals and for each one is made a decision to extend current signal phase or switch to another signal phase. With regards to the sequencing approach, the rolling horizon approach and some extensions are usually applied in literature (see [6]).

In our case every time interval (i.e. each 15 minutes) the decision variables are computed considering the observed flows at 1 leg before the end of the current time interval.

With regards to the optimisation criteria, the coordination approach is adopted, in particular the multicriteria optimisation has been applied to get single junction green timings, then the offsets are computed by monocriteria optimisation (see [7]) The purpose of the paper is to simulate the traffic flow in case of signalised junctions ([8];[9];[10]) by focusing on the flow profiles and queue dynamics in terms of queue length, position of the end of queue, etc.

In general, three main classifications are adopted for traffic flow models: i) the macroscopic models, which describe traffic at a high level of aggregation as a flow without distinguishing its constituent parts (for instance, the traffic stream is represented in an aggregate manner using characteristics as flow-rate, density, and speed); ii) the mesoscopic models, based on the aggregation of vehicles in packets, and all the interactions are described at low detail level; iii) the microscopic models, which focus on the explicit simulation of drivers' behaviour.

In particular, the paper aims at comparing a macroscopic traffic flow model with a mesoscopic traffic flow model.

At a macroscopic level (as described above) two main classes of models can be identified: the *space discrete models* and the *space continuous models*.

The space discrete models, describe the propagation of flows through a link in relationships between whole link variables such as link travel time, link inflows, outflows or link volume (i.e. the number of vehicles on the link) at each point in time. The continuous space models, derive from the analogy between vehicular flow and the flow of continuous media, yielding flow models with a limited number of partial differential equations that allows to describe the dynamics of the variables.

The most straightforward continuous traffic flow model is the first order model developed concurrently by [11] and [12], based on the assumption that the number of vehicles is conserved between any two points if there are no entrances (sources) or exits (sinks). This produces a continuous model known as the Lighthill-Whitham-Richards (LWR). As the differential equations used in the LWR model are difficult to solve, different approximate techniques have been proposed to solve those equations. [13] introduced a simplified theory of kinematic waves in which, by using cumulative inflow/outflow curves, the state of flow at an extreme can be predicted without considering traffic conditions at intermediate sections. Consistently with simplified first order kinematic wave theory after Newell, [14] presented the Link Transmission Model (LTM) in which link volumes and link travel times are derived from cumulative vehicle numbers.

Another way to solve the LWR space continuous problem was introduced by [15] through the Cell Transmission Model (CTM), developed as a discrete analogue of the LWR's differential equations in the form of difference equations which are easy to solve and also consider high density changes. The cell transmission model is usually based on a simplified trapezium form of the fundamental diagram, and provides constant values of the free-flow speed v , at low densities, and backward shockwave speed w , for high densities. As a result, the model does not predict fully realistic traffic behavior (such as the effect of dispersion on arrival flows). To avoid this, several authors propose alternative flow-speed relations to test the effect of different shapes of the fundamental diagram on platoon dispersion by including, in particular, realistic variable cell lengths ([16]; EL-CTM [17]). According to such issue the modified version of CTM, presented into this paper, provides the well-known Drake's flow-density relationship to deal with the effect of platoon dispersion over the links.

Mesoscopic models rely on the consideration of the movement of single users or groups of users and may take varying forms. One form is vehicles grouped into "packets", which are routed through the network (CONTRAM; [18]). The packet of vehicles acts as one entity and its speed on each road (link) is derived from a speed density function defined for that link. Each packet is dealt with as a single entity which experiences the same traffic conditions.

Two approaches can be distinguished: *continuous packets* ([19];[20];[24]), within which vehicles are considered uniformly distributed in space or in time and which are generally identified by two main points i.e. the head and the tail of the packet; *discrete packets* ([18]; [21]; [22]; [23]), where all users are grouped into a single point (for instance the head of the packet) and therefore preside contemporarily in the same position over the link.

The paper focuses on the comparison, on a given artery, between a CTM macroscopic traffic flow model, in a modified version build up to represent the Dispersion of vehicles along the link (CTMD) and a mesoscopic traffic flow model (see [24]).

6.2 Signal Setting Design (SSD)

The arterial signal setting design for interacting junctions involves three decision variables: the green timings, the scheduling and the offsets. Based on these variables, the optimisation problem may be solved through three approaches:

- i. the three steps optimisation, if the decision variables are optimised separately;
- ii. the two steps optimisation, one for green timings and scheduling and one for the offsets;
- iii. finally, the one step optimisation, in which all variables are computed in only one step.

In this paper, the three steps optimisation has been adopted even though the scheduling is not explicitly calculated by the optimisation procedure but the stage matrix is acquired by enumeration approach.

6.2.1 The optimisation problem

a. Basic definitions and notations

Let

c be the cycle length, assumed known;
 t_j be the duration of stage j , a decision variable;
 t_{ar} be the so-called all red period at the end of each stage to allow the safe clearance of the junction, assumed known (and constant for simplicity's sake);
 Δ be the approach-stage incidence matrix (or stage matrix for short), with entries $\delta_{kj} = 1$ if approach k receives green during stage j and 0 otherwise, assumed known;
 l_k be the lost time for approach k , assumed known;
 $g_k = \sum_j \delta_{kj} t_j - t_{ar} - l_k$ be the effective green for approach k ;
 $r_k = c - g_k$ be the effective red for approach k ;
 q_k be the arrival flow for approach k , assumed known;
 s_k be the saturation flow for approach k , assumed known;
 $(s_k \cdot g_k) / (c \cdot q_k)$ be the capacity factor for approach k .
 Φ be the absolute offset.

b. Constraints

Some constraints are introduced in order to guarantee:

stage durations being non-negative

$$t_j \geq 0 \quad \forall j$$

effective green being non-negative

$$g_k \geq 0 \quad \forall k$$

consistency among the stage durations and the cycle length

$$\sum_j t_j = c$$

the minimum value of the effective green timing

$$g_k \geq g_{\min} \quad \forall k$$

c. Objective functions

At single junction, the multicriteria SSD has been computed and the multiobjective function is based on the combination between the total delay, to be minimised and the capacity factor, to be maximised.

At arterial level, the monocriteria optimisation has been carried out, based on the total delay minimisation.

6.2.2 The Algorithms

With regards to single junction, the SSD has been solved by multi-criteria (Pareto-based) Genetic Algorithms.

The offsets optimisation is performed by applying a Hill- Climbing algorithm.

Some specific considerations are also made with respect to the adopted metaheuristics in following sub-sections.

a Genetic Algorithms

Recently Genetic Algorithms (GAs) have been proposed to solve several kinds of optimisation problems.

Each solution is described by a vector of decision variables called a chromosome made up of genes. In our case, each gene is representative of the stage duration; the number of genes in each chromosome depends on the number of stages computed by the use of the relationship between approaches and stages.

GAs are based on the genetic operators application such as crossover and mutation; while the former promotes the ability of the algorithm to explore the wider areas in order to search for solutions, the latter introduces some (random) diversifications of the same type.

In case of multi-criteria optimisation the application of these algorithms is extended including the estimation of the Pareto front in the selection mechanism.

The general approach proposed by Goldberg estimates (i.e. Goldberg's Pareto Ranking) the Pareto front based on the dominance criteria and on the rank of each solution. The successive extension in NSGA-II method (NSGA-II; [25]), introduces an additional criterion for solution in a same Pareto-rank; the criterion is the value of crowding distance given by the Eulerian distance between the vector of the fitness functions of the solution and the vector of the best fitness function values.

In this paper the multicriteria method has been carried out by the application of NSGA-II.

b Hill Climbing

The Hill Climbing is a neighbourhood-based metaheuristic algorithm, without memory, which is deterministic in its basic version. The name originates from its

ability in generating a succession of solutions exploring the objective function surface which, if plotted, could be thought of as a series of hills and valleys in a multiple-dimensional world.

In this approach, starting from an initial solution, successive iterations in the neighbourhood are performed until the current solution is not further improved. The algorithm stops when a local minimum/maximum is reached.

The algorithm approach has been also specified with different extended versions. For instance, the method can be applied starting from multiple initial solutions randomly generated (as in case of Shot-Gun Hill Climbing).

In this paper a basic Hill Climbing has been applied for minimising the Total Delay. In order to reduce the risk of being trapped in a poor local optimum, a list of both small and large incremental offset alterations has been set up (such increments are listed as percentages of the cycle time). Thereby, low increments allow to find an approximate local minimum of the Total Delay whilst high increments avoid getting trapped in that minimum.

6.3 Traffic Flow Models

As in previous described two traffic flow models have been compared: the macroscopic cell transmission model in which the dispersion is explicitly included and a mesoscopic traffic flow model.

6.3.1 The Cell Transmission Model with Dispersion (CTMD)

In a cell-transmission model a highway is broken down into small sections (cells) and the track of the cell contents (number of vehicles) as time goes on is kept. The record is updated at closely spaced instants (clock ticks) by calculating the number of vehicles that crosses the boundary separating each pair of adjoining cells during the corresponding clock interval.

This average flow on the link i from clock tick t to clock tick $t+1$ is given by:

$$Y_i(t) = \min \{n_i, \min[Q_i, Q_{i+1}], d_{i+1}[N_{i+1} - n_{i+1}]\} \quad (\text{Eq.6. 1})$$

And it is the result of a comparison between the maximum number of vehicles that can be “sent” by the cell directly upstream of the boundary:

$$S_i(t) = \min\{Q_i, n_i\} \quad (\text{Eq.6. 2})$$

and those that can be “received” by the downstream cell:

$$R_i(t) = \min \{Q_i, d_i[N_i - n_i]\} \quad (\text{Eq.6. 3})$$

where:

n_i : is the number of vehicles on the cell i ;

Q_i : is the maximum flow rate in cell i

d_{i+1} : is the wave speed coefficient of cell $i+1$
 N_{i+1} : is the maximum number of vehicles present in the cell $i+1$.

Hence the flow $Y_i(t)$ can be rewritten as:

$$Y_i(t) = \min \{S_i(t), R_{i+1}(t)\} \quad (\text{Eq.6. 4})$$

As shown above, in the CTM equations due to the assumption that all vehicles travel at the same speed (keeping so the same density) for getting in the downstream section, no platoon dispersion might be detected. To overcome this lack, the flow propagation has been modeled by employing the well-known Drake speed-density relationship in which the corresponding flow, $X_i(t)$, is computed as follows:

$$X_i(t) = k_i(t) * v_0 * \exp[-0.5(k_i(t)/k_m)^2] \quad (\text{Eq.6. 5})$$

where:

$k_i(t) = [n_i(t) + n_{i+1}(t)]/2L$: is the density of cell i and cell $i+1$ at time t , being L the length of the cell;

v_0 : is the free-flow-speed;

k_m : is the traffic density at maximum flow.

The flow $Y_i(t)$ has been so calculated as:

$$Y_i(t) = \min \{S_i(t), R_{i+1}(t), X_i(t)\} \quad (\text{Eq.6. 6})$$

6.3.2 The Mesoscopic Traffic Flow Model (TRAFFMED – TRAFFIC Analysis and Flow Forecasting MEsoscopic Dynamic model)

This section describes the general formulation of the mesoscopic model (see [24]). In particular, the main characteristics of the loading model are briefly shown in the following.

The simulation is carried out for discrete time intervals assumed, for the sake of simplicity of constant upper bound δ , indicating with τ the current time within the time interval, $\tau \in [0, \delta]$.

Outflow conditions are considered homogeneous on each link and constant for the entire duration of an interval. They are estimated at the beginning of each interval and considered unchanging for the entire duration of the interval. This hypothesis, that the shorter the amplitude of the time interval, the better, also allows that the results are independent of the order in which the packets are moved. Once the outflow characteristics on links for an interval are known, it is possible to track the movements of vehicles on each link depending on the assumptions of the movement rules defined below.

A packet $P \equiv \{\eta, rs, u\}$, is characterised by a departure time η , an origin/destination pair rs , and a vehicle class u .

The network is represented with a graph $G(N,A)$, where N is the set of nodes and A the set of links. Let L_a be the link length, x_a^s the abscissa of section S in link a . We define *running segment* the link portion in the interval $[0, x_a^s]$, *queuing segment* the remaining portion (interval $[x_a^s, L_a]$) (see **Fig.6.1**). The position of section S depends on the length of the queue and is evaluated, at the beginning of each time interval, considering the number of vehicles forming the queue at the end of the previous interval and the occupation factor of each class of the vehicles forming the queue. Moreover, let k_{max}^a be the maximum density on the link and Q_a the capacity of the final section of the link.

In the case of road links, and considering car as a reference for vehicle, either a mono-regime or a two-regime formulation can be used ([26]).

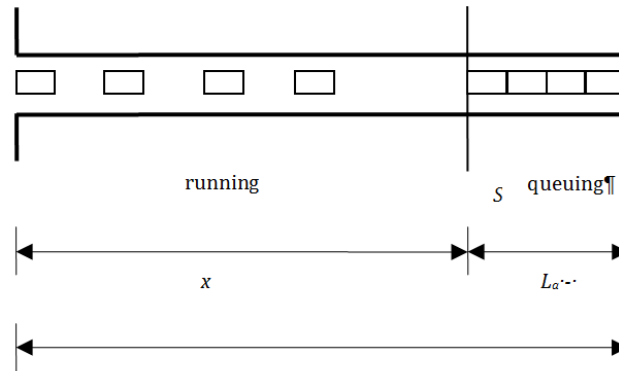


Fig.6.1- Network representation.

Let

v_a^t be the speed on running segment of this link during the interval t ;
 k_{max}^a be the maximum density on the link;
 Q_a be the capacity of the final section of the link.

The following cases may occur:

- i. if $x < x_a^s$ the packet P is located on the running segment, it moves at a speed v_a^t and, within the interval t , it may reach a maximum abscissa x_a^s ; consequently the distance that can be covered on the segment is given by

$$\min\{x_a^s - x, (\delta - \tau) \cdot v_a^t\}$$

- if $x_a^S - x < (\delta - \tau) \cdot v_a^t$, the packet P enters the queuing segment of link a , before the end of the interval, at time $\tau + ((x_a^S - x) / v_a^t)$, otherwise at the end of the interval it remains located on the running segment;
- ii. if $x \geq x_a^S$, the packet P moves on the queuing segment; runoff on this segment is regulated by the capacity of the final section of the link, the length of the queuing segment travelled (within the queue) from packet P by the end of the interval is given by

$$\Delta = ((\delta - \tau) \cdot Q_a) / k_{max}^a$$

if $x + \Delta > L_a$, the packet P leaves the link at time $\tau + ((L_a - x) \cdot k_{max}^a) / Q_a$ during interval t otherwise it remains on the queuing segment.

In terms of the above mentioned, once the length of the running segment of link a^+ is found to be not null, that is $x_{a^+}^S > 0$, it must be verified that the residual capacity of link a is able to allow for the output of packet P .

Let $\tau \in [0, \delta]$ be the time of interval t when packet P is about to leave link a , Q_a be the capacity of the final section of the link a , $U_a(\tau)$ the number of vehicles exiting the link during interval t until time τ , and $n_e(P)$ the number of vehicles making up the packet P ; it may leave link a if $[U_a(\tau) + n_e(P)] / \tau \cdot \delta / T \leq Q_a$ where the factor δ/T is necessary to convert the value to the time unit T considered for the capacity, otherwise it means that the link a is saturated and the packet P is on the link until there is a residual capacity sufficient to enabling the exit.

When a packet P reaches the end of the link a , that is the abscissa L_a , before it can move onto the next link, a^+ , it is necessary to verify that the length of the running segment is not null, that is $x_{a^+}^S > 0$. If $x_{a^+}^S > 0$, packet P may enter link a^+ otherwise it means that the entire length of link a^+ is occupied by a queue and the packet P remains on link a until the queue length on link a^+ is smaller than the length of the link, that is for the time until the condition $L_{a^+} - x_{a^+}^S < L_a$ is verified.

With reference to the introduced notation, it is possible to express the queue length as:

$$L_a - x_a^S$$

In terms of time it is computed as following:

$$((L_a - x_a^S) \cdot k_{max}^a) / Q_a$$

6.4 Application

In order to compare the two traffic flow models, an arterial has been considered. A more detailed description of each junctions is also shown in **Fig.6.2** (upstream junction) and in **Fig.6.3** (downstream junction).

The cycle length has been fixed at 90 seconds thus it has not been included in the optimisation procedure. In figures are also shown the characteristics of the network in terms of flows and saturation flows. All results are shown in **Table 6. 1**.

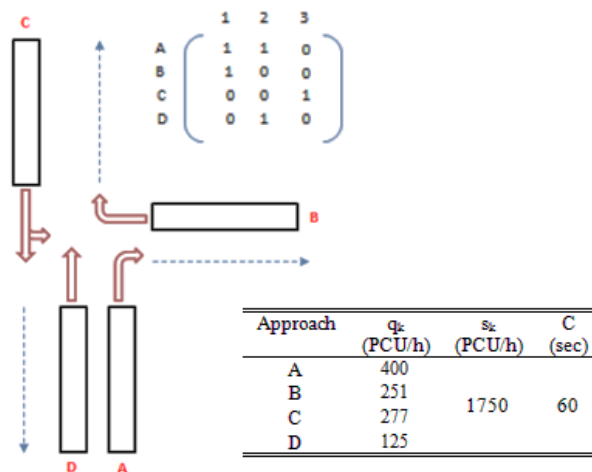


Fig.6.2- Upstream junction.

A sequence of 5 ranges of distance between the successive junctions (see Length field in **Table 6. 1**) has been considered in order to evaluate the sensitivity of the model in simulating an increasing interactions between successive junctions (see [7]).

The performance indicators considered to compare the two models are the Total Delay (*TD*) and the Degree of Saturation (*DOS*).

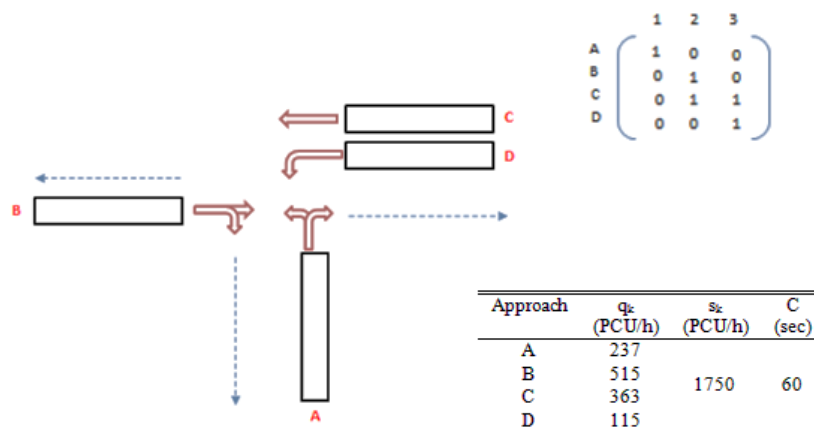


Fig.6.3- Downstream junction.

Signal Setting decision variables have been computed every 15 minutes over a 1 hour simulation. For brevity's sake, results here shown refer to the fourth interval of the simulation.

Table 6. 1 - TRAFFMED vs. -CTM-D

TRAFFMED			CTM-D			Length
Offset	TD	DOS	Offset	TD	DOS	[m]
[s]	[PCU-hr/hr]	[%]	[sec]	[PCU-hr/hr]	[%]	
20	10.05	67.22	45	15.12	70	500
20	9.57	86.54	34	14.71	78	400
69	8.75	92.03	22	12.9	92	300
83	14.04	96.24	10	23.97	98	200
---	---	Overs.*	---	---	Overs.*	100

*Oversaturation conditions

As shown in **Table 6. 1**, TRAFFMED generally outperforms the CTMD in terms of Total Delay (TD) minimisation, especially when lower distances among junctions occur.

Nevertheless, the models may be considered comparable in terms of Degree of saturation (DOS) returned.

6.5 Conclusions and future perspectives

The paper aims at investigating the effectiveness of traffic flow model in simulating the interactions among vehicles at signalised junctions. In particular two different traffic flow models are compared: the mesoscopic TRAFFMED and the Cell Transmission model extended by including dispersion (CTM-D). The sensitivity analyses are carried out by varying the distances between successive junctions. Models may be considered comparable with respect to the degree of saturation, however the mesoscopic traffic flow model outperforms the cell transmission model in terms of total delay.

The future perspectives will be related to the investigation of the proposed approaches at network level also including : i) the extension of multicriteria optimisation for the offsets computation; ii) further analyses on rolling horizon approaches; iii) the computation of the stage matrix by optimisation approaches.

References

- [1] A.J. Miller "A computer control system for traffic network", in *Proc., 2nd Int. Symp. On Theory of Road Traffic Flow*, London, 1965, pp. 201-220.
- [2] P.B. Hunt and D.I. Robertson, R.D. Bretherton and R.I. Winton "SCOOT - a traffic responsive method of coordinating signals" TRRL Laboratory Report 1014, 1981.
- [3] D. I. Robertson and R. D. Bretherton "Optimizing networks of traffic signals in real time: The SCOOT method", *IEEE Transactions on Vehicular Technology. vol. 40 no. 1*, 1991.

- [4] N.H. Gartner "Demand-responsive decentralised urban traffic control" US Department of Transportation DOT/RSPA/DPB -50/81/24, 1982.
- [5] J.J. Henry, J.L. Farges & J.Tufal, "The PRODDYN real time traffic algorithm" in *Proc. of the IFAC Symposium, Baden-Baden*, 1983.
- [6] N. H. Gartner, F. J. Pooran, and C. M. Andrews "Implementation of the OPAC adaptive control strategy in a traffic signal network" in *Intelligent Transportation Systems, 2001. Proc. 2001 IEEE*, pp. 195-200, IEEE.
- [7] G. E. Cantarella, R. Di Pace, S. Memoli, S. de Luca, "The Network Signal Setting Problem: the Coordination approach vs. the Synchronisation approach". In: *IEEE Comp. Soc.. Proc. of UKSim 15th Intern. Conf. on Computer Model. and Sim.*, Cambridge 2013 pp. 575- 579.
- [8] R. Grau, and J. Barcelo "The CARS demand-responsive traffic control system: recent improvements and experiences", in *Proc. of Road Traffic Monitoring and Control, 1994, Seventh International Conference*, pp. 93-96.
- [9] H. K. Lo, , E. Chang, Y. C. Chan "Dynamic network traffic control," *Transportation Research Part A: Policy and Practice*, Elsevier, vol. 35(8), pp. 721-744, 2001.
- [10] M.W. Wey and R. Jayakrishnan "Performance studies of a network adaptive traffic control algorithm via simulation model", in *Intelligent Transportation Systems, Proceedings. 2001 IEEE*, 2001, pp. 663-668.
- [11] M.J. Lighthill and G.B. Whitham "On kinematic waves. II. A theory of traffic flow on long crowded roads", in *Proc. Roy. Soc. London.*, 1955, pp. 317–345.
- [12] P.I. Richards "Shockwaves on the highway", *Operations Research*, Vol. 4, pp.42-51, 1956.
- [13] G.F. Newell "2A simplified theory of kinematic waves in highway traffic, Part I: General theory, Part II: Queuing at freeway bottlenecks, Part III: Multidestination flows", *Transportation Research 27B*, pp 281-313, 1993.
- [14] I.Yperman, S. Logghe and B. Immers, "The Link Transmission Model: an efficient implementation of the kinematic wave theory in traffic networks", in *Proc. Of 10th EWGT Meeting and 16th Mini-EURO Conf., Poznan, Poland*, 2005.
- [15] C.F. Daganzo "The cell-transmission model. Part 2": *Network traffic, University of California, Berkeley, California*, 1994.
- [16] O. Feldman, M.Maher,. "The optimisation of traffic signals using a cell transmission model". In: *Proceedings of the 9th Meeting of the EURO Working Group on Transportation, Bari, Italy*, pp. 503–507, 2002.
- [17] W.Y. Szeto. "Enhanced Lagged Cell-Transmission Model for Dynamic Traffic Assignment". *Transportation Research Record: Journal of Transportation Research Board. No. 2085*, pp. 76-85, *Transportation Research Board of the National Academies, Washington, D.C.*, 2008.
- [18] D.R. Leonard, P. Gower, N.B. Taylor "CONTRAM: Structure of the model". *TRRL Research Report 178, Crowtorne*, 1989.
- [19] M. Di Gangi, "Continuous-flow approach in dynamic network loading" in: *Second International CAPRI Seminar on Urban Traffic Networks*, 1992.
- [20] M. Di Gangi, "Modelling Dynamic Network Loading on Transportation Networks through a Continuous Packet Approach". *Mathematical Modelling of Systems*, vol.2, issue 3, pp. 175-196, 1996.
- [21] M. Dell'Orco, "A dynamic network loading model for mesosimulation in transportation systems," *European Journal of Operational Research*, vol. 175, no. 3, pp. 1447–1454, 2006.
- [22] H. B Celikoglu and M. Dell'Orco "Mesoscopic simulation of a dynamic link loading process," *Transportation Research C: Emerging Technologies*, vol. 15, no. 5, pp. 329–344, 2007.
- [23] M. Di Gangi and P. Velonà "Multimodal mesoscopic approach in modelling pedestrian evacuation" *Transportation Research Record*, vol. 2090, 2009; pp. 51-58.
- [24] M. Di Gangi "Modeling evacuation of a transport system: application of a multi-modal mesoscopic dynamic traffic assignment model", *Transactions on Intelligent Transportation Systems, IEEE*, 12(4), pp.1157-1166, 2011.
- [25] K. Deb, S. Agrawal, A. Pratap and T. Meyarivan "A Fast and Elitist Multi-Objective Genetic Algorithm: NSGA II", *IEEE Trans. On Ev. Comp.*, vol.6, pp.182- 197, 2002.

- [26] M. Papageorgiou, and A. Kotsialos, "Short term traffic forecasting with metanet." *Daccord Short Term Forecasting Workshop* TU Delft, 1998.

7. Network Traffic Control based on a Mesoscopic Dynamic Flow Model

*** Di Gangi¹, M., Cantarella², G. E., Di Pace², R., Memoli², S., (2015). Network Traffic Control based on a Mesoscopic Dynamic Flow Model. Transportation Research Part C, article in press, ISSN:0968090X, DOI: 10.1016/j.trc.2015.10.002.**

¹Department of Civil Engineering, University of Messina, Italy, EU

²Department of Civil Engineering, University of Salerno Italy, EU

* For friendship policy, the authors order is organised in alphabetical order with respect to the affiliation, then in alphabetical order of the authors name within each affiliation.

Abstract

The paper focuses on Network Traffic Control based on aggregate traffic flow variables, aiming at signal settings which are consistent with within-day traffic flow dynamics. The proposed optimisation strategy is based on two successive steps: the first step refers to each single junction optimisation (green timings), the second to network coordination (offsets). Both of the optimisation problems are solved through meta-heuristic algorithms: the optimisation of green timings is carried out through a multi-criteria Genetic Algorithm whereas offset optimisation is achieved with the mono-criterion Hill Climbing algorithm. To guarantee proper queuing and spillback simulation, an advanced mesoscopic traffic flow model is embedded within the network optimisation method. The adopted mesoscopic traffic flow model also includes link horizontal queue modelling. The results attained through the proposed optimisation framework are compared with those obtained through benchmark tools.

7.1 Background and motivation

This paper proposes a Network flow based Traffic Control method carried out through two successive steps: the first step refers to each single junction optimisation (green timings), the second to network coordination (offsets). Furthermore, the adopted traffic flow modelling is a mesoscopic packet based approach, TRAFFMED (Traffic Analysis and Flow Forecasting Mesoscopic Dynamic) developed from an existing model for evacuation plan design (Di Gangi, 2011).

In the following the most used approaches for network traffic control are summarised with respect to the existing literature.

7.1.1 Strategies for Network Signal Setting Design

In general, strategies for signalised junctions may be classified as flow based (A) or vehicle arrival based (B).

In the case of flow based strategies (A), the input data are aggregate variables. Such strategies can be implemented as:

- A.1) fixed timing plans (pre-timed), as long as the settings are constant over periods of the day, depending on the flow evaluation based on historic values;
- A.2) timing plan selection, when the plan is periodically updated in real-time by choosing a plan from among a library of pre-timed signal plans depending on detected flows;
- A.3) timing plan computation, when the plan is periodically updated in real-time by computing a new plan depending on detected flows.

In the case of strategies based on disaggregate input data (B), the signal settings are obtained by detecting vehicles approaching the junction.

a Flow based strategies

One of the most used flow based network control strategies (strategy A.1) is the TRANSYT method, first developed by Robertson in 1969 and then enhanced in successive releases (Vincent et al., 1980; Chard and Lines, 1987, Binning et al., 2010). Such a method is based on traffic modelling through cyclic flow profiles of arrivals at each junction; results of the traffic model are used to compute a performance index, P.I. (i.e. the sum of a weighted linear combination of delays and the number of stops per unit time) assuming signal timings and node offsets as optimisation variables and stage composition and sequence as input. In recent versions apart from the Platoon Dispersion Model (PDM), the Cell Transmission Model (CTM; see Daganzo, 1994) can be used for traffic flow modelling; meta-heuristic algorithms (Hill Climbing or Simulated Annealing) are applied for signal setting optimisation.

Notwithstanding their wide-spread use, fixed timing strategies may perform poorly when actual flows are greatly different from those used for optimisation due to within-day fluctuations, as well as day-to-day variations. In order to overcome such a limitation, some authors have developed network control strategies based on real-time observed flows, despite them generating high operational costs (in terms of sensors, communications, local controllers, etc.). Methods which fall within such a group are SCOOT (Split Cycle Offset Optimisation Technique; Hunt et al., 1981, Bretherton et al., 1998, Stevanovic et al., 2009) and SCATS (Luk, 1984, Stevanovic et al., 2008). These require traffic data to be updated on-line in order to get input flow for the optimiser (such as TRANSYT, Binning et al., 2010) and

arrange green timings, offsets and cycle time duration. These methods match the off-line approach with on-line data (strategies A.2 and A.3).

Whichever optimisation method is employed, Network flow based Traffic Control strategies require within-day-dynamic traffic flow modelling. Several approaches can be adopted for within-day dynamics in a transportation network. They can be classified with respect to: i) users' variables, which can be aggregate variables, such as path link flow and link density or disaggregate variables such as trajectory and the position of a single user; ii) the level of service variables, such as travel time or space mean speed which may refer to a flow of users or to each single user. Three main groups of within-day dynamic models are identified:

1. Macroscopic models where users' behaviour variables are aggregate (link density or entry flows can be obtained from the vehicle position on the link) as well as level of service variables (space mean speed, link performance functions are derived from fundamental diagram);
2. Mesoscopic models where users' behaviour variables are disaggregate (packets of users or single users are considered; link density or entry flows can be obtained from packets/users position on the link) and the level of service variables are aggregate (such as space mean speed; link performance functions are derived from the fundamental diagram);
3. Microscopic models where users' behaviour variables are disaggregate (single users are considered; link density or entry flows can be obtained from the users' position on the link) as well as the level of service variables (time speed and link performance functions are derived from the drivers' behaviour models such as car-following models).

Almost all Network flow based Traffic Control strategies proposed in literature use macroscopic models (see Cantarella et al., 2015 for an in depth analysis of the state of the art), whereas very rarely, microscopic models are used in embedded optimisation methods. Very few authors have investigated the effect of traffic management in a mesoscopic traffic simulation (DYNAMIT; Ben-Akiva et al., 1996; DYNASMART, Jayakrishnan et al., 1994; CONTRAM; Leonard et al, 1989). This paper proposes a Network flow based Traffic Control strategy (NTC TRAFFMED) using a mesoscopic discrete packet modelling. Indeed, major emphasis is not placed on the mesoscopic model but on the integration within the optimisation procedure. To the authors' knowledge such an approach has been never pursued.

The State of the art in terms of mesoscopic traffic flow models is discussed in more details in the following subsection b.

b Vehicle activated strategies

For the sake of completeness, a brief review on network traffic responsive control strategies (i.e. vehicle-actuated control), which require knowledge of vehicle arrivals (strategy B), has been hereunder developed.

DYPIC (Robertson and Bretherton, 1974) is a backward dynamic programming algorithm based on the rolling horizon procedure for the heuristic solution search. Three steps can be identified: first of all, a planning horizon is split into a 'head' period with detected traffic information and a 'tail' period with synthesised traffic information; secondly, an optimal policy is calculated for the entire horizon and implemented only for the 'head' period and finally, for the next discrete time interval, when new detected information is available, the process rolls forward and repeats itself. Gartner (1983) gives a detailed description of the rolling horizon approach in OPAC. Rather than aiming at dynamic programming, OPAC uses a technique named Optimal Sequential Constrained Search to plan for the entire horizon, penalising queues left after the horizon. PRODYN (Henry et al., 1983), also by adopting the rolling horizon approach, optimises timings via a forward dynamic programming (FDP). The FDP records the optimal trajectory of control policy in the planning horizon and the process rolls forward as it is defined in the rolling horizon approach.

UTOPIA (Urban Traffic Optimization by Integrated Automation, Mauro et al., 1989, Mauro, 2002) is a hybrid control system that combines on-line dynamic optimisation and off-line optimisation. This is achieved by adopting a hierarchy structure with a wide-area level and a local level. An area controller continuously generates and provides a reference plan for local controllers. The local controllers then adapt the reference plan and coordinate signals dynamically in adjacent junctions. The rolling horizon approach is then used again by the local controller to optimise single junction design variables. To automate the process of updating reference plans that are generated by TRANSYT, an AUT (Automatic Updating of TRANSYT) module is developed. These traffic data are continuously updated by evaluating the mean of the flows collected by some of the detectors in the network. The data are processed to predict traffic flow profiles for different parts of the day to be used when calculating new reference plans. AUT generates the data to be used in the TRANSYT computation.

7.2 Mesoscopic traffic flow modelling

7.2.1 Brief state of the art

Mesoscopic models may be classified in terms of flow representation. Two different approaches may be identified in the literature: the packet, say the group of users/ the single user representation and the single vehicle representation. A packet of vehicles acts as one entity and its speed on each road (link) is derived from a speed-density function defined for that link. Each packet is dealt with as a single entity which experiences the same traffic conditions. Several authors have proposed methods based on packets of vehicles to reduce the computed effort with

respect to available computer resources. This feature is significant in order to classify papers proposed in the past since computing resources which are currently available make it possible to consider each packet made up of one vehicle only and, therefore, this distinction is no longer available.

A further classification of packet based models can be made in terms of a discrete packet (Leonard et al., 1989; Cascetta et al., 1991; Dell'Orco, 2006; Celikoglu, Dell'Orco, 2007) and a continuous packet (de Romph, 1994; Di Gangi, 1992). In the first case, a discrete distribution of vehicles in the packet is considered. All users are grouped in a single point (for instance the head of the packet) and, therefore, are located contemporarily at the same position over the link. In the continuous packet based approach, the vehicles are considered uniformly distributed (in time or space) in the packet, which is thus identified by two main points i.e. the head and the tail of the packet. Due to their inherent difficulties related to numerical problems of internal consistency when instantaneous density variations between adjacent simulation steps occur, only a few authors in literature (Di Gangi, 1992,1996) have investigated continuous packet models and the most relevant contributions rely on discrete packet methods. It is worth noting that Continuous packets are relevant only when packets are made up of more than one vehicle.

As alternatives to packets representation, DYNAMIT (Ben-Akiva et al., 1996) moves individual vehicles along segments according to speed-density (Underwood, 1961; Drake et al., 1967; Drew, 1968) relationships and a queuing model whereas DYNASMART (Jayakrishnan et al., 1994) represents individual vehicles by point particle. The latter also uses macroscopic speed-density relationships but adopts a more detailed representation of signalised junctions to model delays at these facilities. A very similar product to DYNASMART is the release 1 of INTEGRATION (Van Aerde, and Yagar, 1988). In the case of signalised junctions some authors have investigated the use of stochastic queue servers at the nodes. During each simulation interval, vehicle dynamics are defined in accordance with the macroscopic speed-density relationship and a queue-server at the nodes accounts for delays caused by traffic signals, downstream capacity limits and interaction with additional traffic. Examples of such models are the traffic flow models in DYNAMIQ (Mahut, Florian et al., 2002) and MEZZO (Burghout, 2004).

Mesoscopic traffic flow models may be further classified in terms of queuing representation in link based models, which are in turn grouped further depending on the performance function, and node based models (Celikoglu and Dell'Orco, 2007; Celikoglu et al., 2009), usually referring to the models which consider the flow splitting rates. In particular, link based models are also divided in i) travel time models (CONTRAM, Taylor, 2003; Bliemer, 2006; Astarita, 1996; Adamo et al., 1999) and ii) exit function models (such as the M-N model proposed by Merchant and Nemhauser, 1978a, 1978b; the CTM model developed by Daganzo 1994, 1995; the point queue- PQ- model proposed by Smith, 1983 and the model proposed by Kuwahara and Akamatsu, 1997).

Travel time models may also be classified depending on the link representation: by considering the whole link so that the travel time is the sum of running travel time

and queuing time (Jayakrishnan et al., 1994) or by considering the link divided in a running part (free flow moving) and a queuing part. In order to capture the effect of dynamic horizontal queuing and then of the spillback simulation in the proposed model, a link based approach (falling in the category of travel time models) considering the link split in a running and a queuing part is adopted. In particular, unlike Ran and Boyce (1996), in which the representation of the running and the queuing part of the link is instantaneous (not time dependent), the adopted mesoscopic traffic flow model was based on a 'variable' queue lengths representation which is affected by the previous simulation (see He, 1997). Such a representation makes it possible to clearly identify the boundary of the two parts of the link and then to avoid unreliable delay computations. Unlike the model presented in He (1997), based on fixed outflow capacity, the proposed model, as in Bliemer (2006) considers a dynamic queue-running part representation by changing outflow capacities ('dynamic queue approach'). Finally, as in Ben-Akiva et al. (1996), Jayakrishnan et al. (1994), Celikoglu and Dell'Orco (2007), Bliemer (2006), in the proposed contribution a path choice model is implemented.

7.2.2 Adopted mesoscopic traffic flow model

The adopted mesoscopic traffic model is based on discrete packet representation and each packet is made up of a single vehicle only. It is a link based model, falling within the class of the travel time models, and makes it possible to explicitly represent: horizontal queues; proportional traffic leaving a node through an explicit path choice model; the dynamic generation of the path-flows incidence matrix.

Furthermore, in the considered traffic flow model a speed-density relationship is adopted to evaluate the speed of a packet on the running part only and it is not used as a cost function. In this way, double counting the delay experienced by the vehicles is avoided, firstly due to the traversal speed (which is computed as a function of the actual density) and then due to the queuing delay (which is obtained as a result of the simulation).

Moreover, as in Bliemer (2006), we consider a dynamic queue-running part representation by including link capacity restrictions which are due to the signal timings variations over the time steps of simulation.

A more detailed overall model description is shown in **sub-section 7.2.3**.

It is worth noting that this paper does not aim to propose a mesoscopic model formalisation but our main contribution relies on embedding the mesoscopic traffic flow simulation within a network control strategy.

7.2.3 Proposed NTC strategy

The goal of this paper is twofold:

1. proposing a Network Traffic Control (NTC) procedure based on mesoscopic traffic flow modelling (TRAFFMED); the control procedure is based on two steps: in the first step, decision variables are the green timings and a multi-criteria optimisation is adopted; in the second step,

decision variables are the offsets and a mono-criterion optimisation is applied; the proposed optimisation procedure is a further development of one described in Cantarella et al (2015) which is based on macroscopic traffic flow modelling;

2. testing a discrete packet approach for mesoscopic traffic flow modelling, TRAFFMED, aiming to consider the link queuing dynamic simulation (including spillback simulation).

The paper is organised as follows: in **section 7.3** the Network Traffic Control problem in terms of objective functions, decision variables, constraints and metaheuristic solution algorithms are shown; in **section 7.4** a detailed description of the proposed mesoscopic traffic flow model is provided; in **section 7.5** the results from the application of the proposed methodology on a toy network are shown and compared with a benchmark tool; in **section 7.6** some considerations are expressed regarding the effects and the developments of control strategy on an urban level.

7.3 Network Traffic Control

The Network Traffic Control procedure presented in this section aims at optimising green timings and node offsets (see below for a formal definition) as decision variables.

A two-step optimisation is proposed, thus two separate optimisation levels are identified:

1. in the first level the single junction optimisation is carried out (green timings are decision variables)
2. in the second level the network coordination is performed (offsets are decision variables).

Single junction optimisation is carried out following the multi-criteria approach and the considered objective functions are the capacity factor (defined in **subsection 7.3.2**) and the total delay (further/other objective functions may be considered such as fuel consumption; air pollution etc.); the stage composition and sequence as described by the stage matrix are assumed known; the set of optimal solutions, identifying the points of the Pareto front, are found through Genetic Algorithms. Network optimisation is carried out following mono-criterion optimisation. In this case, once the green timings are known, the cycle length and the stage matrix and composition (derived from multi-criteria single junction optimisation), the node offsets are optimised considering the network total delay minimisation, and are computed by applying the traffic flow model described in **section 7.4**.

The proposed control strategy may operate off-line or on-line, in both cases no traffic flow prediction is carried out and the decision variables are optimised every control interval considering the most recently observed flows (see **Fig.7.1**).

In our application the off-line conditions are adopted and TRAFFMED acts as plant model and meanwhile it provides the flows for decision variables design. Furthermore, in **Fig.7.2**, the network traffic control procedure is shown in more details.

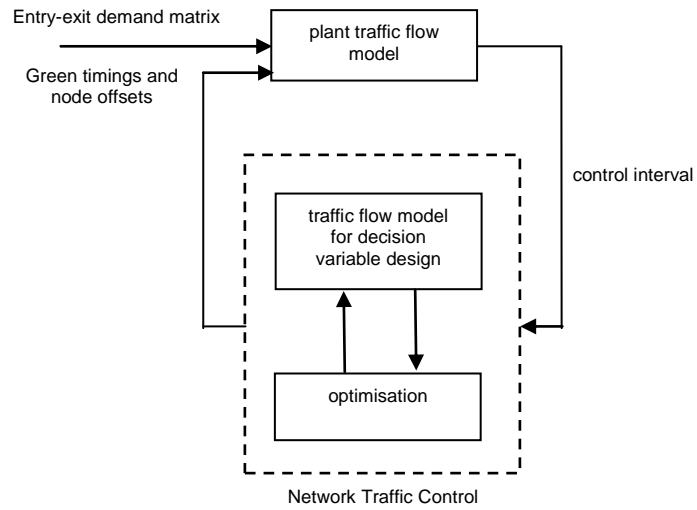


Fig.7.1 - Detailed description of the whole framework.

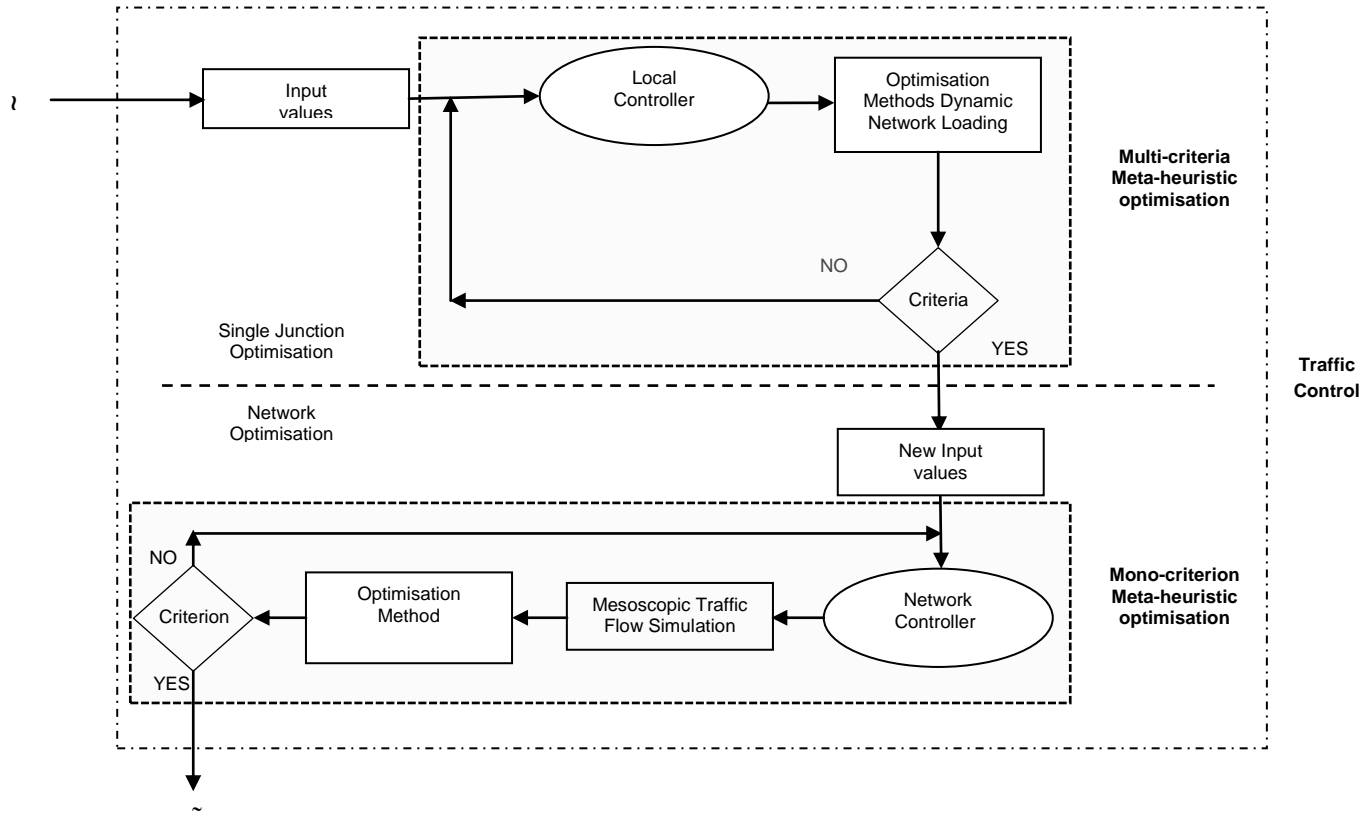


Fig.7.2 - Detailed description of the Network Traffic Control Procedure.

7.3.1 Variables and Constraints

Assuming that the stage composition and sequence are explicitly considered, let:

c be the cycle length, assumed known or as a decision variable (common to all junctions);

for each single junction (not explicitly indicated) let:

t_j be the duration of stage j as a decision variable;

Δ be the approach-stage incidence matrix, or stage matrix for short, with entries $\delta_{kj} = 1$ if approach k receives green during stage j and $= 0$ otherwise, assumed known;

t_{ar} be the so-called all red period at the end of each stage to allow the safe clearance of the junction, assumed known (and constant for simplicity's sake);

l_k be the lost time for approach k , assumed known;

$g_k = \sum_j \delta_{kj} \times t_j - t_{ar} - l_k$ be the effective green timings for approach k , needed to compute the total delay as described in equation 7;

y_k be the arrival flow for approach k , assumed known;

s_k be the saturation flow for approach k , assumed known;

Some constraints were introduced in order to guarantee:

stage durations being non-negative

$$t_j \geq 0 \quad \forall j$$

effective green timings being non-negative

$$g_k \geq 0 \quad \forall k$$

[this constraint is usually guaranteed by the non-negative stage duration, but for a too short cycle length with respect to the values of all-red period length and lost times, say the cycle length is less than $\sum_j \text{MAX}_k (\delta_{kj} l_k + t_{ar})$ this condition might not be met]

consistency among the stage durations and the cycle length

$$\sum_j t_j = c$$

the minimum value of the effective green timing

$$g_k \geq g_{min} \quad \forall k$$

A further constraint may be included in order to guarantee that the capacity factor is greater than 1 (or any other value assumed as a threshold)

$$0 \leq (s_k \times g_k) / (c \times q_k) - 1 \quad \forall k$$

Such a constraint may be added only after having checked that the maximum junction capacity factor for each approach k in the junction i is greater than 1, otherwise a solution may not exist whatever the objective function is.

For each junction i in the network let:

ϕ_i be the node offset, defined as the time shift between the start of the plan for the junction i and the start of the reference plan, say the plan of the junction number 1,

$$\phi_i = 0;$$

ϕ_{ij} be the link offset between each pair of adjacent junctions, $\phi_{ij} = -\phi_{ji}$ needed for computing network total delay. Let the junction network be represented by an undirected graph with a node for each junction and an edge for each pair of adjacent junctions (the actual traffic directions are irrelevant). According to this representation if such a network is loop less, all the m-1 link offsets are independent (as many as the independent node offsets) and may be used as decision variables; arterials are a special case of such kinds of networks; on the other hand, if the network contains k independent loops, the number of independent link offsets will be equal to m - k; in this case, it is better to use the m-1 independent node offsets as optimisation variables.

Finally let's assume

$$c \geq \phi_i \geq 0$$

7.3.2 Objective Functions

The objective functions adopted for the multi-criteria single junction optimisation were:

the junction capacity factor computed as

$$CF = \text{MIN}_k (s_k \times g_k) / (c \times y_k) \quad (\text{Eq.7.1})$$

where $(s_k \cdot g_k) / (c \cdot y_k)$ is the capacity factor for approach k.

the total delay, TD^S , computed through the two terms Webster's formula (Webster, 1958) as

$$TD = \sum_k y_k \times (0.45 \times c \times (1 - g_k/c)^2 / (1 - y_k/s_k) + y_k \times 0.45 / (s_k \times g_k/c) \times ((g_k/c) \times (s_k/y_k) - 1)) \quad (\text{Eq.7. 2})$$

It should be noted that the Webster formula does not take into account the offsets so it is only to be applied for single junction optimisation.

Further objective functions may be introduced such as air pollution, fuel consumption etc.

The Network optimisation was carried out by minimising the total delay, TD^N , computed through the traffic flow model described in the **section 7.4**.

7.3.3 Solution Algorithms

The optimisation framework presented in this paper (as described in the previous sections) introduces a multi-objective procedure to obtain green timings at a single junction (for efficiently finding optimal values of competitive objective functions that were the Capacity Factor and the Total Delay) and a Network optimisation to obtain minimum Network Total Delay. In both cases meta-heuristics were applied. As a matter of fact, such algorithms make it possible to effectively address optimisation problems (see Ozan et al., 2015; Ceylan and Ceylan, 2012) when conflicting objectives are identified, leading to multi-criteria optimisation (as it occurs in the proposed single junction optimisation) or when the objective function may not be expressed in a closed form, thus, derivatives are not easily available (as occurs in the Network optimisation problem).

A brief description of the adopted meta-heuristics, Genetic Algorithm (GA) at a single junction and Hill Climbing (HC) at network level, is shown hereunder.

MULTI-CRITERIA GENETIC ALGORITHM

For multi-criteria optimisation two objective functions are considered. In order to properly select the solution, the dominance criterion is applied and the Pareto front (which is the representation of the set composed by the vector of the objective functions) must be defined. In accordance with the literature, in this paper the GA is adopted as the solution algorithm which is a nature inspired algorithm and is based on the biological evolutionary process reproduced by an iterative procedure.

The solutions are identified by chromosomes each one composed by genes which represent, in our case, the stage durations; the chromosome corresponds to a vector of stages. Each solution is evaluated in terms of fitness described by a specific expression; on the basis of the fitness value, each chromosome could be selected to be parent in a reproductive cycle. In this cycle chromosomes might be modified by the application of two genetic operators, the crossover which generates a new solution by mixing genes of two chromosomes and the mutation which induces random variations of genes in a current solution (chromosome). Summing up, this meta-heuristic algorithm is based on an iterative procedure as summarised herein:

[1] the Population generation, in which the chromosomes are randomly generated; [2] the Fitness Evaluation (Baker,1985), which is strictly related to the probability of being selected to be a parent, in fact, it is computed by normalising the fitness function with respect to the fitness functions of all chromosomes;[3] the Selection, which is based on the fitness function and depends on the ranking evaluation; moreover, in the case of equal fitness, the solution is selected on the basis of the crowding distance computed as in the NSGA-II (Deb, 2002; Deb and Pratap, 2002), by considering the combination in the Euclidean distance of two criteria (such as the capacity factor and the total delay); the crossover [4] and mutation [5] operators, which might be applied for several times until the new population is obtained; the final step [6] is identified with the Stop criterion which is defined in terms of both maximum number of generations (and iterations) and non-improvement of the solution. A detailed description of the adopted algorithm herein briefly summarised may be founded in Cantarella et al. (2015).

HILL CLIMBING ALGORITHM

Hill Climbing is a neighbourhood-based meta-heuristic algorithm, without memory, deterministic in its basic version. The name originates from its ability to generate a succession of solutions by exploring the objective function surface which, if plotted, could be thought of as a series of hills and valleys in a multiple-dimensional space.

The algorithm was applied to optimise the node offsets for each controller and it is launched at the end of the multi-criteria GA procedure to get green timings at a single junction.

The algorithm first calculates the initial Network Delay (DN), and then begins an iterative process; Let n be the number of junctions and k be the number of independent loops in the network, on each iteration, the algorithm moves to junction $i = 1, \dots, n-k$ and tests out a series of $j = 1, \dots, h$ adjustments (as percentages of the cycle time C) on independent node offsets. The adjustments are stored in a vector $H = \{5, 15, 40, 15, 40, 15, 5, 1, 1\}$: other values should be assigned to the adjustment vector; the assumed values are those generally adopted. Hill climbing methods, such as the one herein described, do not guarantee finding the global minimum. To reduce the possibility of finding a poor local optimum, we use both large and small adjustments for the successive optimisation of each junction. The 15 per cent adjustments find an approximate local minimum of the objective function whilst the 40 per cent adjustments avoid becoming 'trapped' in that minimum. To ensure that H can contain a 1 second adjustment, the value of elements $H_{j=8}$ and $H_{j=9}$ are interpreted as 1 second and not as a percentage. At junction i and at iteration $j = 1$, the adjustment $H_j \cdot C$ is applied to the initial node offset ϕ_i so that the new adjusted node offset is $\phi_i(\text{adj}) = \phi_i + H_j \cdot C$. Such a new node offset is then tested to obtain the trial delay $D(\text{trial})$ on the link approaching the junction i . This is compared to the delay value $D(i)$ obtained considering the

initial node offset ϕ_i . If Delay has been reduced ($D(\text{trial}) < D(i)$), the new adjusted node offset $\phi_i(\text{adj})$ is saved as the best known and the search direction is kept for the next iteration; otherwise it is discarded (maintaining the previous node offset) and the search direction is inverted so that H_{j+1} becomes $-(H_{j+1})$. After the first junction has been ‘visited’ by all the adjustments $H_j=1\dots h$, the same process is applied for the second junction and so on until $i = n - k$. Residual dependent node offsets are then computed via an algebraic sum. **Fig.7.3** presents an illustration of how the HC algorithm works.

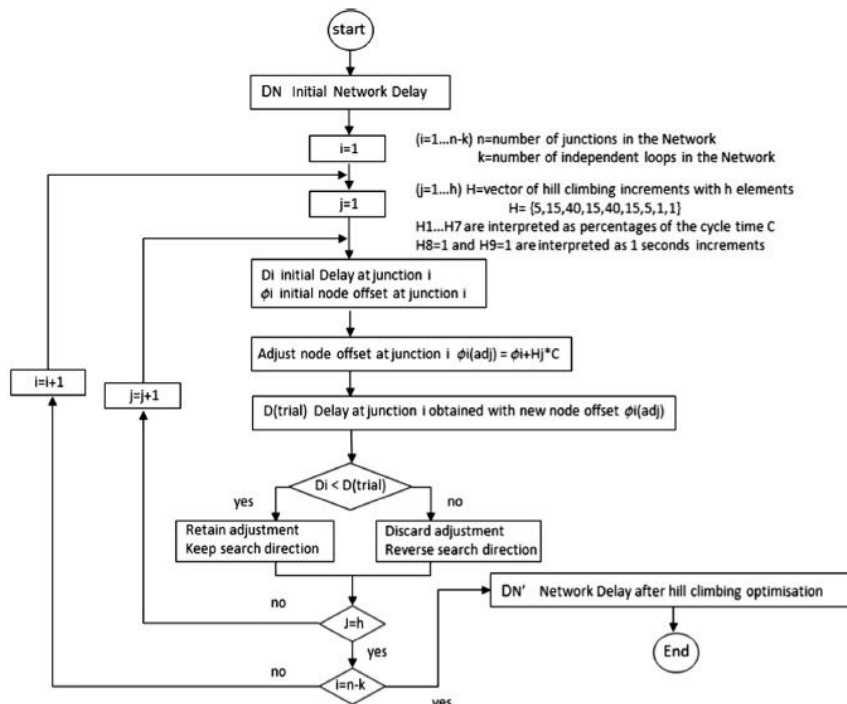


Fig.7.3 - General description of the Hill Climbing algorithm.

7.4 Traffic flow modelling

The adopted traffic flow model (as an enhancement of the model presented in Di Gangi, 2011) is discussed in this section. In particular, a detailed description of network representation, travel times and cost variables, flow representation and traffic dynamics is shown.

In the following, intervals will be denoted by integer indices, while time instants and lengths will be denoted by real values.

The considered mesoscopic model is based on a discrete packet approach, where the simulation is carried out over discrete *simulation time intervals*, t , further divided into *traffic dynamics sub-intervals*, k with length δ .

7.4.1 Packet generation

The generic packet P , is characterised by a departure time η within the *departure interval*, h (which is the simulation interval during which the departure occurs) and an origin/destination pair rs ; path choice is evaluated at the beginning of each departure interval, say at the origin and may be updated at the beginning of each simulation interval t , if en route rerouting occurs. For instance, the length of the simulation interval t and the length of the departure interval h , may be equal to 5 minutes and each simulation interval is further divided into sub-intervals with each one equalling 5 seconds.

The simulation is initialised considering a warm up period in order to guarantee the traffic congestion effects and avoid unrealistic free flow conditions.

A packet P moves in the network (see below Traffic dynamics for more details) and it is subject to queuing phenomena. The network is represented by a graph $G(N,A)$, with N the set of nodes and A the set of links.

Let:

- L_a be the length of link a ,
- x_a^S be the abscissa of a section S in link a that divides the link into two parts named respectively running part $[0, x_a^S)$ and queuing part $[x_a^S, L_a]$,
- $\zeta_a = L_a - x_a^S$ be the part of the link occupied by the queue.

The position of section S is obtained by the dynamic network loading (a detailed description is provided in **Fig.7.4**) and is updated at each time interval t considering the number of packets in the queuing part of the link at the previous interval $t-1$; the speed on the running part of the link at interval t depends on the outflow conditions of the previous interval $t-1$. This approach is slightly inconsistent since the level of service in the interval is considerably affected by flows and queues in the previous interval and not in the current. This assumption makes it possible to greatly simplify the computation while the inconsistency can be limited by reducing the length for the sub-interval, τ .

Outflow conditions on each link are considered homogeneous and constant for the entire duration of a sub-interval, δ . They are estimated at the beginning of each sub-interval and maintained for its entire duration, thus avoiding the occurrence of the internal fixed point problem. Summing up, the movement of a generic packet depends on its position on the link (running or queuing) and it can change its outflow conditions by moving forward from the running part to the queuing part of the link.

At each simulation time, the length of the queue on the link (and, consequently, the abscissa of section S) is updated and it is possible to take into account the eventual occurrence of queue spillback.

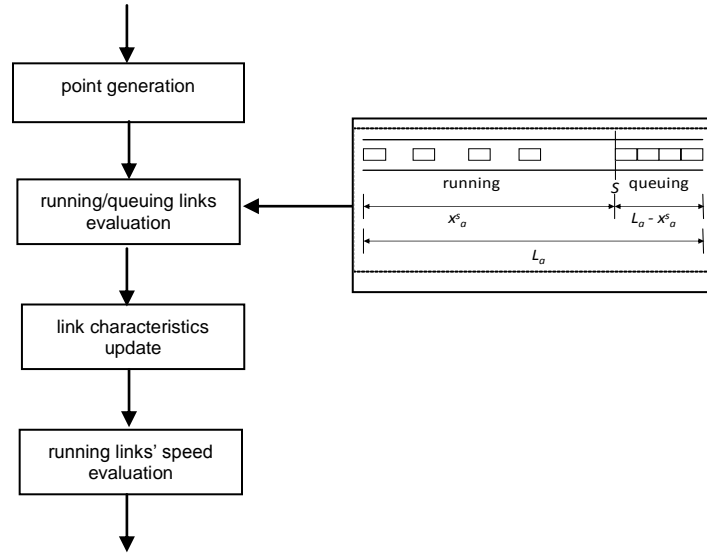


Fig.7.4- Detailed description of the Dynamic Network Loading.

7.4.2 Traffic dynamics

In this section both the movement rules of packets within each part of the link and the queue spillback simulation are described. These rules also make it possible to compute the travel time on the running part and on the queuing part.

Let $\Gamma_{rs\eta} = (N^s \subset N, A^s \subset A)$ be a sub-graph composed by the set of links that belong to the feasible paths connecting the O/D pair rs computed at time η , and let x be the abscissa on link a belonging to sub-graph $\Gamma_{rs\eta}$ representing the position, at time τ of interval t , of packet P left at the time η of the departure interval $h \leq t$ (if $h = t$ then $\eta < \tau$).

All variables of the traffic dynamics are declared below. In particular, let:

ρ_a^k be the density of the running part of the link a at beginning of sub-interval k given by the total amount of vehicles in each packet on link a during sub-interval k ;

v_a^k be the speed on the running part of the link a , updated at the beginning of each sub-interval k , as a function of density ρ_a^k ;

ρ_{crit}^a be the critical density on the link a , i.e. the density corresponding to the capacity in the fundamental diagram;

Q_a be the capacity at the final section of the link a ,

δ be the length of the generic sub-interval k and let $\tau \in [0, \delta]$.

With reference to the link model described above, the following cases may occur:

if $x < x_a^S$, the packet P is located on the running part of the link, it moves forward at a speed v_a^k ; then within the interval t , the packet P may reach a maximum abscissa x_a^S ; consequently:

if $(x_a^S - x) < (\delta - \tau) \cdot v_k^a$, the packet P enters the queuing part of the link a , before the end of the interval, at time τ' computed as in follows:

$$\tau' = \tau + ((x_a^S - x) / v_k^a)$$

otherwise, at the end of the interval it remains located on the running part;

if $x \geq x_a^S$, the packet P moves on the queuing part of the link a ; the length of the queuing part travelled by the packet P by the end of the interval is given by:

$$\Delta = ((\delta - \tau) \cdot Q_a) / \rho_{crit}^a$$

if $x + \Delta > L_a$, the packet P leaves the link a during the interval t at time:

$$\tau'' = \tau + ((L_a - x) \cdot \rho_{crit}^a) / L_a$$

otherwise it remains on the queuing part of the link a .

When packet P reaches the end of link a , it is necessary to identify the next link of its followed path. Let A^+ be the choice set of available links (made up by the links of $\Gamma_{rs\eta}$ whose initial node correspond to the final node of link a), a choice weight π_a^f is associated to each one of these links so that the link-choice problem can be expressed as: $a^+ \in \Gamma_{rs\eta} \mid \pi_a^f > \pi_{a_i}^f \forall a_i \in A^+, a_i \neq a^+$.

Since $\Gamma_{rs\eta}$ is a DAG (Directed Acyclic Graph), all links in the set A^+ , belong to at least one path connecting the rs pair, starting from r at time η , the arrival at destination of the packet is ensured.

As described above, for each O/D pair, rs , and departure time η a DAG sub-graph $\Gamma_{rs\eta} = (N^s \subset N, A^{rs} \subset A)$ of the network is associated to the packet P . Such a sub-graph is composed of the set of links that belong to the feasible paths connecting the O/D pair rs computed at time η . A choice weight π_a^f is associated to each link $a' \in A^{rs}$; the sub-graph $\Gamma_{rs\eta}$ is generated by implicit paths enumeration (i.e. Dial's STOCH; see Dial, 1971) and π_a^f is directly obtained (considering the total travel time on the link), thus the link-path incidence matrix is dynamically generated at the beginning of each interval t as well as the path flow patterns.

Before entering the next link a^+ it must be verified that i) link a^+ can accept the incoming packet and ii) the residual capacity of link a allows the packet P to move to the next link a^+ .

When the packet P reaches the end of the link a , before moving forward to the next link a^+ , it has to be verified that the length of the running part of the link a^+ is not null, that it is $x_{a^+}^S > 0$.

If $x_{a^+}^S > 0$, the packet P may enter the link a^+ otherwise it means that the entire length of link a^+ is occupied by a queue and the packet P remains on link a until the queue length on link a^+ is smaller than the length of the link a^+ , that is for the time until the condition $L_{a^+} - x_{a^+}^S < L_{a^+}$ is satisfied.

Once verified that the length of the running part of a link $a^+ \in A^+$ is not null, that is $x_{a^+}^S > 0$, it must be verified that the residual capacity of the link a allows the packet P to move to the next link a^+ .

Let:

- $n_e(P)$ be the number of elements of packet P , that is, $n_e(P) = l$,
- $U_a(\tau)$ be the number of packets which left the link until time τ ,
- Q_a be the capacity at the final section of the link a ,
- δ/T be the ratio between the length of interval k and the time unit T considered for the capacity.

if $[U_a(\tau) + n_e(P)] / \tau \leq Q_a \cdot \delta / T$ the packet P may leave the link a , otherwise the link a is saturated and the packet P remains on the link as long as a residual capacity, which allows the packet to leave the link, becomes available.

With reference to the introduced notation, it is possible to express the total delay (the total queuing time):

$$TD^N(\phi_j) = \sum_a (\zeta_a(\phi_j) \cdot \rho_{crit}^a) / Q_a \quad (Eq.7. 3)$$

7.4.3 Implementation remarks

In order to implement the above described method the following inputs must be defined:

- the study interval length;
- the simulation interval length;
- the traffic dynamics interval length;
- the number of vehicles in each packet P assumed one vehicle at most;
- the demand flows over the simulation interval;
- the path choice model applied at beginning of each choice model interval ;
- the departure time choice model, applied after the path flows have been defined; in a simple approach departures are uniformly distributed over time with constant headway given by the ratio between the length of simulation and the flow;
- the speed-density function to compute the congested speed, as explained in more detail in Appendix A.

7.5 Application

In this section the case study and the comparison between the results obtained through the proposed method and the benchmark tool are discussed. In particular, the results of two applications are shown, simple network (A), also considered for validation, and a more complex network (B). The proposed method was implemented in a code developed in Python (according to the JetBrains PyCharm Community Edition 3.0.2 framework was adopted) and run on a server machine which has an Intel(R) Xeon(R) CPU E5-1603, clocked at 2.8GHz and with 4GB of RAM.

7.5.1 The case study (a)

In this section an empirical validation test of NTC-TRAFFMED is discussed. The proposed approach is shown in **Fig.7.5**; in particular, two steps were considered:

1. the first step aims at validating the network traffic control procedure;
2. the second step aims at validating the traffic flow model.

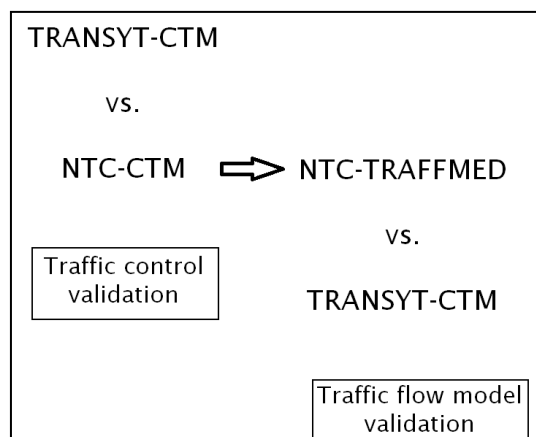


Fig.7.5- Overview of a two-step validation test.

In the first step, the results carried out by NTC (Network Traffic Control procedure corresponding to the same shown in **section 7.3**) considering the Cell Transmission Model (CTM) were compared with those carried out by a benchmark tool TRANSYT[®]TRL still considering the same traffic flow model; in this step the network traffic control procedure (published in Cantarella et al., 2015) is compared with the optimisation procedure in TRANSYT[®]TRL. Moreover, the same set of input conditions (path choice at each choice model interval, free flow speed, saturation flow, the demand flows over the simulation interval) is adopted in order

to guarantee the consistency of the comparison; the adopted values of input parameters are described in the following section.

In the second step, the NTC-TRAFFMED is compared with the NTC-CTM used at the previous step and then the procedures are characterised by the same network traffic control procedure but by two different traffic flow models.

The considered network was composed of a bi-directional triangular network connecting 3 zones, as shown in **Fig.7.6**. Traffic signal 1, which was located at node 1, was set to be the master signal relative to which the offset of traffic signal 2 (at node 2) and 3 (at node 3) was referenced. The demand entry-exit flows were summarised in **Table 7.1**.

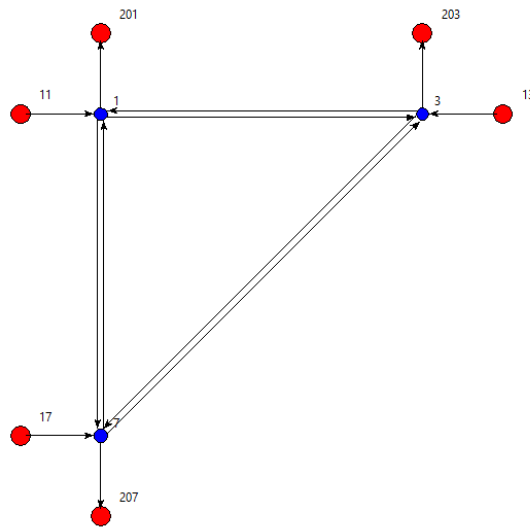


Fig.7.6- Network used for calibration/validation.

Table 7.1 – Entry-exit matrix.

Entry [veh/h]	Exit [PCU/h]		
	201	203	207
11	#	73	45
13	28	#	50
17	65	45	#

A cross comparison among the three optimisation strategies is shown in the following tables (see **Table 7.2** and **Table 7.3**) in terms of performance indicators (TD, total delay and DOS, degree of saturation) and in terms of node offsets.

Table 7.2 - Performance indicators of NTC-TRAFFMED, NTC-CTM and TRANSYT-CTM.

Performance indicator	NTC-TRAFFMED	NTC-CTM	TRANSYT-CTM
TD[PCU-hr/hr]	9.58	10.83	10.61
DOS[%]	37.00	44.00	46.00

The obtained results showed that NTC CTM mimics TRANSYT CTM well while it slightly differs from NTC-TRAFFMED; in particular, in terms of total delays, similar values are obtained from NTC-CTM and TRANSYT-CTM and a slightly different value is obtained from NTC-TRAFFMED. De facto, the total delays estimation, as well known, depends on the adopted traffic flow model; in terms of DOS, as in previous cases, similar values are obtained from NTC-CTM and TRANSYT-CTM and an outperforming value is obtained from NTC-TRAFFMED. Finally, as expected, different results are obtained in terms of node offsets.

Table 7.3 - Node offsets of three traffic flow models.

Node offset	NTC-TRAFFMED	NTC-CTM	TRANSYT-CTM
$\Phi 2-1$ [s]	79	57	52
$\Phi 3-1$ [s]	60	28	28

In conclusion, the above results show that the effectiveness of the proposed optimisation method in NTC is comparable with the one in TRANSYT, see comparison between NTC-CTM and TRANSYT-CTM; moreover, that Network Traffic Control with mesoscopic traffic modelling may produce better results, as

shown by the DOS indicator that is independent of the method (unlike the total delay).

A further validation of the optimisation procedure was carried out by considering TRAFFMED for decision variable design and CTM as the plant model (NTC_{TRAFFMED}-CTM). Results displayed in **Table 7.4** show the effectiveness of the proposed mesoscopic model for decision variable design within the optimisation procedure, in fact, NTC-TRAFFMED outperforms NTC_{TRAFFMED}-CTM.

Table 7.4 - Performance indicators of NTC-TRAFFMED and NTC_{TRAFFMED}-CTM.

Performance indicator	NTC-TRAFFMED	NTC _{TRAFFMED} -CTM
TD[PCU-hr/hr]	9.58	9.81
DOS[%]	37.00	40.00

7.5.2 The case study (B)

In order to further test the proposed optimisation strategy NTC-TRAFFMED, a grid network (see **Fig.7.7**) was considered, made up of 4 origins, 4 destinations (totalling 17 nodes including connectors) o-d pairs, 9 nodes and 12 bidirectional links (totalling 32 links including connectors), each one with one lane for each direction; the saturation flow of each lane is assumed equal to 1800 PCU/h. In terms of the links' length, links connecting node 5 with other nodes (2-5, 5-6; 5-8) are equal to 400 m, other links on the network are equal to 800 m. The speed could be obtained by any speed density function (fundamental diagram – stable regime) such as the link performance BPR function, the conical function (see Spiess, 1989) etc. In this paper the BPR-like function $v(f) = v_0 / (1 + a (f/Q)b)$ was adopted by assuming suitable values of parameters $a = 2.0$ and $b = 3.5$ (see more details in Appendix A).

The o-d pair demand flows are shown in the following **Table 7.5**. The path choice behaviour was modelled considering three paths for each o-d pair (a selective approach to path choice specification) varying over the simulation intervals depending on network conditions, and a logit choice model. The perceived utility of each path from origin o to any destination d is assumed distributed with variance σ_o^2 equal to 20% of the average of the minimum path costs across all destinations. Thus, the dispersion parameter for origin o is given by $\theta_o = (\sqrt{6/\pi}) \cdot \sigma_o$.

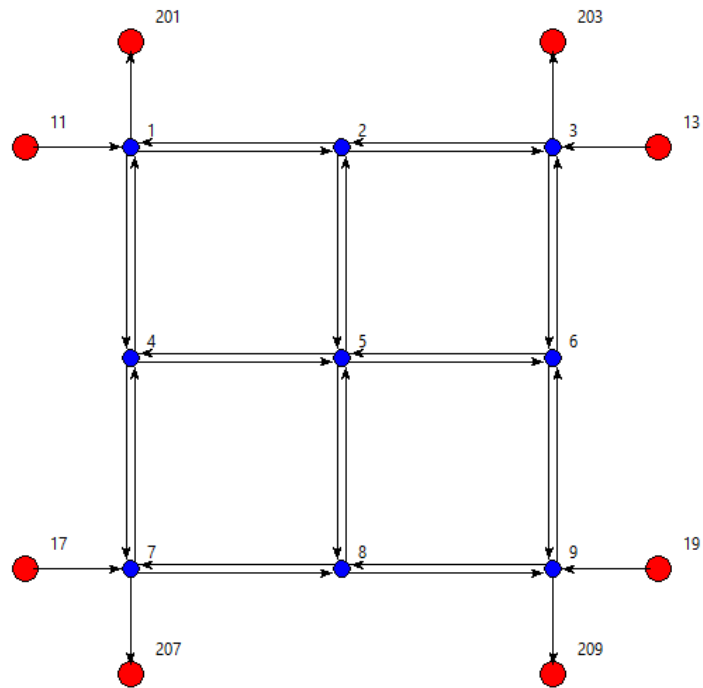


Fig.7.7 - Test network with 9 signalised junctions.

Table 7.5 - entry-exit matrix.

Entry [veh/h]	Exit [PCU/h]			
	201	203	207	209
11	#	480	384	336
13	432	#	288	384
17	480	624	#	423
19	336	576	432	#

In our application the total simulation time was 60 minutes and was divided into 12 departure intervals, h , each of them equal to 5 minutes and was coincident with the simulation interval, t ; during t , the path choice remains constant and packets

departures are uniformly distributed; each simulation interval t was further divided into sub-intervals with length δ equal to 5 seconds¹.

The network was preloaded at 60% of the links' capacities (6480 PCU in all) in order to properly compute the congested speed from the beginning of the simulation.

An example of the path flow patterns (at time interval 3), is shown in **Table 7.6**.

In the same table the (adjusted) flows and the discharge headway according to each simulation interval are also shown. It is worth noting that in most cases only one path has a positive flow but this path may change over the simulation intervals.

The traffic signals were updated every 15 minutes, which we refer to as the control interval, by implementing the optimisation procedure, on the basis of the flows measured at the previous time interval.

In particular, the traffic flow data are obtained by using the same mesoscopic model as previously described and are used as inputs of the Network Traffic Control procedure in order to get consistent values of the decision variables (green timings at local level and offsets at network level). Therefore, let the length of the simulation interval be equal to 5 minutes, the Dynamic network loading is run for n successive times until $n \cdot 5$ equals the length of the control interval (15 minutes).

¹The computer time needed for 1 hr of simulation depends on the length of the sub-interval: if the length of sub-interval is equal to 5 seconds, the simulation takes 12 seconds; if the length of sub-interval is equal to 10 seconds, the simulation takes 16 seconds; if the length of sub-interval is equal to 15 seconds, the simulation takes 22 seconds.

Table 7.6 - Path flow patterns at time interval 3.

O	D	path	path nodes	flow rates	flow [veh/300s]	headway [s/veh]
11	209	1	{11 1 4 5 6 9 209}	0.73	21	14.3
11	209	2	{11 1 4 5 8 9 209}	0.27	7	42.9
11	209	3	{11 1 4 5 8 5 6}	0.00	0	0.0
11	203	1	{11 1 4 5 6 3 203}	1.00	40	7.5
11	203	2	{11 1 2 3 203 - -}	0.00	0	0.0
11	203	3	{11 1 4 5 8 5 6}	0.00	0	0.0
11	207	1	{11 1 4 7 207 - -}	0.98	32	9.4
11	207	2	{11 1 4 5 8 7 207}	0.02	0	0.0
11	207	3	{11 1 4 5 4 7 207}	0.00	0	0.0
17	201	1	{17 7 4 1 201 - -}	0.99	39	7.7
17	201	2	{17 7 4 5 2 1 201}	0.01	0	0.0
17	201	3	{17 7 8 5 2 1 201}	0.00	0	0.0
17	203	1	{17 7 4 5 6 3 203}	0.99	51	5.9
17	203	2	{17 7 8 5 6 3 203}	0.01	0	0.0
17	203	3	{17 7 4 5 8 5 6}	0.00	0	0.0
17	209	1	{17 7 8 9 209 - -}	0.97	34	8.8
17	209	2	{17 7 4 5 6 9 209}	0.03	0	0.0
17	209	3	{17 7 4 5 8 9 209}	0.01	0	0.0
19	201	1	{19 9 8 5 2 1 201}	0.99	27	11.1
19	201	2	{19 9 8 5 4 1 201}	0.01	0	0.0
19	201	3	{19 9 6 3 2 1 201}	0.00	0	0.0
19	203	1	{19 9 6 3 203 - -}	1.00	47	6.4
19	203	2	{19 9 8 5 6 3 203}	0.00	0	0.0
19	203	3	{19 9 6 3 6 3 203}	0.00	0	0.0
19	207	1	{19 9 8 7 207 - -}	1.00	35	8.6
19	207	2	{19 9 8 5 8 7 207}	0.00	0	0.0
19	207	3	{19 9 8 5 4 7 207}	0.00	0	0.0
13	201	1	{13 3 2 1 201 - -}	1.00	35	8.6
13	201	2	{13 3 6 3 2 1 201}	0.00	0	0.0
13	201	3	{13 3 2 5 2 1 201}	0.00	0	0.0
13	209	1	{13 3 6 9 209 - -}	1.00	31	9.7
13	209	2	{13 3 6 3 6 9 209}	0.00	0	0.0
13	209	3	{13 3 2 5 6 9 209}	0.00	0	0.0
13	207	1	{13 3 2 5 8 7 207}	0.65	15	20.0
13	207	2	{13 3 6 5 8 7 207}	0.29	6	50.0
13	207	3	{13 3 6 9 8 7 207}	0.06	1	00.0

7.5.3 Analyses

In this sub-section the effect of the proposed procedure was evaluated by considering some indicators such as the Mean Maximum Queue (MMQ) which is the mean over the simulation intervals of the position of the back of the queue at its peak during the cycle, measured in PCU back from the stop line, the Total Delay (i.e. TD) and the Capacity Factor (i.e. CF). As mentioned before, the optimisation was computed every 15 minutes, until the end of the simulation time, thus, results were compared by considering three successive intervals; the first time interval was not considered for any comparison due to the well-known effect of the warm up simulation. The results were evaluated firstly at a disaggregated level over time (at each time interval) and an aggregated level for the entire network and secondly at a disaggregated level over time (as mentioned previously, at each time interval) and a disaggregated level for individual regions of the network.

With respect to the network results shown in **Table 7.7**, it should be observed that higher values of MMQ, TDs and CFs are obtained at time interval 3, however, at time interval 4 the network may be considered as under-saturated.

Table 7.7 - Results over the time intervals achieved by NTC-TRAFFMED.

NTC-TRAFFMED			
Interval	MMQ [PCU]	TD ^N [PCU-h/h]	CF
1	116.58	60.44	1.25
2	109.50	58.82	<1*
3	164.61	106.27	<1*
4	106.33	36.26	1.58

A further analysis was carried out by considering the results which were disaggregated for every region within the network; in particular, a clustering of links was identified by distinguishing internal, border and external regions as shown in **Fig.7.8** (i.e. links are clustered in external, internal and on the border between the internal and the external region). Their corresponding Queue Lengths (measured in meters), TDs and CFs, computed respectively as the average, the maximum and the minimum over links composing each region, are shown in the following **Table 7.8**.

As in the previous case, we exclude the results obtained for the first time interval.

Table 7.8 - Results over the time intervals of simulation for the external, internal and border regions, achieved by NTC-TRAFFMED.

NTC-TRAFFMED									
Interval	Queue Length [m]			TD ^N [PCU-h/h]			CF		
	EXT	BOR	INT	EXT	BOR	INT	EXT	BOR	INT
1	78.82	17.74	12.53	18.10	3.32	1.72	1.96	1.25	2.13
2	65.87	21.11	6.97	25.73	4.31	0.87	0.52	1.12	4.55
3	56.04	27.93	39.57	16.36	28.7	32.74	0.83	0.67	0.72
4	34.94	19.95	22.39	2.85	5.62	4.25	1.59	1.69	1.75

Table 7.8 shows that congestion is equally distributed over three regions confirming procedure suitability in queuing phenomena management.

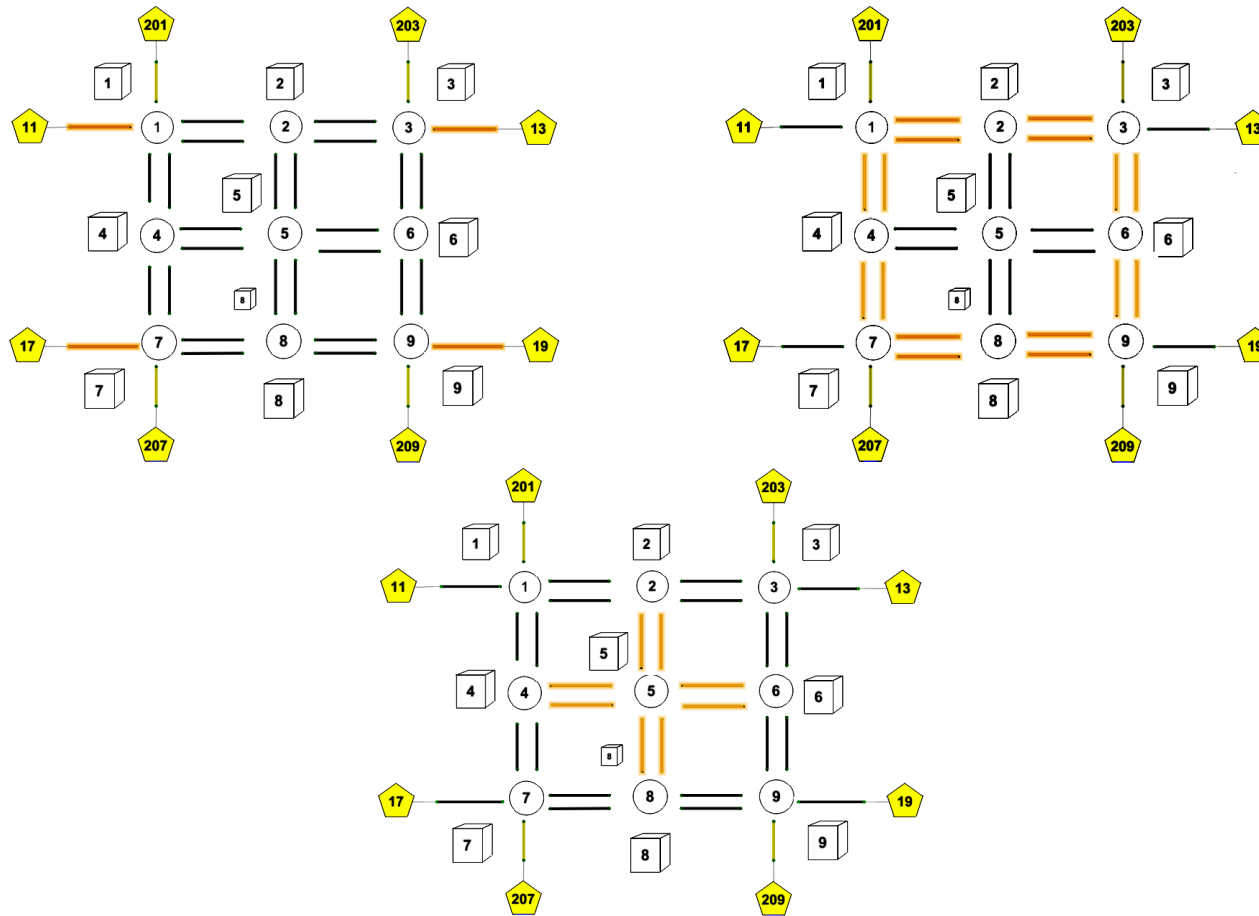


Fig.7.8 - Links composing the external, border, internal region.

7.5.4 Comparisons: NTC-TRAFFMED vs. TRANSYT-CTM

In this sub-section the effect of the procedure was evaluated by considering the comparisons of the results (a more detailed description of considered indicators will be given below) collected from our simulation environment with ones carried out by using TRANSYT[®]TRL. Furthermore, the considered flow rates in TRANSYT[®]TRL were consistent with the link-path incidence matrix generated in NTC-TRAFFMED.

As previously described, in the optimisation procedure the green timings and the (node) offsets were optimised in two separate steps, and the stage matrix at each single junction was given.

The obtained results were evaluated by considering, as in previous sections, the indicators such as the Mean Maximum Queue (MMQ), the Total Delay (i.e. TD) and the Capacity Factor (i.e. CF). The traffic flow model adopted in TRANSYT[®]TRL in the case of optimisation and simulation is the Cell Transmission Model (CTM, Daganzo, 1994). The input parameters for each link that can be set in the TRANSYT-CTM are: i) the cruise speeds (assumed 30kph), whence the cell length, the number of cells and the maximum occupancy is obtained; the maximum flow (or the cell saturation flow, evaluated by taking account of link characteristics), whence the maximum number of vehicles that can flow into cell *i* from time *t* to *t*+1 is obtained. NTC-CTM was not applied for brevity's sake.

The comparisons were made firstly on results which were disaggregated over time (at each time interval) and aggregated for the entire network and secondly on results which were disaggregated over time (as mentioned previously, at each time interval) and disaggregated for individual regions of the network.

In **Table 7.9** the network results are shown. The first column refers to the time intervals of the simulation, as described above. The MMQ, the TDs and the CFs reported in the remaining columns represent the aggregate network values.

Table 7.9 - NTC-TRAFFMED vs. TRANSYT-CTM.

Interval	NTC-TRAFFMED			TRANSYT-CTM		
	MMQ [PCU]	TD ^N [PCU-h/h]	CF	MMQ [PCU]	TD ^N [PCU-h/h]	CF
1	116.58	60.44	1.25	85.04	26.12	1.35
2	109.50	58.82	<1*	126.86	99.52	<1*
3	164.61	106.27	<1*	118.33	97.05	<1*
4	106.33	36.26	1.58	344.76	319.2	<1*

As aforementioned, we will not compare results carried out at the first time interval of simulation. However, further considerations can be made on numerical results of successive time intervals.

In particular, it can be observed that:

- i. at the second interval of simulation, NTC-TRAFFMED outperforms TRANSYT-CTM with respect to the TD (i.e. 58.82 PCU-h/h against 99.52 PCU-h/h);
- ii. at the third interval of simulation the results reached by both approaches may be considered comparable (i.e. 106.27 PCU-h/h in NTC-TRAFFMED and 97.05 PCU-h/h in TRANSYT-CTM);
- iii. even though at the second and the third interval of simulation, the oversaturation conditions are reached in both of the cases, at the last time interval NTC-TRAFFMED outperforms TRANSYT-CTM not only with respect to the TD (i.e. $TD_{NTC-TRAFFMED}=36.26$ [PCU-h/h] and $TD_{TRANSYT-CTM}=319.2$ [PCU-h/h]) but also with respect to the CF (i.e. $CF_{NTC-TRAFFMED}=1.58$ and $CF_{TRANSYT-CTM}<1$).

With reference to the results of the 4th interval for NTC-TRAFFMED and TRANSYT-CTM in **Table 7.10**, they can be observed, both in terms of MMQ and of TD, and meaningful differences can be pointed out (The TD is reduced by a factor of 10). Such a result is due to the different traffic flow model adopted in the optimisation framework. As a matter of fact, the CTM in its basic version allows vehicles to be held at the upstream cells rather than advancing forward to the available-capacity downstream cells, which leads to the holding-back phenomenon (see Doan and Ukkusuri, 2012). Unlike CTM, our proposed TRAFFMED model allows for over the time intervals of the simulation to distribute the queue over the network links, leading to a satisfactory aggregate network Total Delay. A further analysis was carried out by considering the results which were disaggregated for every region within the network as described in the previous **Fig.7.8**. Their corresponding Queue Lengths (measured in meters), TDs and CFs, computed respectively as the average, the maximum and the minimum over links composing each region, are shown in the following **Table 7.10**. As in the previous case, we exclude any consideration of the results carried out at the first time interval.

Table 7.10 - Results over the time intervals of simulation for the external, internal and border regions, achieved by NTC-TRAFFMED and TRANSYT-CTM.

NTC-TRAFFMED									
Interval	Queue Length [m]			TD ^N [PCU-h/h]			CF		
	EXT	BOR	INT	EXT	BOR	INT	EXT	BOR	INT
1	78.82	17.74	12.53	18.10	3.32	1.72	1.96	1.25	2.13
2	65.87	21.11	6.97	25.73	4.31	0.87	0.52	1.12	4.55
3	56.04	27.93	39.57	16.36	28.7	32.74	0.83	0.67	0.72
4	34.94	19.95	22.39	2.85	5.62	4.25	1.59	1.69	1.75

TRANSYT-CTM									
Interval	Queue Length [m]			TD ^N [PCU-h/h]			CF		
	EXT	BOR	INT	EXT	BOR	INT	EXT	BOR	INT
1	32.43	19.54	8.49	2.24	1.96	1.06	2.04	1.37	2.94
2	153.95	6.63	4.92	31.78	1.75	0.72	0.42	2.13	7.14
3	153.59	4.14	3.67	31.78	1.27	0.68	0.52	1.12	4.55
4	245.99	5.57	4.23	115.74	0.73	1.05	0.42	2.14	2.94

Table 7.10 shows that it may be appropriate to compare the results obtained by NTC-TRAFFMED and TRANSYT-CTM for the three network regions at progressive time intervals of simulation. In fact, numerical results ensure that indicators of external regions in TRANSYT-CTM are usually dominated by ones collected in NTC-TRAFFMED and vice versa that indicators of internal and border regions in TRANSYT-CTM usually dominate those reached in NTC-TRAFFMED. However, if the averages of TDs and CFs are considered, a relative increase of the CF in TRANSYT-CTM, equal to 37% and a relative decrease of the TD, in NTC-TRAFFMED, equal to 50% is observed (presumably induced by the different optimisation strategy). In order to evaluate the effectiveness of the multi-criteria optimisation, a further analysis was carried out by comparing graphs (see **Fig.7.9**) which represent the values of CFs optimised in NTC-TRAFFMED and TRANSYT-CTM. As in accordance with previous numerical considerations, it should be stated that TRANSYT-CTM induces higher values of CF over border and internal links unlike external links where higher values are obtained in NTC-TRAFFMED.

A final in depth investigation was carried out in terms of queues on the network. In particular, for each strategy, the Queues distribution at each time interval and for each cluster of links was obtained (see **Fig.7.10**).

It appears noteworthy that the queues observed in TRANSYT-CTM in the internal and border links are relatively lower compared with ones experienced in the external links. In contrast, a homogeneous outcome (for each region and over the time intervals of simulation) of such an indicator may be revealed as regards the proposed strategy.

Indeed, the main advantage of NTC-TRAFFMED lies in a higher efficiency in managing the dynamics of overflow queues, unlike TRANSYT-CTM in which the

higher values of queues are concentrated over all the time intervals of simulation on external areas.

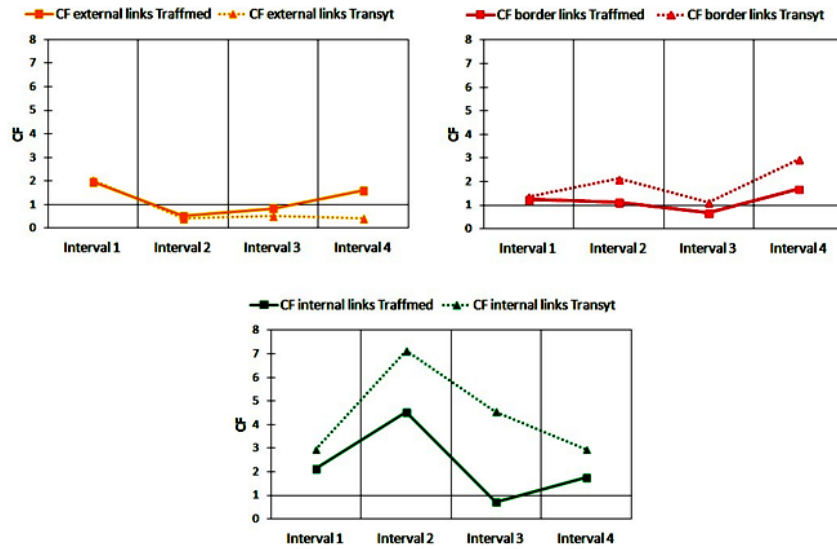


Fig.7.9 - CFs over the time intervals of simulation for the external, internal and border regions, achieved by NTC-TRAFFMED and TRANSYT-CTM.

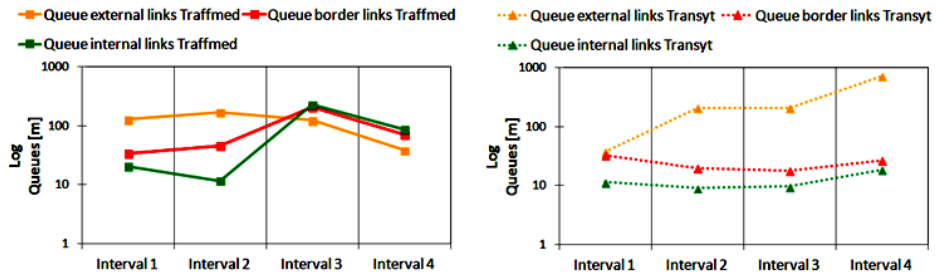


Fig.7.10 - Queues distribution over the time intervals of simulation for the external, internal and border regions, achieved by NTC-TRAFFMED (on the left) and TRANSYT-CTM (on the right).

7.6 Conclusions

The paper aims at developing a flow based Network Traffic Control method in which the optimisation is carried out through a two-step approach where the first step refers to each single junction optimisation (green timings), the second to network coordination (offsets).

The control strategy is affected by the optimisation procedure itself, as well as by the adopted traffic flow model. Due to this consideration, another focus of the paper lies in traffic flow modelling. In this research, an advanced mesoscopic approach with horizontal queuing is proposed (TRAFFMED- Traffic Analysis and Flow Forecasting Mesoscopic Dynamic).

TRAFFMED is based on the representation of each link in a running and a queuing part whose length may change over time due the changes of the queue length.

Metaheuristic solution algorithms are applied since they seem to be more suitable to solve the optimisation problems for the single junction, where a multi-criteria optimisation is carried out (Cantarella et al., 2014), and for the network, where the objective function (i.e. total delay) is affected by traffic flow simulation and not expressed in a closed form.

Two cases of study are considered: a simple three nodes network and a more complex grid network composed by 9 nodes; the first one was adopted for calibrating and validating the NTC-TRAFFMED, the second for the further analysis of the model performances. For this purpose, in the second application case, the obtained results (NTC-TRAFFMED) are compared with those carried out from a benchmark tool (TRANSYT-CTM), with respect to simulated flow data. The most significant insight is observed in the case of links clustered in three network regions (external, border, internal); unlike the benchmark tool, in which queuing phenomena are significantly concentrated only on the external area, in the proposed approach queuing phenomena may be considered better spread over the network.

Percentage-based horizontal stacked bar plots (the red bars refer to the queuing part, the light blue bars refer to the running part) are shown in **Fig.7.12** and **Fig.7.12**, for the simulation intervals 3 and 4. In order to compare the control strategies mentioned above (a schematic representation of the running and the queuing part of the signalised link is illustrated at the top of **Fig.7.12** and **Fig.7.12**). The representation of the links against the queue percentage aims to highlight the relevant difference between NTC TRAFFMED, in which from interval 3 to interval 4 queue distributions significantly change, and TRANSYT, in which from interval 3 to interval 4 queue distributions slightly change.



NTC-TRAFFMED

Interval 3

TRANSYT-CTM

■ queueing arc ■ running arc

■ queueing arc ■ running arc

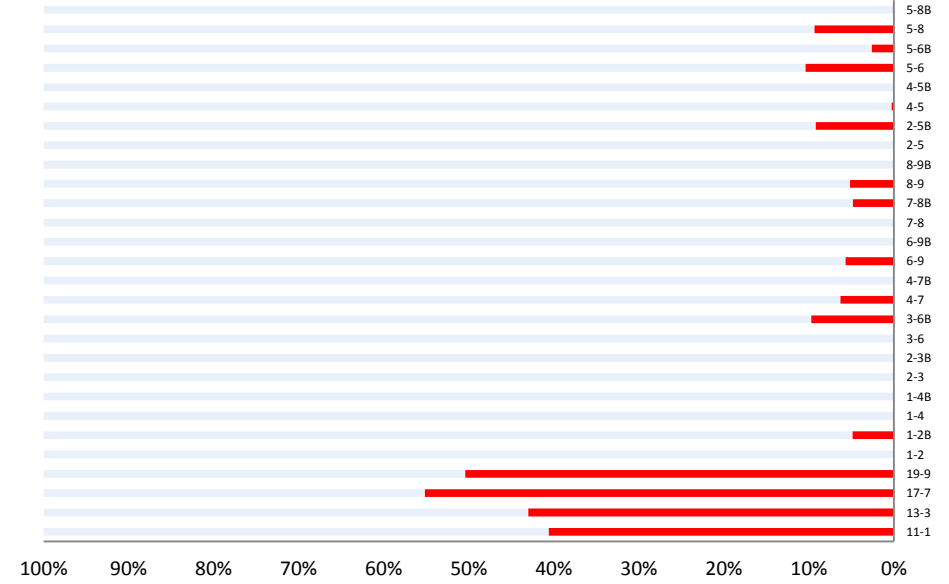
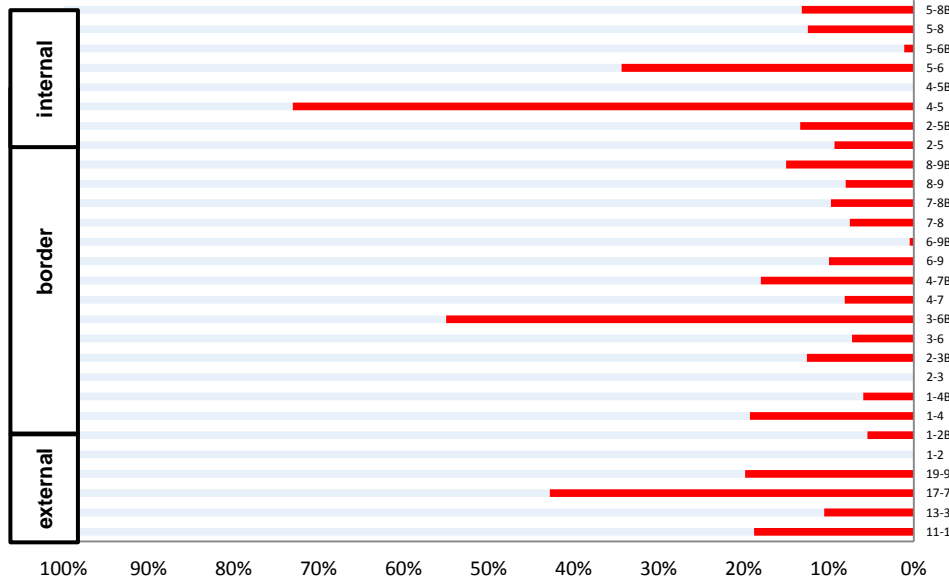


Fig.7.11 - Queues dynamic evolution over the time intervals (3-4) of simulation



NTC-TRAFFMED

Interval 4

TRANSYT-CTM

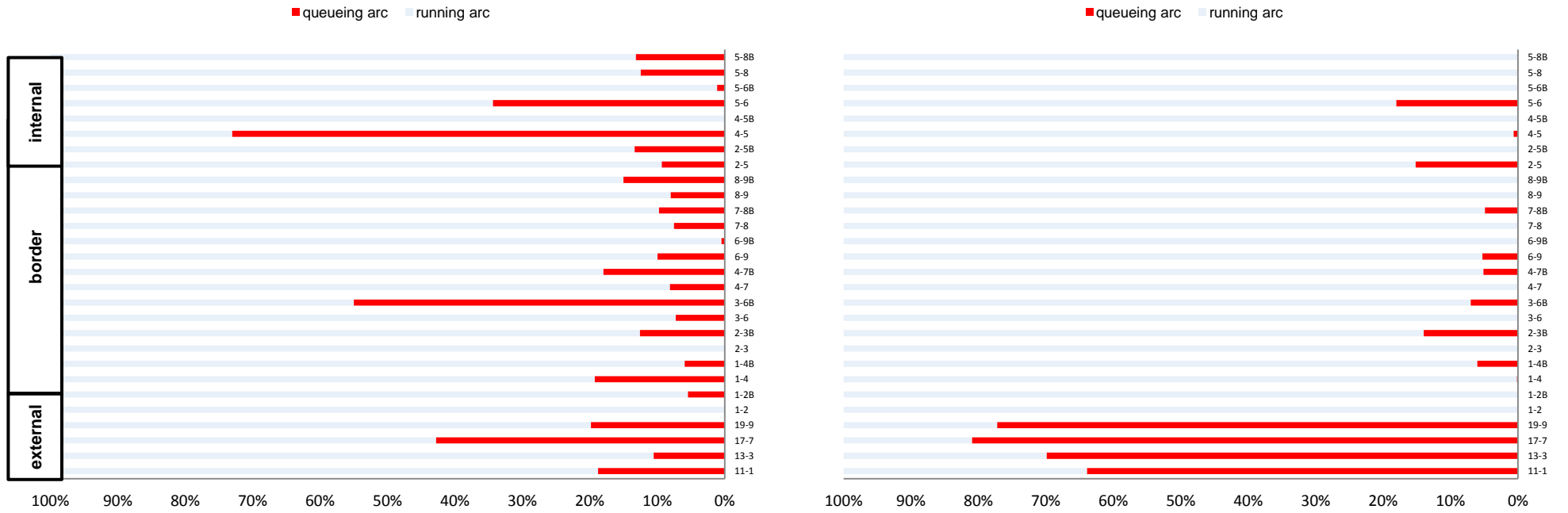


Fig.7.12 - Queues dynamic evolution over the time intervals (3-4) of simulation

In future works, several issues will be addressed as far as the traffic control method is concerned:

- i. a network traffic control procedure where green timing, green scheduling and offsets are simultaneously optimised (scheduled synchronisation);
- ii. a thorough investigation of the platoon dispersion representation in the proposed model;
- iii. the integration of the proposed approach for network traffic control with a flow forecasting model;
- iv. the extension of the proposed mesoscopic NTC TRAFFMED model to store and a forward strategy (Gazis and Potts, 1963).

As regards the traffic modelling the calibration and validation of the proposed mesoscopic model will be carried out considering actual data. Furthermore, the mesoscopic traffic flow modelling herein adopted will be compared with other ones eventually with different levels of variables aggregation.

It will be worth considering also the proposed NTC strategy within the Urban Network Design Problem, including lane allocation optimisation as well as user path choice behaviour modelling.

Acknowledgments

Authors wish to thank three anonymous reviewers for their helpful comments. This research has been partially supported by the University of Salerno, Italy, EU under local grant n. ORSA132945 financial year 2013, Campania Region PON- Ricerca e Competitività 2000-2013 grant n. 04a2_G “SMART TUNNEL”, and PhD program on transportation (Ph.D. School in Environmental Engineering).

An earlier version of this research is contained in the preprints of Dynamic Traffic Assignment Conference 2014 and 11th International Conference on Advances in Civil Engineering (ACE2014).

Authors wish to thank to Nikolas Geroliminis, met at the University of Naples ‘Federico II’ for his helpful comments.

For friendship policy, the authors order is organised in alphabetical order with respect to the affiliations, then in alphabetical order of the authors name within each affiliation.

References

- Astarita, V. (1996). A Continuous Time Link Model for Dynamic Network Loading Based on Travel Time Function. 13th International Symposium on Theory of Traffic Flow, pp. 79–103.
- Adamo, V., Astarita, V., Florian, M., Mahut, M., & Wu, J. H. (1999). Modelling the spill-back of congestion in link based dynamic network loading models: a simulation model with application. In 14th International Symposium on Transportation and Traffic Theory.

- Baker, J.E.,(1985).“Adaptive selection methods for genetic algorithms”, Proceedings of the 3rd International Conference on Genetic Algorithms and Applications, In: Grefenstette, J.J. (ed.), New Jersey, Lawrence Erlbaum: Hillsdale, pp. 100-111.
- Ben-Akiva, M.E., Bierlaire, M., Koutsopoulos, H.N., and Mishalani, R., G., (1996). DynaMIT: Dynamic network assignment for the management of information to travelers. In Proceedings of the 4th Meeting of the EURO Working Group on Transportation, Newcastle.
- Binning, J.C., Crabtree, M.R., & Burtenshaw, G.L., (2010). Transyt 14 user guide. Transport Road Laboratory Report nr AG48. APPLICATION GUIDE 65 (Issue F).
- Bliemer B.C.J., (2006). Dynamic queuing and spillback in an analytical multiclass dynamic traffic assignment model. Proceedings of the 1st International Symposium on Dynamic Traffic Assignment, 21-23 June 2006, The University of Leeds, UK.
- Bretherton R.D., Wood K., and Bowen, G.T., (1998). “SCOOT Version 4.” Proceedings IEE 9th International Conference on Traffic monitoring and control. London.
- Burghout, W., (2004). Hybrid Microscopic-Mesosopic Traffic Simulation. Doctoral Dissertation, Royal Institute of Technology.
- Burghout, W., Koutsopoulos,H.N., and Andréasson,I., (2005). Hybrid mesoscopic-microscopic traffic simulation. Transportation Research Record 1934, pp. 218-225.
- Burghout, W. and Wahlstedt,J., (2007). Hybrid traffic simulation with adaptive signal control, Transportation Research Record 1999, pp. 191-197.
- Bureau of Public Roads (1964). Traffic assignment Manual. US Dept. of Commerce, Urban Planning Division, Washington, DC.
- Cantarella, G. E., de Luca, S., Di Pace, R., & Memoli, S., (2015). Network Signal Setting Design: Meta-heuristic optimisation methods. Transportation Research Part C: Emerging Technologies.
- Cantarella, G. E., de Luca, S., Di Pace, R., & Memoli, S. (2014). Signal Setting Design at a Single Junction Through the Application of Genetic Algorithms. In Computer-based Modelling and Optimization in Transportation Springer International Publishing, pp. 321-331.
- Cascetta, E., Cantarella, G.E., Di Gangi, M., (1991). Evaluation of Control Strategies through a Doubly Dynamic Assignment Model, Transportation Research Rec. N.1306.
- Celikoglu, H.B., Dell'Orco, M., (2007). Mesoscopic simulation of a dynamic link loading process,Transportation Research C: Emerging Technologies, vol. 15, no. 5, pp. 329–344.
- Celikoglu, H.B. Gedizlioglu, E. Dell'Orco M., (2009). A node-based modeling approach for the continuous dynamic network loading problem. IEEE Transactions on Intelligent Transportation Systems, vol. 10, no. 1, pp. 165-174.
- Ceylan, H., Ceylan, H., (2012). A hybrid harmony search and TRANSYT hill climbing algorithm for signalized stochastic equilibrium transportation networks. Transportation Research C: Emerging Technologies, vol. 25, pp. 152-167.
- Chard B.M., Lines C.J., (1987). Transyt-the latest developments. Traffic Engineering and Control, 28 (1987), pp. 387–390.
- Daganzo, C.F., (1994). The cell-transmission model. Part 2: Network traffic, University of California, Berkeley, California.
- Daganzo, C.F. (1995). “The Cell Transmission Model ii: Network Traffic.” Transportation Research Part B 29, pp. 79–93.
- de Romph, E., (1994).A dynamic traffic assignment model—Theory and applications,Ph.D. dissertation, Fac. Civil Eng., Delft Univ. Technol., Delft, The Netherlands.
- Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., (2002). A Fast and Elitist Multi-Objective Genetic Algorithm : NSGA II. IEEE Trans. On Evolutionary Computation, vol.6, pp.182-197.
- Dell'Orco, M., (2006). A dynamic network loading model for mesosimulation in transportation systems,”European Journal of Operational Research, vol. 175, no. 3, pp. 1447–1454.
- Dial, R.B., (1971). A probabilistic multi-path traffic assignment model which obviates path enumeration. Transportation Research, vol. 5, pp. 83-111.

- Di Gangi M., (2011). Modeling Evacuation of a Transport System: Application of a Multimodal Mesoscopic Dynamic Traffic Assignment Model. *IEEE Transactions On Intelligent Transportation Systems*, Vol. 12, No.4, pp. 1157-1166
- Di Gangi, M., (1996). Modelling Dynamic Network Loading on Transportation Networks through a Continuous Packet Approach. *Mathematical Modelling of Systems*, vol.2, issue 3, pp. 175-196
- Di Gangi, M., (1992). Continuous-flow approach in dynamic network loading. In: *Second International CAPRI Seminar on Urban Traffic Networks, Compendium*, vol. 1.
- Doan, K. and Ukkusuri, S.V., (2012). On the holding back problem in the cell transmission based dynamic traffic assignment models, *Transportation Research Part B*, vol. 46, no. 9, pp. 1218–1238.
- Drake, J.S., Schofer, J.L., May, A.D., (1967). A Statistical Analysis of Speed Density Hypotheses. *Third International Symposium on the Theory of Traffic Flow Proceedings*, Elsevier North Holland, Inc. New York.
- Drew, D.R., (1968). *Traffic Flow Theory and Control*, McGraw-Hill Book Company, Chapter 12.
- Gartner, N.H., (1983). OPAC: A demand-responsive strategy for traffic signal control, *Transportation Research Record* 906, pp. 75-81.
- Gawron, C., (1998). *Simulation-Based Traffic Assignment; Computing User Equilibria in Large Street Networks*. University of Cologne.
- Gazis, D.C., Potts, R.B., (1963). The oversaturated intersection. *Proceedings of the 2nd International Symposium on Traffic Theory*, London, U.K., pp. 221-237.
- He, Y., (1997). *A flow-based approach to the dynamic traffic assignment problem: formulations, algorithms, and computer implementations* (Doctoral dissertation, Massachusetts Institute of Technology).
- Henry, J.J., Farges, J.L., and Tuffal, J., (1983). The PRODYN real time traffic algorithm, *Proceedings of the 4th IFAC-IFIP-IFORS conference on Control in Transportation Systems*, pp. 307-311.
- Hunt, P.B., Robertson, D.I., Bretherton, R.D. and Winton, R.I., (1981). SCOOT – A Traffic Responsive Method of Coordinating Signals. *RRL Report LR 1041*, Road Research Laboratory, U.K.
- Jayakrishnan, R., Mahmassani, H.S., and Hu, T.-Y., (1994). An Evaluation Tool for Advanced Traffic Information and Management Systems in Urban Network. *Transportation Research C*, 2, pp. 129-147.
- Kuwahara, M. and T. Akamatsu. (1997). Decomposition of the Reactive Dynamic Assignment with Queues for a Many-to-Many Origin-Destination Pattern. *Transportation Research Part B* 31.
- Leonard, D.R., Gower, P., Taylor, N.B., (1989). CONTRAM: Structure of the model. *TRRL Research Report 178*, Crowtorne.
- Luk, J.Y.K., (1984). Two traffic-responsive area traffic control methods: SCAT and SCOOT, *Traffic Engineering and Control*, 25, pp. 14-22.
- Mahut, M., Florian, M. and Tremblay, N., (2002). Application of a simulation-based dynamic traffic assignment model. *IEEE 5th International Conference on Intelligent Transportation Systems*, Singapore.
- Mauro, V., (2002). UTOPIA—Urban Traffic Control—Main Concepts, presented at the EU–China ITS Workshop, Beijing, China.
- Mauro, V., Di Taranto C., (1989). Utopia, in: *Proc. of the 2nd IFAC-IFIP-IFORS Symposium on Traffic Control and Transportation Systems*, pp. 575–597.
- Ozan, C., Baskan, O., Haldenbilen, S., & Ceylan, H., (2015). A modified reinforcement learning algorithm for solving coordinated signalized networks. *Transportation Research Part C: Emerging Technologies*, 54, pp. 40-55.
- Papageorgiou, M., Kotsialos, A., (1998). Short term traffic forecasting with metanet. *Daccord Short Term Forecasting Workshop TU Delft*.
- Ran, B., and Boyce, D.E., (1996). *Modeling Dynamic Transportation Networks: An Intelligent Transportation System Oriented Approach*. Second edition, Springer-Verlag, Berlin.
- Robertson, D.I., (1969). *TRANSYT: a traffic network study tool*. *RRL Report LR 253*, Road Research Laboratory, England.

- Robertson, D.I. and Bertherton, R.D., (1974). Optimum control of an intersection for any known sequence of vehicular arrivals, Proceedings of the 2nd IFAC-IFIP-IFORS Symposium on Traffic Control and Transportation system, Monte Carlo.
- Smith, M.J. (1983) The Existence of an Equilibrium Distribution of Arrivals at a Single Bottleneck. *Transportation Science* 18, pp. 385–394.
- Spiess, H. (1990). Technical note—Conical volume-delay functions. *Transportation Science*, 24(2), pp. 153-158.
- Stevanovic, A., Kergaye, C., and Martin, P.T., (2008). “Field Evaluation of SCATS Traffic Control in Park City, UT,” presented at 15th World Congress on ITS, New York City.
- Stevanovic, A., Kergaye, C., and Martin, P.T., (2009). “SCOOT and SCATS: A Closer Look into Their Operations,” 09-1672, Proceedings of the 88th Annual Meeting of the Transportation Research Board, Washington, D.C.
- Taylor, N.B. (2003) The CONTRAM Dynamic Traffic Assignment Model, *Networks and Spatial Economics*, Vol. 3, Issue 3, pp. 297-322.
- Underwood, R.T, (1961). Speed, Volume, and Density Relationship. *Quality and Theory of Traffic Flow*, Yale Bur. Highway Traffic, New Haven, Connecticut, pp.141-188.
- Vincent, R.A., Mitchell, A.I. and Robertson, D.I., (1980). User guide to TRANSYT version 8. Transport and Road Research Laboratory Report, LR888, Crowthorne, Berkshire, U.K.
- Van Aerde, M., and Yagar, S., (1988., *Dynamic Integrated Freeway/Traffic Signal Networks: Problems and Proposed Solutions*, *Transportation Research Record A*, Volume 22A, Number 6, pp. 435-443.

Appendix A: BPR-like link performance functions

The BPR function was developed in the late 1950's by fitting data collected on uncongested freeways and appeared in 1965 HCM as:

$$t(f) = (L/v_0) \cdot (1 + 0.15 (f/Q)^4)$$

where:

$t(f)$ is the mean travel time,
 L is the link length,
 v_0 is the free flow speed,
 f is the flow,
 Q is the capacity.

Afterwards, BPR-like link performance functions have been introduced for urban applications as:

$$t(f) = (L/v_0) \cdot (1 + a (f/Q)^b)$$

where:

a is a congestion parameter, such that $t(Q) = (a + 1) \cdot (L/v_0)$
 b is a shape parameter,
and Q also take into account the green time/cycle length ratio.

In terms of speed-flow relationship (fundamental diagram – stable regime) the above function can be written as:

$$v(f) = v_0 / (1 + a (f/Q)^b)$$

The original value of parameter a was 0.15, thus when flow (volume) equals the capacity, the congested travel time is only 15% greater of the free flow travel time and the standard BPR function may greatly over-estimates the speed at maximum capacity in particular in urban areas. Furthermore when the volume is very low, the predicted travel time is approximately equal to free-flow travel time.

To correctly estimate the travel time on urban links different values of parameters a and b are used such as $a = 2.0$ and $b = \{3.0, 3.5\}$ which seem to be more suitable values.

Such BPR-like functions, along with the standard BPR curve, are plotted in **Fig.7.13** for L equals 1 km, v_0 equals 50 kph and Q equals 2000 veh/h. The travel time, as expected, increases faster for urban link curve than for uncongested freeway curve. Overall, with BPR-like functions the travel time increases more quickly for small flow-capacity ratio whilst, on uncongested freeway, the travel

time stays relatively constant for low flow. From these considerations the original BPR function does not seem to be well suited for urban applications.

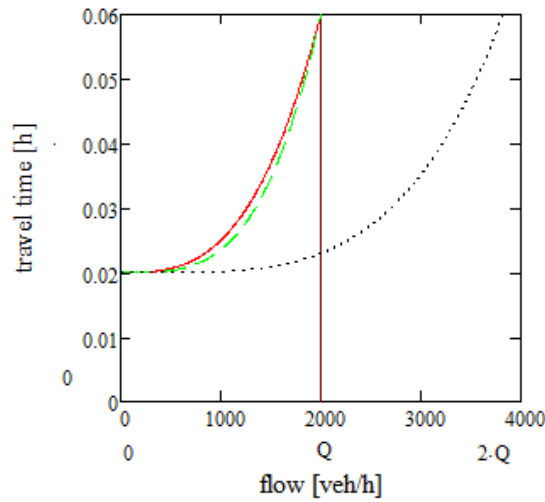


Fig.7.13 - Travel time over $L=1$ km vs. flow ($v_0=50$ kph, $Q=2000$ veh/h). Continuous red line refers to BPR like curve for $a=2.0$ and $b=3.0$. Dashed green line refers to BPR like curve for $a=2.0$ and $b=3.5$. Dotted grey line refers to BPR standard curve.

Coherent considerations can be made with respect to the speed-flow curves, as shown in **Fig.7.14** for $a = 2.0$ and $b = \{3.0;3.5\}$ only.

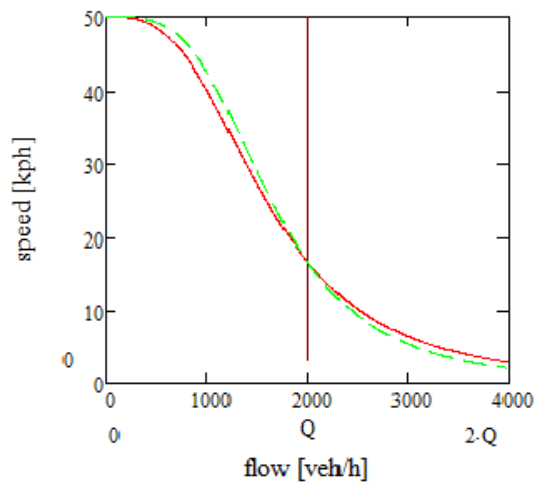


Fig.7.14 - Speed vs. flow (stable regime). Continuous red line refers to BPR like curve for $a=2.0$ and $b=3.0$. Dashed green line refers to BPR like curve for $a=2.0$ and $b=3.5$.

Spiess (1989) suggests the use of conical functions given as:

$$t(f, \alpha) = [2 + [\alpha^2 \cdot (1 - f/Q)^2 + (2\alpha - 1/2\alpha - 2)^2]^{0.5} - \alpha(1 - f/Q) - (2\alpha - 1/2\alpha - 2)] \cdot (1/v_0)$$

with $\alpha > 1$

to estimate travel time on urban links. A minor flexibility in modelling the effect of congestion could be observed in conical functions, differently from BPR-like. As a matter of fact in the conical functions only a shape parameter α is considered. Whichever is the value of α , $t(Q) = 2 \cdot L/v_0$. Conversely in the BPR-like function two parameters are identified: b which is a shape parameter and a which takes into account the congestion in such way that $t(Q) = (a+1) \cdot (L/v_0)$. As shown in **Fig.7.15**, BPR-like function with $a=1$ (say $t(Q) = 2 L/v_0$) and conical functions are not appreciably different.

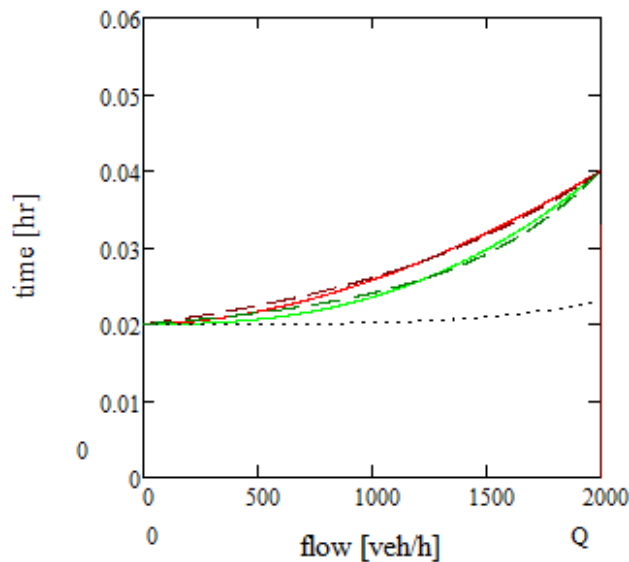


Fig.7.15 - Travel time over $L=1$ km vs. flow ($v_0=50$ kph, $Q=2000$ veh/h). Continuous red line refers to BPR like curve for $a=1.0$ and $b=1.8$; Continuous green line refers to BPR like curve for $a=1.0$ and $b=2.5$. Dashed red line refers to conical curve for $\alpha=2$; dashed green line refers to conical curve for $\alpha=3$. Dotted grey line refers to BPR standard curve.

Remembering that under steady-state condition the flow f is related to the density k and the mean speed v through the relation:

$$f = \rho \cdot v$$

the congested speed versus the density is given by solving the following equation:

$$v_0 / (1 + a \cdot (\rho v / Q)^b) - v = 0$$

or

$$(a/Q^b) \cdot \rho^b \cdot v^{b+1} + v - v_0 = 0$$

It can be shown that such equation always has exactly one positive solution for any positive value of ρ , thus the above polynomial equation defines an implicit function $v(\rho)$ between speed and density (with parameters a, b, Q, v_0).

In **Fig.7.16** the function $v(\rho)$ for $a = 2$ and for different values of b is graphically represented. As it arises from the graph, varying the value of b does not affect significantly the shape of the function. The Underwood speed-density function given as:

$$v_{Underwood}(\rho) = v_0 \cdot \exp(-(\rho/\rho_{crit}))$$

is also plotted to compare/validate the adopted curves.

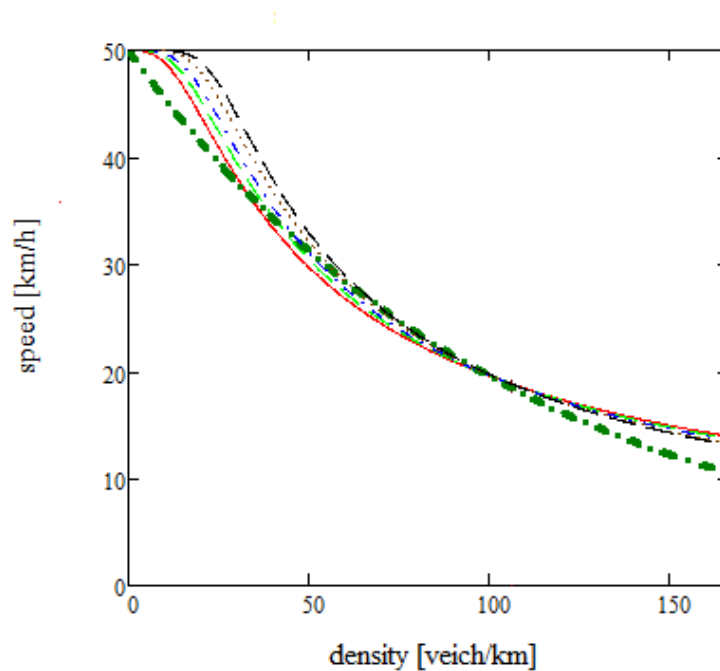


Fig.7.16 - Speed density function. Dashed black line refers to Underwood speed-density function; speed density function derived from the BPR like function are so identified: continuous red line for $a=2.0$ and $b=3.0$; continuous green line for $a=2.0$ and $b=3.5$; dotted blue line for $a=2.0$ and $b=4.0$; dashed brown line for $a=2.0$ and $b=5.0$; dashed black line for $a=2.0$ and $b=6.0$.

8. Network Signal Setting Design: metaheuristic optimisation methods

*** Cantarella, G. E., de Luca, S., Di Pace, R., Memoli, S., (2015). Network Signal Setting Design: metaheuristic optimisation methods. Transportation Research Part C, Emerging Technologies, Vol. (55), pp.24-45, DOI: 10.1016/j.trc.2015.03.032**

Department of Civil Engineering, University of Salerno Italy, EU

* For friendship policy, the authors order is organised in alphabetical order of the authors name.

Abstract

This paper aims to investigate the application of meta-heuristic optimisation methods to Network Signal Setting Design. The adopted approaches are (i) three step optimisation, in which first the stage matrix (stage composition and sequence), the green timings at each single junction are optimised, then the node offsets are computed in three successive steps; (ii) two step optimisation, in which the stage matrix is defined at a first step, then the green timings and the node offsets are computed at a second step. In both approaches the stage matrix optimisation is carried out through explicit complete enumeration.

In the first approach multi- criteria optimisation is followed for single junction signal setting design (green timings), whilst the coordination (node offsets) is approached through mono-criterion optimisation, as well as for the synchronisation (green timings and offsets) in the second approach.

A new traffic flow model mixing CTM and PDM has been applied. This model allows to explicitly represent horizontal queuing phenomena as well as dispersion along a link. Some meta-heuristic algorithms (i.e. genetic algorithms, hill climbing and simulated annealing) are investigated in order to solve the two problems

The proposed strategies are applied to two different layouts (a two junction arterial vs. a four junction network) and their effectiveness is evaluated by comparing the obtained results with those from benchmark approaches implementing mono-criterion optimisation only.

8.1 Introduction and motivation

Traffic lights are one of the most common ways to control a road junction network. The design of control variables can be formulated as an optimisation problem, often named Network Signal Setting Design (NSSD).

In accordance with literature, the main ongoing issues in connection with network signal setting design are related to:

- the decision variables involved in the optimisation (green timings, green scheduling, and node offsets);
- the number of objective functions (criteria) considered in the optimisation methods (mono-criterion vs. multi-criteria);
- the traffic flow model needed to compute delay indicators.

In terms of decision variables, two main approaches are adopted in literature: coordination and synchronisation. In the first case, optimisation may be achieved in two ways: (i) first the green scheduling, say the stage matrix is defined, then the green timings at each single junction and eventually the node offsets are computed in separated steps; (ii) the green scheduling and timings are computed together in the first step, the coordination is carried out in the second step.

In the case of synchronisation, the scheduling is carried out at a first step, say the stage matrix is defined, and the green timings and the coordination (node offsets) are computed at a second step for each junction in the network.

Two of the most straightforward software programs are: TRANSYT14²® (TRL, UK) and TRANSYT-7F® (FHWA, USA). Both are able to compute the signal timings, the offsets and the cycle length by combining a traffic flow model and a signal setting optimiser. Both may be used for coordination or synchronisation; in the former case Oscady Pro® (TRL, UK; Burrow, 1987) may be used for green timings and scheduling optimisation. TRANSYT14® generates several (but not all) significant stage sequences to be tested but the optimal solution is not endogenously generated, while TRANSYT-7F® is able to optimise the stage sequence for each single junction starting from the ring and barrier NEMA (i.e. National Electrical Manufacturers Association) phases. Further investigations, with respect to the stage sequence optimisation may be found in Park et al. (2000). They carried out a simulation framework made up of a mesoscopic flow simulator and Goldberg's genetic algorithm optimiser which, showed that the developed simulator provided better results than those obtained using the software TRANSYT-7F®. Other investigations may be found in Hadi and Wallace (1993; 1994).

As regards the solution methods, in TRANSYT14®, the Hill-Climbing and the Simulated Annealing are implemented, whereas in TRANSYT-7F®, the Hill

² A new recent release TRANSYT15® is available.

Climbing and some Genetic Algorithms may be used, but none of them are able to solve the multi-criteria optimisation (they optimise linear combinations of two criteria).

With regards to the latter case, some interesting contribution may be found in Ceylan (2006) which focused on the integration of a combined optimisation procedure (based on GAs and HC) with TRANSYT for the evaluation of a performance index and in Putha et al., (2012) which compared the GAs with the Ant Colony Optimisation, by applying the coordination approach at network level. Differently from previous authors that considered the PDM for traffic flow modelling, Lertworawanich, (2011) proposed the application of GAs for network traffic signal setting through CTM for traffic flow modelling. All above cited authors considered the combination (by weights) of objective functions as a proxy of multi-criteria optimisation.

Finally traffic models may be classified according to the level of detail adopted for representing the traffic systems. The first are microscopic models, which describe both the space-time behaviour of the systems' elements (i.e. vehicles and drivers) as well as their interactions at disaggregated level. The second are mesoscopic models, which represent traffic by groups of vehicles or a single vehicle, whereby the activities and interactions are not described at a detailed level. The final models are macroscopic flow models which describe the traffic at a high level of aggregation as a flow without distinguishing its constituent parts.

In the case of a signalised network, two main issues are to be addressed: (i) the modelling of the dispersion between interacting junctions, which is strictly related to the distance travelled on the connecting links and (ii) the spillback (i.e. the link blockage) and the merging and diverging modelling (i.e. the lane blockage).

All these phenomena may be observed in macroscopic traffic flow models. In general, the approach proposed by Robertson with the platoon dispersion model (PDM) in the 1969 is the most straightforward for modelling the dispersion of platoons by estimating vehicle arrivals at downstream locations based on an upstream vehicle departure profile and a desired traffic stream speed. However, this model shows a main weakness since it cannot describe the spillback phenomena and it does not model the effects of blocking back (i.e. horizontal queuing). Based on this consideration, recent enhancements in traffic models are related to the application of the cell transmission model (CTM) (Daganzo, 1994; Lo, 1999, 2001; Lo and Chow, 2004; Chow and Lo, 2007) where a link is represented (discretised) by 'cells' which experience a flow carried out as the minimum value between the (maximum) number of vehicles that can be 'sent' by the upstream cell and those that can be 'received' by the downstream cell. It should be noted that CTM cannot describe density variations.

Promising approaches leading for stochastic CTM have been proposed in Sumalee et al. (2011), in which the sending and receiving functions are reproduced by random parameters of the fundamental flow-density diagram. An approach for signalised arterials has been proposed in Jabari and Liu (2011), in which a variety of time headway distributions, that are dependent on traffic states, are simulated;

more extensive studies regarding density estimation have been dealt with in enhanced L(agged)-CTM by Szeto (2008). Other researchers have focused on the lane blockage simulation (Li and Chang, 2010; Li, 2011) by considering the ‘sub-cell concept’. Currently, in TRANSYT14[®] (Binning et al. 2008), the CTM may be adopted as an alternative to the PDM, for short distances whereas in the case of long distances the PDM is still preferred to the CTM.

On the basis of all the previous considerations, in this paper:

- the stage matrix (SM), defined by stage composition and stage sequence, can be obtained by enumerating (Ex.en) all possible stages; other methods such as Oscady, allows for green timings and scheduling optimisation; in this case the SM is obtained from the scheduling (see also Improta and Cantarella, 1984);
- the green timings (GT) and offsets (Off) are optimised;
- some meta-heuristic algorithms are analysed for the solution of the mono-criterion/multi-criteria optimisation and applied to two different layout configurations (a two junction arterial vs. a four junction network);
- finally, in terms of traffic flow modelling, an extension of the CTM is proposed to include the traffic dispersion (CT&PDM).

Summing up, the paper contributions are reported in bold in the following **Table 8.1** where criteria (Crt) and objective functions (Of) are considered. A further comparison is shown with regards to the optimisation (Opt) in terms of Benchmark (Ben), Proposal (*) and Future Perspectives (FP).

Table 8.1 - Paper contributions, Benchmarks and Future perspective on Network signal setting design

Opt	var	Strategy								
		three step		two step				one step		
Crt	SM	Ex.en	Ex.en	Ex.en.	Ex.en.	Ex.en	mono	multi	mono	multi
	GT	mono	multi	mono	mono	multi	mono	mono		
	Off	mono	mono				mono	mono		
Of	SM	ND	ND	CF	CF	CF	CF	CF & of	TD	TD & of
	GT	CF	CF & of	TD	TD	TD & of	TD	TD		
	Off	TD	TD				TD	TD		
Tools		Ben	*	Ben	*	FP	Ben	FP	FP	FP
	SM	Ex.en	Ex.en	Ex.en	Ex.en					
	GT	OSCADY (GT)	UNISA (GT)				OSCADY (GT&S)			
	Off	TRANSYT (C) PDM/CTM	UNISA (C) CT&PDM	TRANSYT (Sy) PDM/CTM	UNISA (Sy) CT&PDM	x	TRANSYT (C) PDM/CTM	x	x	x

-C: coordination; Sy: Synchronisation; GT&S: Green timing and Scheduling

-CF: capacity factor; TD: total delay

- of: generic objective function

-PDM/CTM: platoon dispersion model or cell transmission model; CT&PDM: cell transmission and platoon dispersion model

The paper is organised as follows: the proposed strategies are described in **section 8.2**, after some preliminary considerations the main purposes of the paper are reported in **sub-section 8.2.4** where the proposed traffic flow model is described, in **sub-section 8.2.5** where the optimisation problem is shown and in **sub-section 8.2.6** where the adopted algorithms are summarised; in **section 8.3**, some computational results are shown; finally, conclusions and research perspectives are discussed in **section 8.4**.

8.2 Proposed strategies

This paper dealt with the network signal setting design through the application of two approaches: three step optimisation and two step optimisation as defined in the previous section.

The basic notations, the constraints and the optimisation problems are described below.

8.2.1 Variables and constraints

Assuming that the green scheduling is described by the stage matrix (i.e. the stage matrix composition and sequence), let

c be the cycle length, assumed known or as a decision variable (common to all junctions);

for each junction (not explicitly indicated)

t_j be the duration of stage j as a decision variable;

t_{ar} , be the so-called all red period at the end of each stage to allow the safe clearance of the junction, assumed known (and constant for simplicity's sake);

Δ be the approach-stage incidence matrix (or stage matrix for short), with entries $\delta_{kj} = 1$ if approach k receives green during stage j and 0 otherwise, assumed known;

l_k be the lost time for approach k , assumed known;

$g_k = \sum_j \delta_{kj} t_j - t_{ar} - l_k$ be the effective green for approach k ;

$r_k = c - g_k$ be the effective red for approach k ;

y_k be the arrival flow for approach k , assumed known;

s_k be the saturation flow for approach k , assumed known;

$(s_k \cdot g_k) / (c \cdot y_k)$ be the capacity factor for approach k ;

and for each junction in the network

ϕ_i be the node offset as the time shift between the start of the plan for the junction i and the start of the reference plan, say the plan of the junction number 1, $\phi_1 = 0$. Given such a reference value, all the other $m-1$ nodes, where m is the number of junctions in the network, are independent variables.

As well known, compute total delay through a traffic flow model needs the link offset between each pair of adjacent junctions, $\phi_{ij} = -\phi_{ji}$.

Let the junction network be represented by an undirected graph with 1 node for each junction and an edge for each pair of adjacent junctions (the actually traffic directions are irrelevant). According to this representation

- if such a network is loop less, all the $m-1$ link offsets are independent (as many as the independent node offsets) and may be used as decision variables; Arterials are a special case of such a network;
- on the other hand, if the network contains k independent loops, the number of independent link offsets will be equal to $m-k$; in this case it is better to use the $m-1$ independent node offsets as optimisation variables.

Some constraints were introduced in order to guarantee:

stage durations being non-negative

$$t_j \geq 0 \quad \forall j$$

effective green being non-negative

$$g_k \geq 0 \quad \forall k$$

this constraint is usually guaranteed by the non-negative stage duration, but for a too short cycle length with regard to the values of all-red period length and lost times, say

$$\sum_j \text{MAX}_k (\delta_{kj} l_k + t_{ar}) \geq c$$

consistency among the stage durations and the cycle length

$$\sum_j t_j = c$$

the minimum value of the effective green timing

$$g_k \geq g_{min} \quad \forall k$$

A further constraint was included in order to guarantee that the capacity factor must be greater than 1 (or any other value)

$$((s_k \cdot g_k) / (c \cdot y_k) - 1) \geq 0 \quad \forall k$$

Such a constraint may be added only after having checked that the maximum junction capacity factor for each approach k in the junction i is greater than 1, otherwise a solution may not exist whatever the objective function is.

Finally let assume

$$c \geq \phi_i \geq 0.$$

8.2.2 Objective functions

At a single junction, the objective functions in the optimisation problems were:

the junction capacity factor computed as

$$CF = \text{MAX}_k (s_k \cdot g_k) / (c \cdot y_k) \quad (\text{Eq.8.1})$$

the total delay computed

- for non-interacting approaches (isolated or external junctions) by the two terms Webster's formula (Webster, 1958) as

$$TD = \sum_k y_k \cdot (0.45 \cdot c \cdot (1 - g_k/c)^2 / (1 - y_k/s_k) + y_k \cdot 0.45 / (s_k \cdot g_k/c) \cdot ((g_k/c) \cdot (s_k/y_k) - 1)) \quad (\text{Eq.8.2})$$

- for the interacting approaches by evaluating vehicles queuing interval by interval and considering input as the flow obtained by cyclic flow profiles. A more detailed expression consistent with the traffic flow modelling will be described in **subsection 8.2.4**.

Further objective functions could be considered such as the queue length, the number of stops etc.

8.2.3 Stage composition and sequence

A set of feasible stages is built up by assuming that all the approaches in a stage are mutually compatible and each stage is maximal. Given the set of stages, that is the stage composition, some of them (not necessary all) produce a feasible sequence if all approaches have green at least in one stage. Commonly if an approach has green in more than one stage, those stages are adjacent (that is relevant for sequence with at least four stages).

All possible sequences can be grouped in equivalent classes; each class contains all sequences leading to the same green timings, the same TDs and CFs. Optimal offsets generally change in a consistent way among all the sequences in an equivalent class, thus:

- when only two stages are available $\{(1,2)\}$ two possible sequences $\{(1,2); (2,1)\}$ are equivalent and only one equivalent class exists;

- when three stages are available $\{1,2,3\}$, two equivalent classes exist, namely, the class of sequences generated by translation $\{(2,3,1); (3,1,2)\}$ and the class of sequences generated by mirroring $\{(3,2,1); (2,1,3); (1,3,2)\}$.

As dealt with in a future paper, this analysis may be further extended when four or more stages are available.

Let m be the number of junctions in the network and n_i the number of equivalent classes for each junction i , $\prod_{i=1:m} n_i^{m_i}$ combinations of classes can be generated and must be tested to achieve optimum.

In this paper an explicit enumeration will be carried out. The implicit enumeration will be addressed in future papers.

8.2.4 Traffic flow model

The most straightforward continuous traffic flow model is the first order model developed concurrently by Lighthill & Whitham (1955) and Richards (1956), based on the assumption that the number of vehicles is conserved between any two points if there are no entrances (sources) or exits (sinks). This produces a continuous model known as the Lighthill-Whitham-Richards (LWR). To solve the LWR space continuous problem the Cell Transmission Model was introduced (Daganzo; 1994). Since CTM assumes the same speed for all the vehicles on a road, it cannot fully predict realistic traffic flow behaviour as the platoons keep the same density when moving from the upstream stop-line section to the downstream section, and all vehicles travel at the same free flow speed.

One of the aim of this paper was to implement a traffic flow model which overcame the limitations of PDM and CTM. In particular major drawbacks of the PDM are in queuing simulation since queues are assumed vertical; on the other hand the CTM include the horizontal queuing at the cost of not considering the platoon dispersion. Finally, the proposed model allows horizontal queues and platoon dispersion modelling.

a Cell transmission model

In a cell-transmission model, a link is represented by a sequence of small sections (cells) and a record of the cell contents (number of vehicles) as time goes on is kept. This record is updated at closely spaced instants (clock ticks) by calculating the number of vehicles that crosses the boundary separating each pair of adjoining cells during the corresponding clock interval.

The main drawback of the CTM is that the density is equally distributed (such as constant) over the cell. The number of vehicles in the cell depends on its spatial extension and smaller is the cell length (thus smaller is the clock tick), lower is the error in density estimation made by finite approximation. On the other hand,

particularly in case of large network, the number of cells (and computational complexity of the problem) will increase.

Thus in the proposed model the clock tick is around 1 second and in order to improve the density approximation as discrete variable, an additional equation based on speed-density relationship, was included.

In the CTM formulation, let

n_i , be the number of vehicles on the cell i ;

Q_i , be the maximum flow rate in cell i

d_{i+1} , be the wave speed coefficient of cell $i+1$

N_{i+1} , be the maximum number of vehicles present in the cell $i+1$.

The average flow $Y_i(t)$ on the link i from clock tick t to clock tick $t+1$ is the result of a comparison between the maximum number of vehicles that can be “sent” by the cell directly upstream of the boundary

$$S_i(t) = \min\{Q_i, n_i\} \quad (\text{Eq.8.3})$$

and those that can be “received” by the downstream cell

$$R_i(t) = \min\{Q_i, d_i[N_{i+1} - n_i]\} \quad (\text{Eq.8.4})$$

Hence the flow $Y_i(t)$ can be written as:

$$Y_i(t) = \min\{S_i(t), R_{i+1}(t)\} \quad (\text{Eq.8.5})$$

b Cell Transmission and Platoon Dispersion Model (CT&PDM)

As shown above, in the CTM equations, due to the assumption that all vehicles travel at the same speed (keeping as such the same density) for getting in the downstream section, no platoon dispersion may be detected. To overcome this shortfall, the flow propagation has been modelled by employing for each cell the well-known Drake speed-density relationship in which the corresponding flow, $X_i(t)$, is computed as follows:

$$X_i(t) = k_i(t) v_0 \exp[-0.5 (k_i(t) / k_m)^2] \quad (\text{Eq.8.6})$$

where:

$k_i(t) = [n_i(t) + n_{i+1}(t)] / 2L$: is the density of cell i and cell $i+1$ at time t , being L the length of the cell;

v_0 : is the free-flow-speed;

k_m : is the traffic density at maximum flow.

The flow $Y_i(t)$ may be so formulated as:

$$Y_i(t) = \min \{S_i(t), R_{i+1}(t), X_i(t)\} \quad (\text{Eq.8.7})$$

c Total Delay computation

Let the period of analysis T be a multiple of the cycle duration in so far as the quantities relating to a cycle were repeated in the subsequent cycles. The interval of duration T was divided into intervals of equal duration $\Delta\tau^3$. For the computation of the total delay the cumulated input flows, $Ciy_k^j(i)$, and those output, $Coy_k^j(i)$, on the stop line on approach k of junction j , in the subsequent sub-intervals i were compared.

The Deterministic Total Delay (DTD_k^j) cumulated in the interval $[0, T]$ for approach k of junction j was then given by the following expression:

$$DTD_k^j = \sum_{i=1..T/\Delta\tau} (Ciy_k^j(i \times \Delta\tau) - Coy_k^j(i \times \Delta\tau)) \Delta\tau \quad (\text{Eq.8.8})$$

Thus delay experienced on an interacting approach is a function of the offsets between the timing plans. In fact, such a delay depends on the output flow in the downstream junction which is obtained by starting from the input flow in the upstream junction through the phenomenon of dispersion.

Let s_k^j be the saturation flow on approach k of junction j , the Stochastic and Oversaturation component of Total Delay $SOTD_k^j$ on approach k of junction j is computed using the following expression

$$SOTD_k^j = \{ [(y_k^j - s_k^j)^2 + (4y_k^j/T)]^{0.5} + (y_k^j - s_k^j) \} T/4 \quad (\text{Eq.8.9})$$

and considering the average of the values of the cyclic flow profile along the connecting link arriving at approach k of considered junction j , y_k^j as input flow.

An example of the model is shown in following with regard to cyclic flow profile and with regard to the flow progression over network links.

8.2.5 Optimisation problems

The paper dealt with the application of two optimisation strategies:

- A three step optimisation; in this case the scheduling, the green timings $A(1)$ at each single junction and the offsets $A(2)$ were computed in three separate steps;
- B two step optimisation; the decision variables might be computed in two steps.

³ in our application a clock tick of 1 second were adopted.

In particular, with regards to the latter this could be applied by following one of two feasible alternative combinations of variables:

- B.1) in the first combination (Sc&Gt-Off), the scheduling (Sc) and green timings (Gt) were computed at a first step, and the offsets (Off) were computed at a second step;
- B.2) in the second combination (Sc-S), the scheduling was carried out at a first step by defining the stage matrix for each junction, then the green timings and the offsets (e.g. the synchronisation approach, S) were computed at a second step.

In this paper, the A and B.2 strategies were adopted. In **Fig.8.1** a general framework of such strategies is shown. The stage- sequence incidence matrix for each single junction was generated externally and then iteratively embedded in the NSSD until the optimum solution was reached.

The explicit enumeration was ended when the iteration $k = \prod_{i=1:m} n_i^{m_i}$.

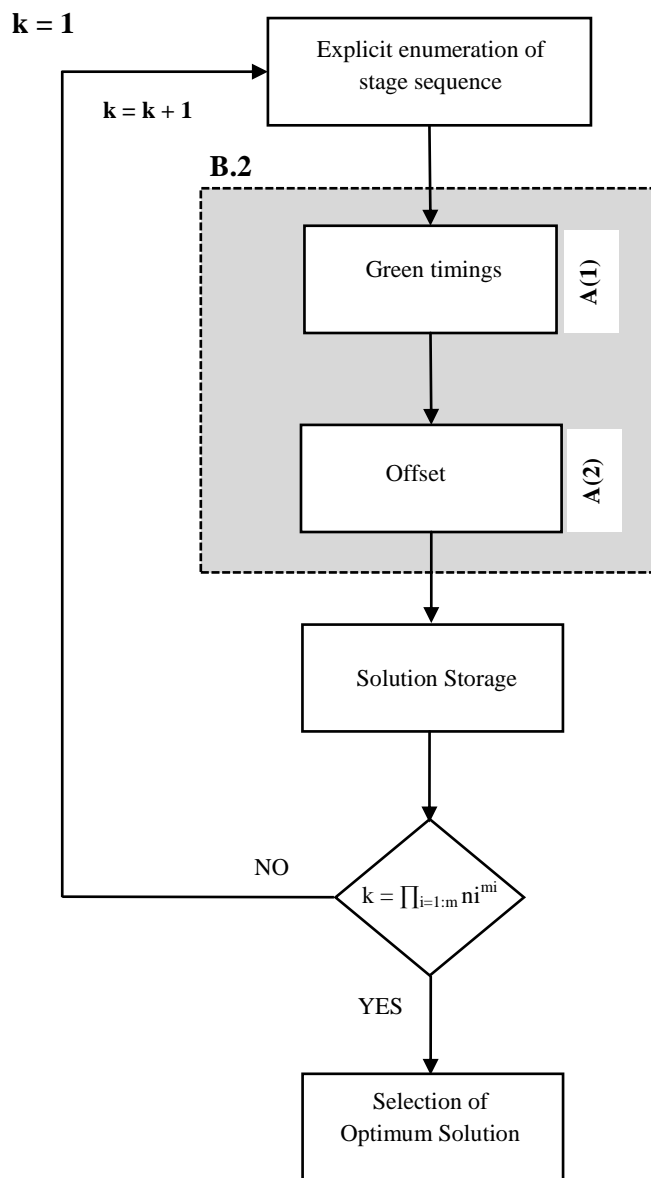


Fig.8.1 - General Framework of A and B.2 optimisation strategies (for each junction)

In three step optimisation (A) once all the sequences are explicitly enumerated, multi-criteria Genetic Algorithms based on total delay minimisation and capacity factor maximisation were applied for green timings at each single junction; the minimisation of the network total delay, with respect to the independent link offsets (loop less network) or node offsets (network with loops) was carried out by

applying a Hill-Climbing algorithm; the scheduling optimisation was computed through explicit enumeration of all the possible sequences.

In the two step optimisation (B.2), the green timings and the offsets were computed simultaneously (Sc-S), by considering mono-criterion optimisation based on total delay minimisation. The adopted algorithm was the Simulated Annealing.

In **Table 8.1** and **Table 8.2** each strategy is summarised with respect to the optimisation method and the objective function; in particular, bold cases refer to applications described in **section 8.3**.

Table 8.2 - Overview of three step strategy (A) w.r.t. Optimisation methods/objective functions

Strategy	Variables		Variables	
	GT	Offsets	GT	Offsets
	Optimisation method		Objective functions	
A	Mono	Mono	TD / CF	
	Multi	Mono	TD & CF	TD

Table 8.3 - Overview of two step strategy (B.2) w.r.t. Optimisation methods/objective functions

Strategy	Variables		Variables	
	GT	Offsets	GT	Offsets
	Optimisation method		Objective functions	
B.2	Mono		TD	
	Multi		TD & o.f.*	

*o.f.: It represents a generic objective function

8.2.6 Solution algorithms

Some main issues should be addressed to solve the NSSD: i) the optimisation approach, mono/multi-criteria functions; ii) the variables in the single junction optimisation or in the network optimisation; iii) the network layout (arterial as network loop less and network with at least one loop).

Based on all the previous considerations, meta-heuristic algorithms are generally adopted since they allowed for the approximation of the optimal solution of difficult combinatorial problems. In general, these approaches have evolved through interactions and analogies derived from biological, physical, computer and decision making sciences (Genetic Algorithms, GAs, Hill-Climbing, HC, Simulated Annealing, SA and other not used in this paper). To get a fully operational algorithms from a meta-heuristic requires the specifications of several functions and/or parameters whose meaning depends on the meta-heuristic itself.

As better described in **sub-section 8.2.5** we cope with different optimisation problems; for each of them the selection of an algorithm was based on the trade-off between the effectiveness of the algorithm related to the space solution exploration, the parameters to be set and the computational effort depending on the algorithm

complexity. Some considerations about benefits and drawbacks of the algorithms used in this paper will be described in following.

In SA the number of the parameters to be set is lower than the GAs and in terms of computational effort, the performance is generally better. Furthermore, SA can easily handle changes in the objective function, while with GAs the selection of the solution based on the fitness function may complicate the search for the optimum. On the other hand for highly dimension space of solution GAs are generally more effective than SA, for their flexibility due to the two genetic operators, crossover and mutation at the cost of a higher number of parameters to be set (for unfamiliar readers a more detailed description of the GAs is given in Appendix A)

GAs, in particular Non Dominated Sorting Genetic Algorithms II (NSGA-II), guarantee the exploration of a wider range of a solutions space, allowing for the computation of an approximation of the entire Pareto front. This provides a higher resilience in multi-criteria optimisation problems.

The HC algorithm uses a small amount of memory with respect to other approaches and it is characterised by lower processing time. One of the main drawbacks of the HC algorithm is the risk of hitting a local minimum/maximum. When the algorithm finds a local minimum/maximum value, which is lower/higher than the others in the neighbourhood, it freezes. However, this does not exclude the presence of far global minima with lower values. The HC search has the same drawbacks of a greedy search. It moves quickly to the best node by also using a ‘short-sighted’ strategy. For this reason its application is generally associated with optimisation problems without large range of a solutions space.

In order to clearly summarise the adopted approaches as well as highlighting the correspondence between the problems described previously and the algorithms, two further columns are added to **Table 8.1** and **Table 8.2** leading to **Table 8.4** for three step optimisation (A) and **Table 8.5** for two step optimisation (B.2).

Table 8.4 - Overview of three step strategy(A) w.r.t. Optimisation methods/objective functions including Algorithms

Strategy	layout	Variables		Variables		Variables	
		GT	Offsets	GT	Offsets	GT	Offsets
		Optimisation method		Objective functions		Algorithms	
A	Two junction arterial	Mono	Mono	TD / CF		HC	HC
		Multi	Mono	TD & CF	TD	GAs	HC
	Four junction network	Mono	Mono	TD / CF		HC	HC
		Multi	Mono	TD & CF		GAs	HC

Table 8.5 - Overview of two step strategy (B.2) w.r.t. Optimisation methods/objective functions including Algorithms

Strategy	layout	Variables		Variables		Variables	
		GT	Offsets	GT	Offsets	GT	Offsets
		Optimisation method		Objective functions		Algorithms	
B.2	Two junction arterial	Mono		TD			HC
		Multi		TD & o.f.*			GAs
	Four junction network		Mono		TD		SA
			Mono		TD & o.f.*		GAs

*o.f.: It represents a generic objective function

8.3 Numerical Applications

In this section, the numerical results obtained by the proposed approaches and benchmark tools are compared; in particular, different layouts, a two junction arterial and a four junction network, different meta-heuristics and different traffic flow models are analysed.

The adopted strategies are shown in **Table 8.6**, in particular:

- the approaches are the UNISA, as described in **section 8.2**, vs. the benchmark approach, OSCADY PRO[®] /TRANSYT14[®] - TRL software;
- the traffic flow model (TM) is the cell transmission and platoon dispersion model (CT&PDM), described in **sub-section 8.2.4** vs TRANSYT14[®] traffic flow models, i.e. the Platoon Dispersion Model (PDM) or the Cell Transmission Model (CTM)⁴;
- the optimisation strategies are three step, A, and two step, B.2;
- -the algorithms are HC,GAs and SA.

Table 8.6 - Adopted strategies

Layout	Strategy	UNISA	TM	TRL (OSCADY& TRANSYT ⁵)	TM
Two junction arterial	A	UNISA-GA&HC	CT&PDM	OSCADY&TRANSYT-HC	PDM or CTM
Four junction network	A B.2	{ UNISA-GA&HC UNISA-SA		{ OSCADY&TRANSYT-HC TRANSYT-SA	

⁴ TRANSYT14 may be applied with either traffic models (alternatively but not simultaneously).

⁵ The authors are aware that the total delay is analytically computed via different ways in UNISA and TRANSYT14.

The implementation of such strategies was worked out using a MATLAB (Release 2013b) code provided by the authors. All the applications were run on a PC having an Intel(R) Xeon(R) CPU E5-1603, clocked at 2.8GHz and with 4GB of RAM.

8.3.1 Two junction arterial

In this section the performances of three step method (A) applied to a two junction arterial (as a very simple example of a loop less network), are described. The layout together with the stage matrix and the main input data are shown in **Fig.8. 2**. Once defined the connecting link between the two junctions their interaction is simulated through the traffic flow model CT&PDM.

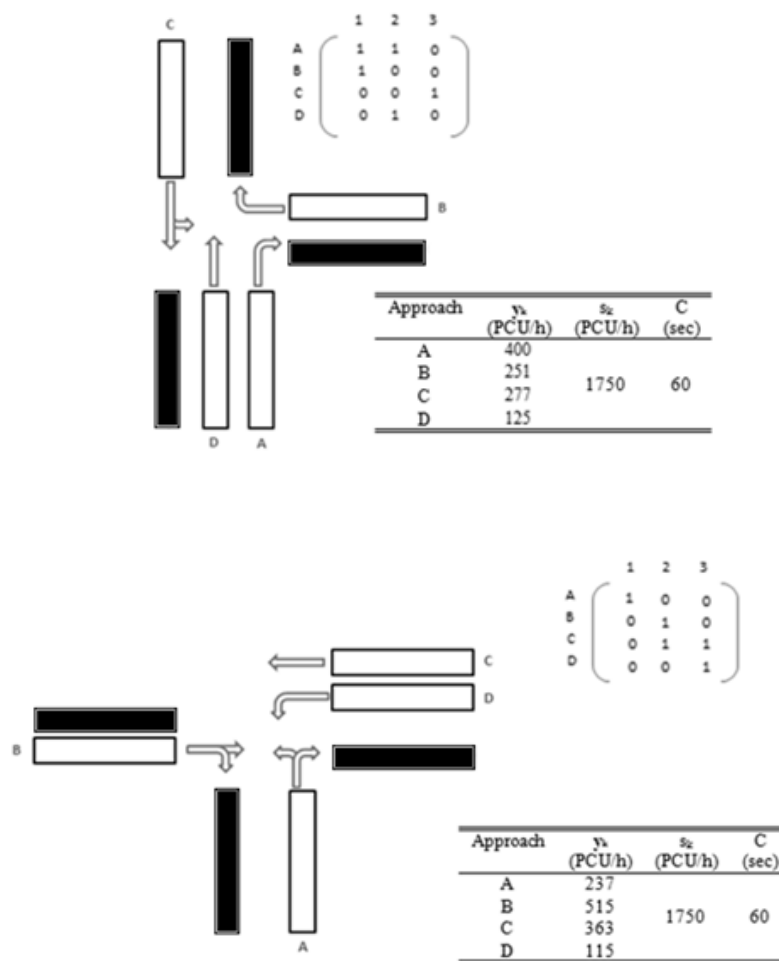


Fig.8. 2 - Layout, Stage matrix, characteristics for junction 1 (up) and junction 2 (bottom)

Firstly UNISA-GA&HC CT&PDM was implemented, through the application of multi-criteria GAs to achieve single junction green timings (see **Table 8.7** and **Table 8.8**) and mono-criterion HC to achieve optimal offsets (see **Table 8.10** and schematic representation in **Fig.8. 3**), by simulating traffic flow with CT&PDM. Successively a comparison, in terms of offsets and total delay, with OSCADY&TRANSYT–HC (PDM) and OSCADY&TRANSYT–HC (CTM) was carried out (see **Table 8.10**). In OSCADY PRO® a mono criterion optimisation, based on the maximisation of single junction capacity factor, was used to achieve single junction timings. In TRANSYT14® HC a mono-criterion optimisation, based on the minimisation of the arterial total delay, to get optimal offsets, was adopted, simulating alternatively traffic flows by PDM or CTM. Graphical outputs are shown in **Fig.8. 4**.

All applications were carried out considering a fixed value of the cycle length computed by following the Webster indication. According to the procedure defined in **sub-section 8.2.3**, let 2 be the number of junctions and 2 the equivalent classes, 4 combinations of sensitive sequences on arterial total delay were tried as shown in **Table 8.9**. The analysis was carried out in OSCADY&TRANSYT–HC (PDM) and as shown in the table, the optimal sequence corresponds to (1,2,3) for junction 1 and (3,2,1) for junction 2. **Table 8.7** and **Table 8.8** show how the results carried out are affected by the trade-off between total delay (minimisation) and capacity factor (maximisation) induced by the multi-criteria procedure. In general, a different effect may be reached by considering the capacity factor as a constraint and the total delay as an objective function, to be minimised; in this latter case, in fact, a dominance effect of the total delay is expected in terms of a performance indicator.

Table 8.7 - Multi-criteria optimisation based on Total delay minimisation and Capacity Factor maximisation at junction 1 (Pop size = 20; Mutation rate = 0.8; Crossover rate = 0.0001). UNISA GA&HC

Stream	Green timings [s]	Total Delay [PCU-hr/hr]	Capacity Factor
A	36	1.32	2.20
B	23	1.55	2.03
C	24	1.69	2.11
D	13	0.99	2.04

Table 8.8 - Multi-criteria optimisation based on Total delay minimisation and Capacity Factor maximisation at junction 2 (Pop size = 20; Mutation rate = 0.8; Crossover rate = 0.0001). UNISA GA&HC

Stream	Green timings [s]	Total Delay [PCU-hr/hr]	Capacity Factor
A	19	1.57	1.81
B	30	2.89	1.40
C	41	0.97	2.80
D	11	2.64	1.48

Table 8.9 - Stage sequences vs. TD (fixed length between junctions equal to 500 m); equivalent sequences are in curly brackets

Stage sequence junction 1*	Stage sequence Junction 2**	TD [PCU hr/hr]
(1,2,3); {(2,3,1); (3,1,2)}	(1,2,3); {(2,3,1); (3,1,2)}	15.18
(1,2,3); {(2,3,1); (3,1,2)}	(3,2,1); {(1,3,2); (2,1,3)}	13.83
(3,2,1); {(1,3,2); (2,1,3)}	(1,2,3); {(2,3,1); (3,1,2)}	21.57
(3,2,1); {(1,3,2); (2,1,3)}	(3,2,1); {(1,3,2); (2,1,3)}	17.01

* for junction 1 (stage 1: A,B/stage 2: A,D/stage 3: C);**for junction 2 (stage 1: A, stage 2: B,C/stage 3: C,D).

For each single junction, the green timings were obtained through the GAs application, and using the optimal stage sequences reported above, a sensitivity analyses of TD with respect to distance between two junctions was carried out. Starting from the stages duration obtained through the GAs implementations, once fixed four layouts based on connecting link lengths variability (i.e. from junction 1 to junction 2: 500 m; 400 m; 300 m; 200 m), the arterial total delay and the offset between the signal timing plans of the interacting junctions were carried out through traffic flow modelling. The considered stage sequences will be consistent with results above obtained (see **Table 8.9**) for a given distance between junctions (L=500m).

Table 8. 10 - Numerical results for OSCADY&TRANSYT-HC (PDM), OSCADY&TRANSYT-HC (CTM) and UNISA-GA&HC (CT&PDM) w.r.t. length variations

Length [m]	OSCADY&TRANSYT HC (PDM)			OSCADY&TRANSYT HC (CTM)			UNISA GA&HC (CT&PDM)		
	Offset [sec]	TD [PCU- hr/hr]	DOS* [%]	Offset [sec]	TD [PCU- hr/hr]	DOS* [%]	Offset [sec]	TD [PCU- hr/hr]	DOS* [%]
500	55	13.83		47	15.84		45	10.12	
400	44	12.77	71	39	14.41	80	34	11.51	67
300	34	11.38		24	14.96		22	11.90	
200	22	9.85		13	26.67		10	23.97	

* DOS: is the maximum degree of saturation between the two junctions

With reference to the results shown in **Table 8. 10** two main observations should be made. The first one is related to the estimation of the maximum degree of saturation between the junctions which was lower in UNISA-GA&HC (CT&PDM) than OSCADY&TRANSYT-HC (PDM/CTM). That was justified by the different structure of the objective function adopted in the single junction signal setting. Indeed, as above described, in UNISA-GA&HC a trade-off between total delay (to be minimised) and capacity factor (to be maximised) was looked for.

The second was related to the total delay/offset estimation with respect to the length variation; as shown in **Fig.8. 4a**, offset almost linearly increases with length for all the optimisation strategies. This trend might be explained by a decrease in the vehicles cruise time which forced a temporary close start of a downstream signal plan compared to the upstream one. With regards to the total delay (see **Fig.8. 4b**) it was noted that such an indicator in OSCADY&TRANSYT-HC (PDM) decreased against link length for distances closer than 300 m, since no horizontal queues were modelled. On the other hand similar trends were observed in OSCADY&TRANSYT-HC (CTM) and UNISA-GA&HC (CT&PDM) and lower total delays were obtained from the latter since a dispersion function was applied. In conclusion, the proposed traffic flow model seemed to overcome the limitations both of PDM (in modelling horizontal queues) and CTM (in modelling platoon dispersion).

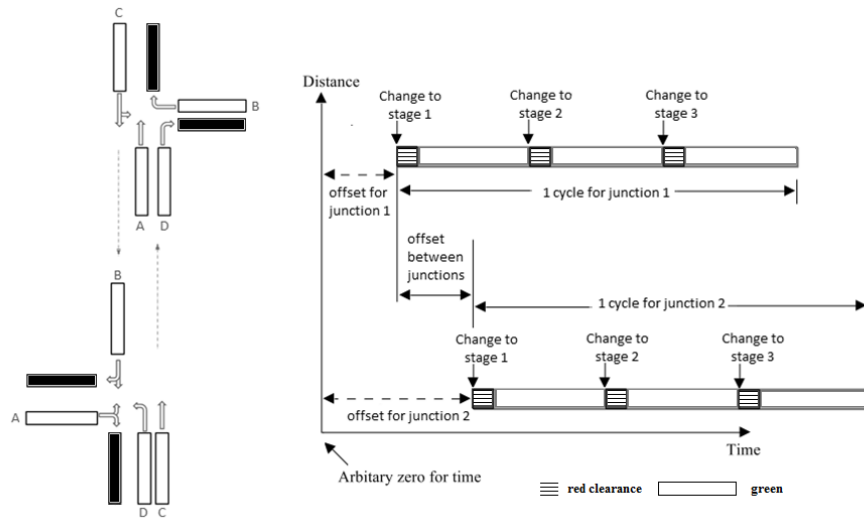


Fig.8. 3 - Schematic representation of the offsets (green and red bars are not proportional to actual values)

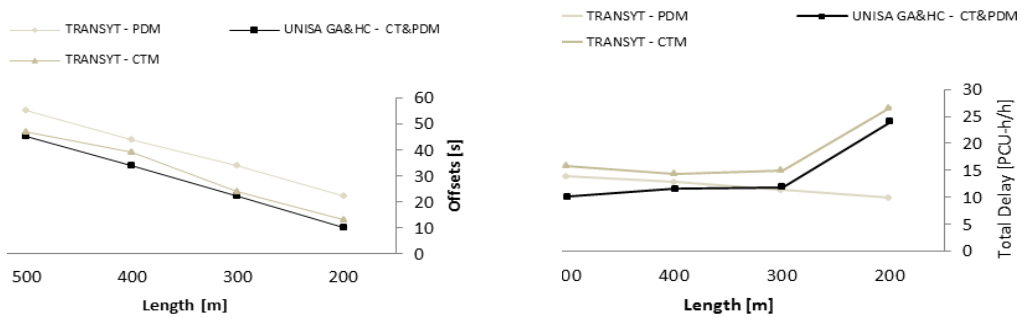


Fig.8. 4 - a) Optimal link offset w.r.t. length; b) Total Delay w.r.t. length

The results obtained in the two junction arterial showed the effectiveness of the adopted model (CT&PDM) in simulating traffic flows approaching signalised junctions further than queue forming phenomena along the connecting link.

8.3.2 Four junction network

In this section, the design strategies and optimisation algorithms were then analysed in more depth through an application to a network with four interacting signalised junctions forming a loop (shown in **Fig.8. 5** as a square for simplicity's sake).

Firstly, three step strategy (A) through UNISA-GA&HC (CT&PDM) was carried out. The NSGA-II algorithm was applied to obtain single junction timings while optimal link offsets were achieved through a HC algorithm.

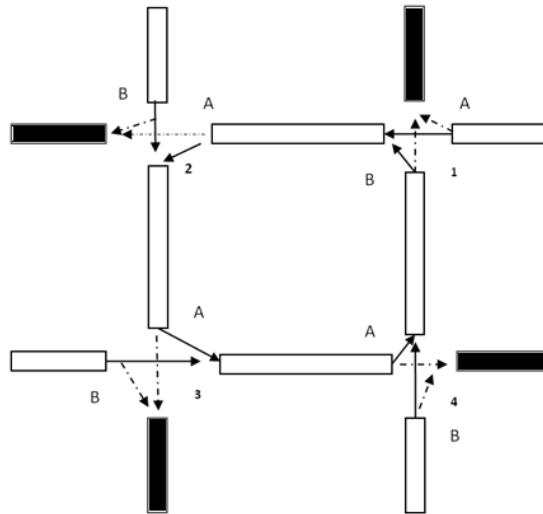


Fig.8. 5 - Four junction network layout

The results, collected in terms of link offsets, network total delay and network degree of saturation, DOS, were then subsequently compared with those obtained by TRANSYT14® - TRL implementation, where mono-criterion optimisations both to achieve single junction timings (obtained by internally implementing OSCADY PRO® - TRL) and optimum link offsets (here a HC was adopted, selected from a choice set HC/ SA) were applied.

Further comparisons were subsequently carried out with reference to two step strategy (B.2) where single junction green timings were designed simultaneously with link offsets considering the minimisation of network total delay. The SA algorithm, as described in **sub-section 0**, was applied.

In terms of flow simulation modelling, a detailed comparison was carried out among TRANSYT14® - TRL (PDM/CTM) and UNISA (CT&PDM). Such a match focuses on the model's capability in capturing both dispersion and traffic bottleneck phenomena when length variations over the link composing the network occurred (once given fixed flows).

a. Threestep optimisation (A):

Once the stage matrix for each junction was fixed as well as the value of the cycle length computed by the Webster indication (equal to 90 sec), a multi-criteria optimisation (minimisation of total delay and maximisation of capacity factor) performed by NSGA-II algorithm was performed to achieve single junction timings.

All information related to the network, in terms of flow and saturation flow for each approach, are summarised in **Table 8.11**.

Table 8.11 - Flow and saturation flow for each approach

Junction	Stage	Yk [veh/h]	Sk [veh/h]
1	A	400	1750
	B	245	
2	A	390	
	B	227	
3	A	422	
	B	286	
4	A	375	
	B	191	

Results carried out for each junction with regard to the optimisation criteria are shown in the following tables (from **Table 8.12** to **Table 8.15**). As in the case of two junction arterial application, the multi-criteria optimisation showed, in terms of performance indicators, the trade-off effect between the two criteria considered in the optimisation; in fact, any solution might be expected to be not dominant with respect to both criteria at the same time.

Table 8.12 - Multi-criteria optimisation based on Total delay minimisation and Capacity Factor maximisation at junction 1 (Pop size = 20; Mutation rate = 0.8; Crossover rate = 0.0001). UNISA-GA&HC

Stream	Total Delay [PCU-hr/hr]	Capacity Factor	Effective green
A	1.27	1.66	30
B	2.35	2.15	55

Table 8.13 - Multi-criteria optimisation based on Total delay minimisation and Capacity Factor maximisation at junction 2 (Pop size = 20; Mutation rate = 0.8; Crossover rate = 0.0001). UNISA-GA&HC

Stream	Total Delay [PCU-hr/hr]	Capacity Factor	Effective green
A	1.75	2.32	39
B	1.91	2.47	46

Table 8.14 - Multi-criteria optimisation based on Total delay minimisation and Capacity Factor maximisation at junction 3 (Pop size = 20; Mutation rate = 0.8; Crossover rate = 0.0001). UNISA-GA&HC

Stream	Total Delay [PCU-hr/hr]	Capacity Factor	Effective Green
A	1.49	1.88	32
B	2.30	2.68	53

Table 8.15 - Multi-criteria optimisation based on Total delay minimisation and Capacity Factor maximisation at junction 4 (Pop size = 20; Mutation rate = 0.8; Crossover rate = 0.0001). UNISA-GA&HC

Stream	Total Delay [PCU-hr/hr]	Capacity Factor	Effective Green
A	1.53	2.38	39
B	1.79	2.44	46

Starting from the timings obtained by the application of the GAs' implementations, once five layouts (varying the connecting links length) were fixed (see **Table 8.16**), the network total delay, the degree of saturation and the link offsets between the signal plans of the interacting junctions were carried out through HC algorithm and modelling network flow propagation by CT&PDM. The results were then compared with those obtained by OSCADY, to achieve single junction timings, and TRANSYT-HC (PDM/CTM) to obtain optimal link offsets (see from **Table 8.17** to **Table 8.19**).

Table 8.16 - Lengths combination and ID

L1 [m]	L2 [m]	L3 [m]	L4[m]	Lengths combination ID
500	200	600	200	I
400	190	500	180	II
300	170	350	160	III

As highlighted in **Table 8.17**, even though comparable results in the computation of network total delay were identified for lengths combination I and II, there was a significant discrepancy for lengths combination III, among the results carried out in OSCADY&TRANSYT-HC (PDM), OSCADY&TRANSYT-HC (CTM) and UNISA-GA&HC (CT&PDM). De facto, as the distances decreased, the Network Total Delay did not change significantly in OSCADY&TRANSYT-HC (PDM), while it increased considerably in UNISA-GA&HC (CT&PDM) and OSCADY&TRANSYT-HC (CTM).

Such a result may be related to the higher reliability of CT&PDM (in UNISA-GA&HC) and CTM (in OSCADY&TRANSYT-HC) in modelling queues in terms of its spatial extensions. Therefore, when short links were considered and traffic was more likely to be restricted by downstream traffic and traffic signals, CT&PDM and CTM could be considered as more reliable compared to PDM which employed vertical queuing.

Table 8.17 - Numerical results for OSCADY&TRANSYT-HC (PDM), OSCADY&TRANSYT-HC (CTM) and UNISA-GA&HC (CT&PDM) w.r.t. length variations

Lengths combination [ID]]	OSCADY&TRANSYT HC (PDM)		OSCADY&TRANSYT HC (CTM)		UNISA GA&HC (CT&PDM)	
	TD [PCU- hr/hr]	DOS [%]	TD [PCU-hr/hr]	DOS [%]	TD [PCU-hr/hr]	DOS [%]
I	10.14		13.11		8.48	
II	12.53	65	12.81	72	12.83	60
III	7.53		22.41		19.34	

Table 8.18 - Optimised values of the offsets achieved by OSCADY&TRANSYT-HC - (PDM) and by UNISA-GA&HC (CT&PDM) w.r.t. length variations

Lengths comb	OSCADY&TRANSYT-HC (PDM)				UNISA-GA&HC (CT&PDM)			
	Φ 1-2 [sec]	Φ 2-3 [sec]	Φ 3-4 [sec]	Φ 4-1 [sec]	Φ 1-2 [sec]	Φ 2-3 [sec]	Φ 3-4 [sec]	Φ 4-1 [sec]
I	77	24	0	79	54	31	58	37
II	80	20	3	77	49	20	54	57
III	35	26	45	74	52	3	46	79

Table 8.19 - Optimised values of the offsets achieved by OSCADY&TRANSYT-HC - (CTM) and by UNISA-GA&HC (CT&PDM) w.r.t. length variations

Lengths comb	OSCADY&TRANSYT-HC (CTM)				UNISA-GA&HC (CT&PDM)			
	Φ 1-2 [sec]	Φ 2-3 [sec]	Φ 3-4 [sec]	Φ 4-1 [sec]	Φ 1-2 [sec]	Φ 2-3 [sec]	Φ 3-4 [sec]	Φ 4-1 [sec]
I	19	51	45	65	54	31	58	37
II	8	32	50	0	49	20	54	57
III	71	33	63	13	52	3	46	79

Furthermore, as already shown in the case of two junction arterial, due to the multi-criteria optimisation, the capacity factor obtained in UNISA-GA&HC (CT&PDM) was better than those obtained by the benchmark strategies.

With respect to the layout of ID I, cyclic flow profiles (CFPs) obtained through UNISA-GA&HC (CT&PDM) are shown below. In particular **Fig.8.6** refers to results obtained at upstream of link 1-2, **Fig.8.7** refers to results obtained in a middle section of the link 1-2, and **Fig.8.8** at downstream of link 1-2.

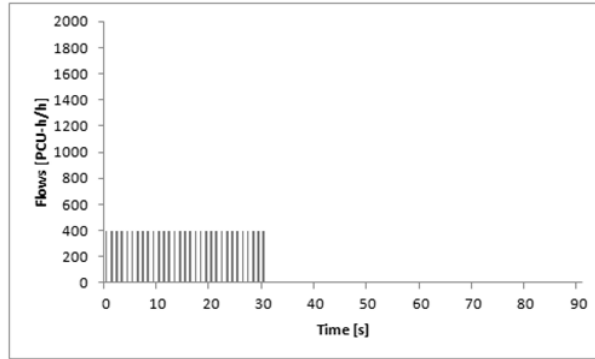


Fig.8.6 - UNISA-GA&HC (CT&PDM) CFP at the beginning of link 1-2 (signalised junction 1)

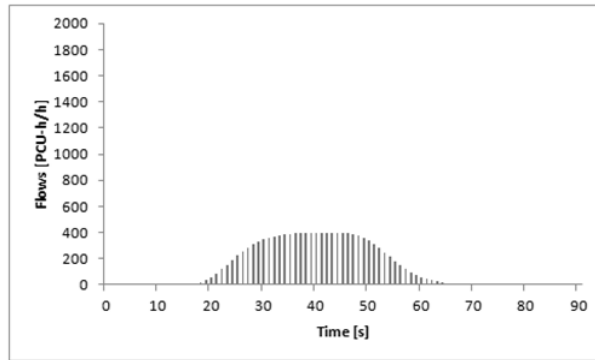


Fig.8.7 - UNISA-GA&HC (CT&PDM) CFP in the middle section of link 1-2

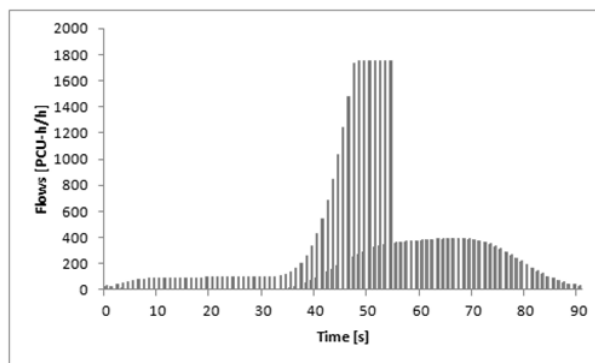


Fig.8.8 - UNISA-GA&HC (CT&PDM) CFP at the end of link 1-2 (signalised junction 2)

b. Twostep optimisation (B.2)

The network parameters (i.e. the stage matrix for each junction, the input flows and the three combinations of links length) were fixed as equal to those adopted in previous implementations. The network total delay, the degree of saturation, the link offsets between the signal plans of the interacting junctions and the green timings (see from **Table 8.20** to **Table 8.24**) were carried out by adopting UNISA-SA (CT&PDM), thus the green timings and link offsets are simultaneously optimised through a SA algorithm and modelling network flow propagation by CT&PDM. The results were then compared with those obtained by TRANSYT-SA (PDM/CTM)

As it was previously stated (in three step optimisation (A)), the adopted approaches provided different queuing models which directly affected the obtained results. As shown in **Table 8.20**, in the case of higher distances (length combinations I and II), the total delay was overestimated (12.04 PCU-hr/hr and 14.25 PCU-hr/hr) in TRANSYT-SA (CTM) whereas lower values were produced by TRANSYT-SA (PDM) (9.04 PCU-hr/hr and 7.21 PCU-hr/hr) and UNISA-SA (CT&PDM) (6.71 PCU-hr/hr and 9.41 PCU-hr/hr); at lower distances (length combination III), the value computed by TRANSYT-SA (PDM) (6.85 PCU-hr/hr) seemed to be not sensitive to the reduction of distances while similar results (15.38 PCU-hr/hr vs. 14.16 PCU-hr/hr) were shown in TRANSYT-SA (CTM) and UNISA-SA (CT&PDM) since they took account of the spatial extension of queues. These results confirmed the suitability of the proposed model for simulating both dispersion and queue blockage phenomena.

Table 8.20 - Numerical results for TRANSYT-SA (PDM), TRANSYT-SA (CTM) and UNISA-SA (CT&PDM) w.r.t. length variations

Lengths combination [id]	TRANSYT-SA (PDM)		TRANSYT-SA (CTM)		UNISA-SA (CT&PDM)	
	TD [PCU-hr/hr]	DOS [%]	TD [PCU-hr/hr]	DOS [%]	TD [PCU-hr/hr]	DOS [%]
I	9.04	53	12.04	69.55	6.71	49
II	7.21	49	14.25	64.27	9.41	62
III	6.85	50	15.38	74.13	14.16	66

Table 8.21 - Optimised values of the offsets achieved by TRANSYT-SA (PDM) and by UNISA-SA (CT&PDM) w.r.t. length variations

Lengths combination [id]	TRANSYT-SA (PDM)				UNISA-SA (CT&PDM)			
	Φ_{1-2} [sec]	Φ_{2-3} [sec]	Φ_{3-4} [sec]	Φ_{4-1} [sec]	Φ_{1-2} [sec]	Φ_{2-3} [sec]	Φ_{3-4} [sec]	Φ_{4-1} [sec]
I	28	21	69	62	54	13	72	41
II	43	20	53	64	49	22	44	65
III	45	19	51	65	63	3	39	75

Table 8.22 - Optimised values of the offsets achieved by TRANSYT-SA (CTM) and by UNISA-SA (CT&PDM) w.r.t. length variations

Lengths combination [id]	TRANSYT-SA (CTM)				UNISA-SA (CT&PDM)			
	Φ 1-2 [sec]	Φ 2-3 [sec]	Φ 3-4 [sec]	Φ 4-1 [sec]	Φ 1-2 [sec]	Φ 2-3 [sec]	Φ 3-4 [sec]	Φ 4-1 [sec]
I	18	14	23	35	54	13	72	41
II	38	34	55	53	49	22	44	65
III	56	12	44	68	63	3	39	75

Table 8.23 - Green timings achieved by TRANSYT-SA (PDM) and by UNISA-SA (CT&PDM) w.r.t. length variations

Lengths combination [id]	TRANSYT-SA (PDM)				UNISA-SA (CT&PDM)			
	g_{1A}^* [sec]	g_{2A} [sec]	g_{3A} [sec]	g_{4A} [sec]	g_{1A}^* [sec]	g_{2A} [sec]	g_{3A} [sec]	g_{4A} [sec]
I	43	33	58	38	37	44	42	45
II	40	41	43	42	33	38	44	33
III	39	37	42	40	38	36	41	31

*It refers to effective green duration of junction i and stage j (i.e. g_{ij})

Table 8.24 - Green timings achieved by TRANSYT-SA (CTM) and by UNISA-SA (CT&PDM) w.r.t. length variations

Lengths combination [id]	TRANSYT-SA (CTM)				UNISA-SA (CT&PDM)			
	g_{1A}^* [sec]	g_{2A} [sec]	g_{3A} [sec]	g_{4A} [sec]	g_{1A}^* [sec]	g_{2A} [sec]	g_{3A} [sec]	g_{4A} [sec]
I	44	56	67	35	37	44	42	45
II	51	51	68	65	33	38	44	33
III	46	54	63	26	38	36	41	31

*It refers to effective green duration of junction i and stage j (i.e. g_{ij})

8.4 Conclusions and research perspectives

This paper focuses on the Network Signal Setting Design. The main purposes of the paper are

- the green timings and offsets optimisation (by embedded procedure) and the stage sequence optimisation (by explicit enumerative approach);
- the application of some meta-heuristic algorithms for the solution of the mono-criterion/multi-criteria optimisation;
- the traffic flow modelling by a combined cell transmission and platoon dispersion model.

An overview of all optimisation strategies is shown in the following **Table 8.25** (repeating **Table 8.4** and

Table 8.5 for the reader's convenience) in which there are the strategies herein applied to two different layouts (i.e. two junction arterial vs. four junction network). In particular, in bold are the strategies (i.e. three step optimisation (**A**)) applied to the two junction arterial, and both strategy (i.e. two step (**B.2**) and three step optimisation (**A**)) applied to the four junction network.

The obtained results have been compared with those obtained by benchmark software (i.e. OSCADY PRO[®] and TRANSYT14[®] - TRL). The proposed strategies may be considered effective with respect to the optimisation methods and the multi-criteria optimisation should be considered a suitable approach.

Table 8.25 - Overview of two step (**B.2**) and three step strategy (**A**) w.r.t. Optimisation methods/objective functions including Algorithms

Strategy	layout	Variables		Variables		Variables	
		GT	Offsets	GT	Offsets	GT	Offsets
		Optimisation method		Objective functions		Algorithms	
B.2	Two junction arterial	Mono		TD		HC	
		Multi		TD & o.f.*		GAs	
	Four junction network	Mono		TD		SA	
		Mono		TD & o.f.*		GAs	
A	Two junction arterial	Mono	Mono	TD / CF		HC	HC
		Multi**	Mono	TD & CF		GAs	HC
	Four junction network	Mono	Mono	TD / CF		HC	HC
		Multi**	Mono	TD & CF		GAs	HC

*o.f.: It represents a generic objective function

Furthermore, the adopted traffic flow model (CT&PDM) aims to provide enhancements for effective traffic modelling among interacting junctions in which

both dispersion and queue blockage phenomena are represented, thus, it combines the skills of CTM and PDM.

The “Time-distance diagrams” in

Fig.8.9 are adopted to show the optimal green timings and offsets achieved through the implementation of UNISA-SA (CT&PDM). For brevity’s sake results refer only to the case highlighted in grey in the above **Table 8.25** and for distances $L_1=500$ m, $L_2=200$ m, $L_3=600$ m and $L_4=200$ m. The graphs provide a representation of the progression of traffic flow from stop-line to stop-line along a specified path (in this case from junction 1 to junction 4) by plotting traffic flow over time and distance. The progression bands are shown through a chromatic scale which indicates the consistency of the flows (considered as veh/clock tick. One clock tick is 1 second) for each cell adopted for the partition of the link length. According to space discretisation of CT&PDM, the cell length must be at least the maximum distance travelled by a vehicle in one clock tick that is the product of the free flow speed and the time measured in clock tick (in our case the free flow speed is 40km/h, the clock tick is 1/3600 h, thus the cell length is around 1/90 km). The number of cells for each link was computed once the cell length and the length of each stretch were known: $L_1= 5 \times 10^{-1}$ km, thus the number of cells is 45; $L_2= 2 \times 10^{-1}$ km, thus the number of cells is 18; $L_3= 6 \times 10^{-1}$ km, thus the number of cells is 54; $L_4= 2 \times 10^{-1}$ km, thus the number of cells is 18.

As regards

Fig.8.9, It is worthy of interest to note that for the links with higher lengths (1-2 of length L_1 and 3-4 of length L_3), the platoon dispersion generates a non-uniform arrival profile, differently from the links 2-3 of length L_2 and 4-1 of length L_4 where the low distances among the junctions prevent the phenomenon of dispersion, inducing a uniform arrival profile.

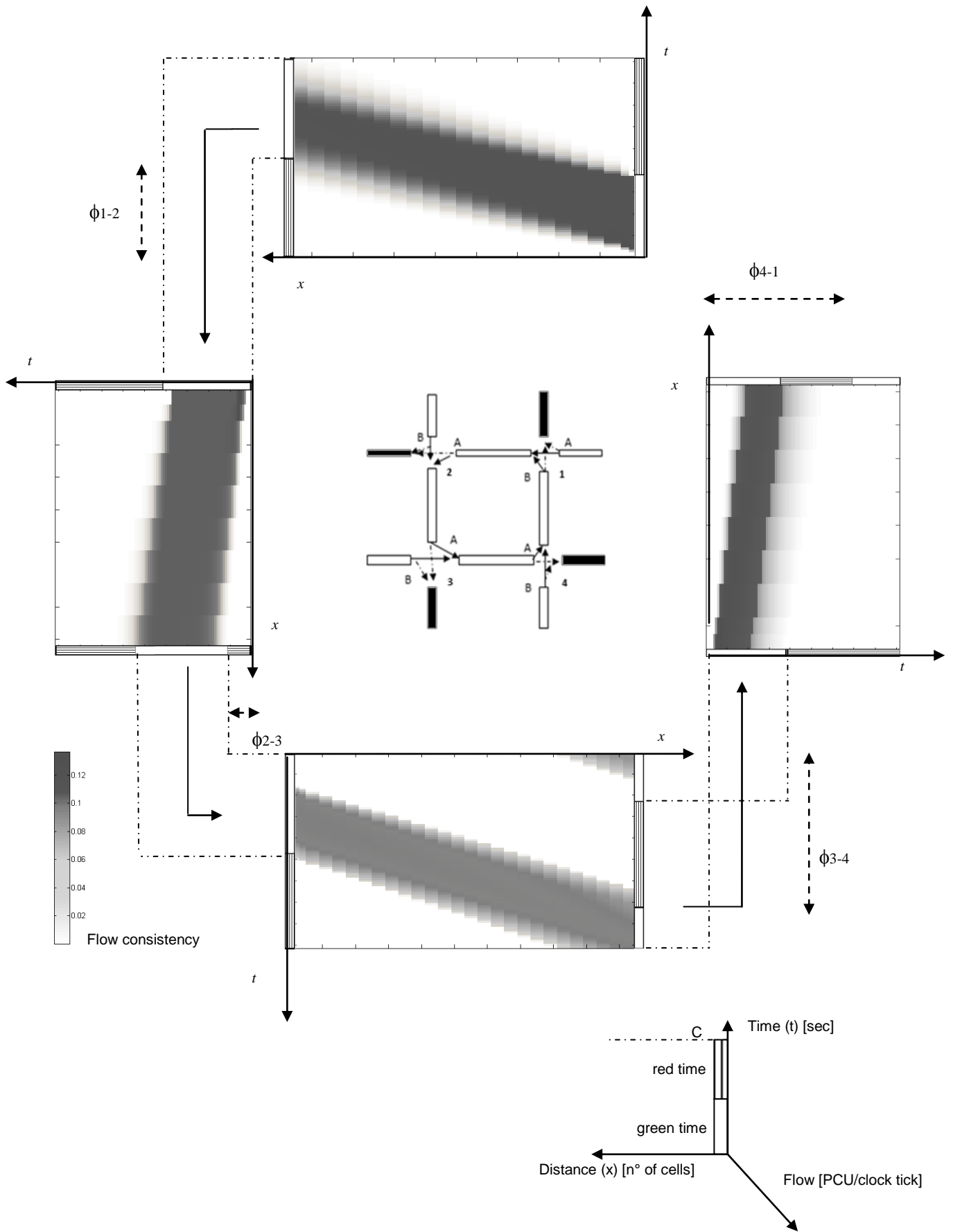


Fig.8.9 - Time Distance Diagram in UNISA-SA (CT&PDM) (Id I, L1=500 m, L2=200 m, L3=600 m and L1=200 m) all over the network layout

The final comparison between the proposed methodology and the benchmarks was carried out in terms of running times. In **Table 8.26** it is shown that no meaningful differences are observed in the case of the simpler layout (i.e. two junction arterial) and when the optimisation algorithm can be easily handled (i.e. HC algorithm, in the case of three step optimisation (A)). Although, some differences can be identified in two step optimisation (B.2), where a more complex algorithm (i.e. SA) is applied to achieve optimal results at a network level; in addition, in this latter case, an increasing level of interaction among decision variables and traffic flows could arise and may strongly affect the optimisation running time. In more detail, the highest value (185 s) is observed in TRANSYT-SA (CTM) and the lowest (74 s) is observed in TRANSYT-SA (PDM); notwithstanding that, the application of PDM (adopted by TRANSYT14® - TRL) may not be reliable when blocking phenomena occur (thus the lowest running time is imputed to the simpler adopted approach). Finally, the UNISA-SA (CT&PDM) running time is 95 s whereas that of TRANSYT-SA (CTM) is 185 s, thus, the application of the proposed approach is justified not only in terms of effectiveness but also in terms of efficiency.

Table 8.26 - Overview of computational time w.r.t. strategy and layout configurations

Strategy	Two junction arterial		Four junction network	
	Id	Running time [s]	Id	Running time [s]
B.2	-	-	UNISA-SA (CT&PDM)	95
			TRANSYT-SA (PDM)	74
			TRANSYT-SA (CTM)	185
A	UNISA-GA&HC (CT&PDM)	8	UNISA-GA&HC (CT&PDM)	14
	OSCADY&TRANSYT-HC (PDM)	8	OSCADY&TRANSYT-HC (PDM)	15
	OSCADY&TRANSYT-HC (CTM)	11	OSCADY&TRANSYT-HC (CTM)	18

In future papers other traffic flow models will be investigated, in particular more explicitly addressing the within day dynamics.

Several future research directions seem worthy of interest. The first one is related to multi-criteria network optimisation, in particular, in terms of optimisation criteria; in fact, even though in multi-criteria optimisation for a single junction, total delay (to be minimised) may be coupled with capacity factor (to be maximised), at network level, the capacity factor is not clearly defined. In fact, it may be intended as the lowest (to be maximised) achieved among all the junctions or as the area throughout (to be maximised). In this last case, the network multi-criteria optimisation may be based on a wide area criterion.

Second research direction may be identified within the one step optimisation in which the stage sequence (scheduling), the green timings and the offsets are computed at the same time (including multi-criteria optimisation).

Finally, addressing the network signal setting design as a part of transportation supply design (with equilibrium assignment) could be worthy of interest.

Acknowledgments

Authors wish to thank anonymous reviewers for their helpful comments. This research has been partially supported by the University of Salerno, Italy, EU under local grant n. ORSA132945 financial year 2013, Campania Region PON- Ricerca e Competitività 2000-2013 grant n. 04a2_G “SMART TUNNEL”, and PhD program on transportation (Ph.D. School in Environmental Engineering).

References

- Baker, J.E. (1985) Adaptive selection methods for genetic algorithms, Proceedings of the 3rd International Conference on Genetic Algorithms and Applications, In: Grefenstette, J.J. (ed.), New Jersey, Lawrence Erlbaum: Hillsdale, pp 100-111.
- Binning, J. C., Crabtree, M. R., & Burtenshaw, G. L. (2010) Transyt 14 user guide. Transport Road Laboratory Report nr AG48. APPLICATION GUIDE 65 (Issue F).
- Burrow, I.J. (1987) OSCADY : a computer program to model capacities, queues and delays at isolated traffic signal junctions: Traffic Management Division, Traffic Group, Transport and Road Research Laboratory, Crowthorne, Berkshire.
- Ceylan H. (2006) Developing Combined Genetic Algorithm-Hill-Climbing Optimization Method for Area Traffic Control. *Journal of Transportation Engineering*, 132(8), pp 663-671.
- Chow, A. H., & Lo, H. K. (2007) Sensitivity analysis of signal control with physical queuing: Delay derivatives and an application. *Transportation Research Part B: Methodological*, 41(4), pp 462-477.
- Daganzo, C.F. (1994) The cell-transmission model. Part 2: Network traffic, University of California, Berkeley, California.
- Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. (2002) A Fast and Elitist Multi-Objective Genetic Algorithm : NSGA II. *IEEE Trans. On Evolutionary Computation*, vol.6, pp 182-197.
- Hadi, M. A., & Wallace, C. E. (1993) Hybrid genetic algorithm to optimize signal phasing and timing *Transpn.Res.Rec.*, 1421, pp 94-103.
- Holland, J.H. (1975) *Adaptation in Natural and Artificial System*. The University of Michigan Press.
- Improta, G. & Cantarella, G. E. (1984). Control system design for an individual signalized junction. *Transportation Research Part B: Methodological*, 18(2), pp 147-167.
- Jabari, S. E., & Liu, H. X. (2012). A stochastic model of traffic flow: Theoretical foundations. *Transportation Research Part B: Methodological*, 46(1), pp 156-174.
- Lertworawanich, P., Kuwahara, M., Miska, M., (2011) A new Multiobjective Signal Optimization for Oversaturated Networks. *IEEE Transactions on Intelligent Transportation Systems*, vol.12, no.4.
- Li, Z. (2011). Modelling Arterial Signal Optimization with Enhanced Cell Transmission Formulations. *ASCE Journal of Transportation Engineering*, 137, pp 445-454.
- Li, Z., Chang, G.L., (2010) Modelling arterial signal optimization with enhanced cell transmission formulations. *Transportation Research Part B*, pp 2-19.
- Lighthill, M. J., Whitham, G. B. (1955) On kinematic waves. II. A theory of traffic flow on long crowded roads, *Proc. Roy. Soc. London.*, pp 317-345.
- Lo, H. K., & Chow, A. H. (2004) Control strategies for oversaturated traffic. *Journal of Transportation Engineering*, 130(4), pp 466-478.
- Lo, H. K. (2001) A cell-based traffic control formulation: strategies and benefits of dynamic timing plans. *Transportation Science*, 35(2), pp 148-164.
- Lo, H.K., (1999). A novel traffic signal control formulation. *Transportation Research 33A* (6), pp 433-448.
- Park, B., B., Messer, J., C., Urbanik II, T., (2000) Enhanced Genetic Algorithm for Signal-Timing Optimization of Oversaturated Intersections. *Transportation Research record 1727*, no.00.1661.
- Putha, R., Quadrifoglio, L., & Zechman, E. (2012) Comparing ant colony optimization and genetic algorithm approaches for solving traffic signal coordination under oversaturation conditions. *Computer-Aided Civil and Infrastructure Engineering*, 27(1), pp 14-28.

- Richards, P.I., (1956) Shockwaves on the highway, *Operations Research*, Vol. 4, pp 42-51.
- Robertson, D.I., (1969) TRANSYT: a traffic network study tool. RRL Report LR 253, Road Research Laboratory, England.
- Sumalee, A., Zhong, R. X., Pan, T. L., & Szeto, W. Y. (2011) Stochastic cell transmission model (SCTM): a stochastic dynamic traffic model for traffic state surveillance and assignment. *Transportation Research Part B: Methodological*, 45(3), pp 507-533.
- Szeto, W. Y. (2008) Enhanced lagged cell-transmission model for dynamic traffic assignment. *Transportation Research Record: Journal of the Transportation Research Board*, 2085(1), pp 76-85.

Appendix A

A1. Genetic Algorithms

GAs seek the optimal solution by simulating the evolution of a “population” of solutions (individuals), by mimicking the basic principle of bacteria evolution. Each solution is described by a vector of decision variables called a chromosome made up of genes. In our case, each gene is representative of the stage’s duration; the number of genes in each chromosome depends on the number of stages computed by the use of the relationship between approaches and stages. Optimisation was carried out through an iterative process which was representative of a reproductive cycle of the individuals (solutions) in the population (see **Fig.8.10**).

GAs are based on the genetic operators application such as crossover and mutation; while the former promotes the ability of the algorithm to explore the wider areas in order to search for solutions, the latter, without departing from the areas which are searching for solutions, introduces some (random) diversifications of the same type. Based on previous considerations the crossover can be called the explorative operator and the mutation can be called the exploiter operator.

For all these reasons, GAs are characterised by a higher effectiveness in approximating the global solutions in optimisation problems as the iterative process of the algorithm allows the GAs to explore large areas of optimal solutions.

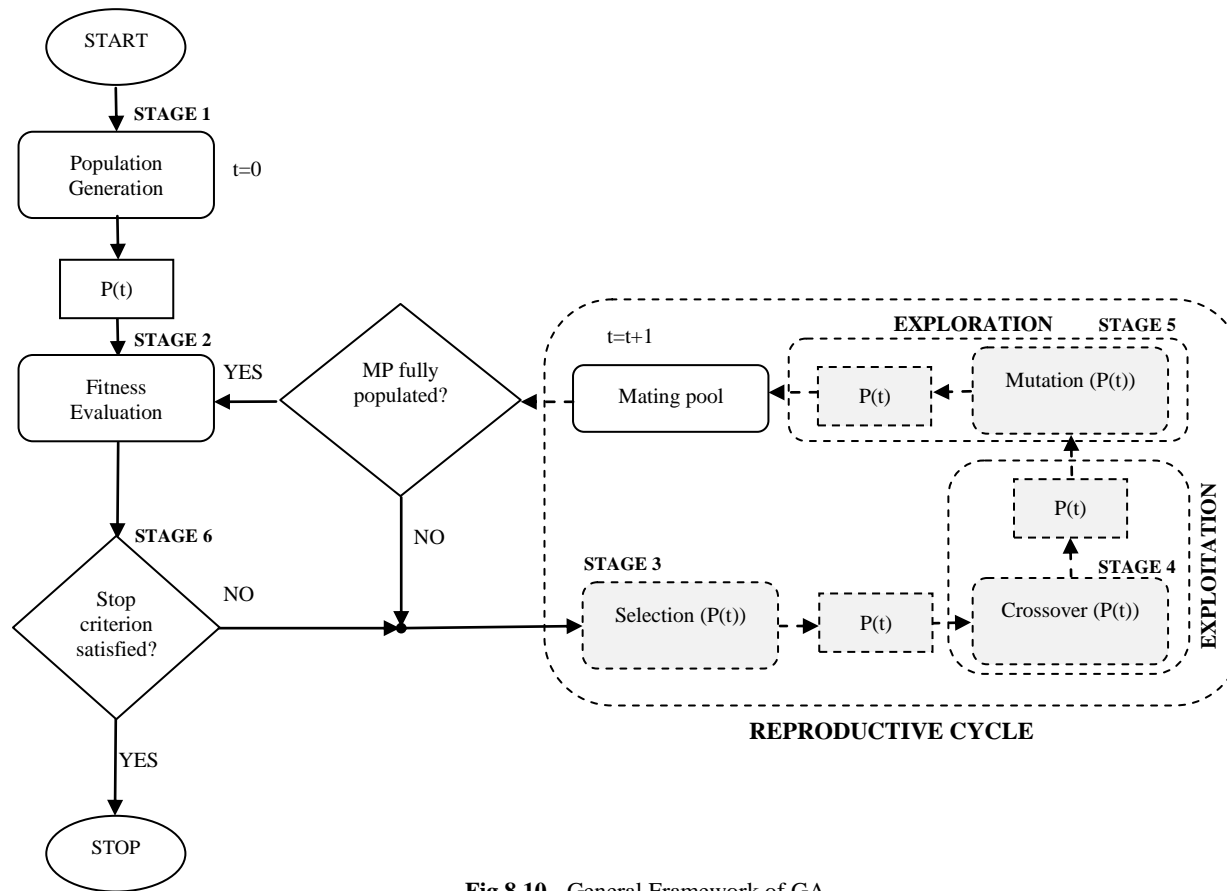


Fig.8.10 - General Framework of GA

Stage [1] Population generation and Stage [2] Fitness Evaluation

At the beginning ($t=0$) the population, $P(t)$, (in terms of each chromosome in the population) is randomly generated. At the successive step, for each solution, i , the fitness functions, ff_i , are calculated, so that the better the fitness function value of a solution, the greater its reproduction probability. In fact, the fitness function is strictly related to the probability, p_i , of being selected to be a parent; the probability is computed by normalising the fitness function, ff_i , of each chromosome, i , with regard to the fitness functions of all chromosomes i.e. $p_i = ff_i / \sum_j ff_j$.

The procedure of probability (of being selected to be parent) computation is representative of the analogy between natural selection and the reproductive cycle (Holland, 1975).

Some more considerations need to be made in terms of fitness function estimation depending on the optimisation problem (e.g. mono-criterion optimisation vs. multi-criteria optimisation).

Stage [3] Selection

Based on the p_i , each solution is submitted to the selection procedure. Different approaches can be adopted for the selection and in our case the roulette wheel was adopted. With regard to each chromosome the p_i is computed, these probabilities are used by “composing” a roulette. The selection operator, generates a random value, depending on the membership range on the roulette wheel of each number, the corresponding solution is selected to be a parent for the successive generation.

Each time a solution is selected, a copy of the same is made and included in the mating pool (MP) until the same is full, i.e. when the mating pool size is equal to the size of the population.

Stage [4] Crossover and Stage [5] Mutation

After reproduction, each chromosome may be modified by two further genetic operators. This process refers to the crossover, whereby two new individuals are generated by mating with two already existing individuals, and through the mutation, an individual is randomly changed. By applying the two operators several times, a new population of solutions is obtained. This iterative procedure is repeated until some conditions (e.g. number of iterations or of improvement of the best solution) are satisfied.

The main parameters to fully specify GAs are the population size, the crossover probability (or rate, PC), and the mutation probability (or rate, PM). In terms of the crossover rate and the mutation rate, it can be observed that in order to balance the exploration and the exploitation effects introduced by two operators, the value rate of each one is usually heuristically chosen in order to respect the order relation: $PM < PC$.

Stage [6] Stop criterion

The iterative procedure is stopped when some criteria are satisfied, such as the fixed number of generations is reached or a solution is found that satisfies optimisation criteria where usually a threshold is introduced to evaluate the difference between the current solution and the solution reached in a previous generation. Convergence is reached when next generations do not lead to the further improvement of the fitness function.

GAs can be applied both to mono-criterion and multi-criteria optimisation problems. In terms of the algorithm procedure, this has an effect on computing the fitness function. In the first case, the general description is sufficient to be able to understand the procedure of the algorithm whereas in the case of the multi-criteria, further considerations are needed.

In particular, GAs for multi-criteria optimisation were implemented following the Non dominated Sorting Genetic Algorithm II (NSGA-II; Deb and Pratap, 2002). In accordance with this method, the selection (of being parents) of the chromosomes was made on the basis of the ranking of the solutions. Before starting the selection step, for each solution a value of rank was associated to the number of times in which the considered solution dominates the others. Using the obtained values, the rank based fitness assignment could be applied, in particular, to the fitness function, ff_i , for a given chromosome i , calculated by the linear ranking (Baker, 1985) approach as follows:

$$ff_i = 2 - SP + 2 (SP - 1) (rank_i - 1) / N$$

where:

SP is the selective pressure fixed to 1.5,

N is the population size,

$rank_i$, corresponds to the solution ranking and it was computed from the chromosome dominance hierarchy. Finally, an additional criterion was introduced for selecting among solutions with the same ranks. Each solution was attached to a value of crowding distance given by the Euclidean distance between the vector of the fitness functions of the solution and the vector of the best fitness function values, each one was defined as the best value among all solutions or in some cases the reference fitness function values were defined with regard to the considered criteria. If two or more solutions had the same rank value, selection at successive stages was based on the best value of the crowding distance.

All the defined multi-criteria problems were solved by applying NSGA-II.

A.2 Hill Climbing (HC) Algorithm

Hill Climbing is a neighbourhood-based meta-heuristic algorithm, without memory, which is deterministic in its basic version. The name originates from its ability to generate a succession of solutions by exploring the objective function

surface which, if plotted, could be thought of as a series of hills and valleys in a multiple-dimensional world.

In this method, starting from an initial solution, successive iterations in the neighbourhood are performed until the current solution cannot be improved further. The algorithm stops when a local minimum/maximum is reached.

Different stochastic variants of this method are proposed in an attempt to endow it with a diversification strategy. For instance, the method can be applied by starting from multiple initial solutions which are randomly generated (as in the case of Shot-Gun Hill Climbing), or by varying the structure of the surroundings during the iterations.

In this paper, a basic Hill Climbing algorithm was applied in order to minimise the Total Delay at the downstream stop line (as a result of connecting link flow simulation). In order to reduce the risk of being trapped in a poor local optimum, a list of both small and large incremental offset alterations was set up (such increments are listed as percentages of the cycle time). Thereby, low increments made it possible to find an approximate local minimum of the Total Delay whilst high increments avoided getting trapped in that minimum.

A.3 Simulated Annealing (SA) Algorithm

The Simulated Annealing algorithm is a neighbourhood based meta-heuristic, which is inspired by the statistical mechanics to find solutions for both discrete and continuous optimisation problems. In particular, it takes the cue from the metallurgical phenomenon of annealing, in which a solid is brought to melting and then is slowly cooled to crystallize in a perfect lattice. Metropolis, in 1953, proposed for the first time a method to calculate the distribution of a particulates system in equilibrium temperature using a method of computer simulation. In this method, assuming that the system is in a configuration q having energy $E(q)$, a new state r , having energy $E(r)$, is generated, moving a particle from its initial position. The new configuration is then compared with the older one. If $E(r) < E(q)$ the new state is accepted, if $E(r) > E(q)$ it is not rejected, but rather, it is accepted with a probability P_a equal to:

$$P_a = \exp(-(E(r) - E(q)) / KT)$$

where K is the Boltzmann constant and T is a control parameter, which by analogy with the original application is called temperature.

According to this method, there is, hence, a non-zero probability of reaching states of higher energy and so energy barriers that separate the global minima from the local minima can be climbed over. Note that the exponential function expresses the ratio between the probability of being in the leader configuration and the probability of being in q . Kirkpatrick used the scheme of SA for combinatorial optimisation problems. To do this, he replaced the energy with a cost function and the states of a physical particles system with the solutions of a minimization problem. The implementation of the SA algorithm requires the specification of

some elements such as: a generator of random changes in solution; a fixed number of iterations before decreasing the temperature (L_k); an annealing schedule, i.e. an initial temperature and the rules for lowering it as the search progresses. Several cooling schedules are covered, including exponential, linear and temperature cycling. The simplest and most common temperature decrement rule is the Kirkpatrick geometric cooling scheme:

$$T_k = \alpha T_{k-1}$$

where α is a constant close to, but smaller than 1, usually assumed equal to 0.95. In this paper the SA technique working on a single objective combinatorial problem, i.e. the minimisation of network total delay, was applied. In following, the algorithm stages are described.

Stage [1]

Let q_0 be an initial configuration which returns an Energy state (solution) $E(q_0)$, let T_0 be the initial temperature and T_f be the final temperature.

Stage [2]

For each decreasing value of T , T_k , the following steps were considered:

- By applying a random alteration of the current problem variables (i.e. green timings, green timings and offsets, Φ , among junctions) a feasible (energy state) new solution $E(r)$ is generated. Such a solution is then compared with the best one archived $E(q)$ (i.e. current solution):

$$\Delta E = E(r) - E(q)$$

- If $\Delta E \leq 0$, the new solution is accepted and so is archived as the new current solution, else, the new solution is accepted with a probability **Errore. L'origine riferimento non è stata trovata.** and substitutes the current solution if such a criterion is satisfied;
- If the thermal equilibrium is not achieved i.e. if the number of iterations at current temperature is less than L_k the algorithm goes to step II a).

Stage [3]

If $T_k > T_f$, the temperature is further decreased and the algorithm goes to stage 2.

9. Network Signal Setting Design with Stage Sequence Optimisation

Memoli, S., Cantarella, G. E., Di Pace, R., de Luca, S., (2016). Network Signal Setting Design with Stage Sequence Optimisation. Transportation Research Part B, under revision.

Abstract

This paper first provides a general framework of one-step methods for Network Signal Setting Design (NSSD); these methods including green scheduling, green timing and coordination lead to the so-called scheduled synchronisation. In accordance with the literature, two main approaches could be adopted, the explicit and the implicit ones, furthermore, within the implicit approaches two methods might be applied: the stage and the approach based methods. Then this paper aims to develop a general stage-based one-step method (CENEO) for Network Signal Setting Design that optimises stage composition and sequence without explicit enumeration of all stage sequences. The proposed method is an extension of the synchronisation method and the traffic flow model (ENEO) proposed in Cantarella et al. (2015).

The results of the proposed method were compared with those from an explicit enumeration of all stage sequences for the network with a different number of junctions.

9.1 Introduction and contribution

In order to mitigate urban traffic congestion, several policies can be adopted which may be applied in the short or long time horizon. With regard to the short term policies, one of the most straightforward is control through traffic lights at a single junction or network level. The main goal of traffic control is avoiding having incompatible streams which have green⁶ at the same time. With respect to this aim, existing methodologies for Network Signal Setting Design (NSSD) can be grouped into two classes: Stage-based and Approach (Phase)-based.

Stage-based signal setting methods divide the cycle into stages, each one being a time interval during which some mutually compatible approaches have green.

⁶ In the following we will use the word 'green' to indicate the period during which the light is green as well as the length of such a period also called 'green time' (or 'green timing') whilst its ratio to the cycle length is also called 'green splits'.

Stage composition, say which approaches have green, and sequence, say their order, can be represented through the approach-stage incidence matrix, or stage matrix for short. Once the stage matrix is given for each junction, the cycle length, the greens and the offsets can be optimised (*synchronisation*) through some well-established methods. It should be stressed that these methods do not allow for stage matrix optimisation at network level, thus, the effects of stage composition and sequence on network performance are not appropriately considered. Two of the most commonly used codes are: TRANSYT14® (TRL, UK) (recently TRANSYT15® has been released) and TRANSYT-7F® (FHWA, USA). Both make it possible to compute the cycle length, the greens and the offsets by combining a traffic flow model and a signal setting optimiser. Both codes may be used for coordination (optimisation of offsets only, once greens are known) or synchronisation. TRANSYT14® generates several (but not all) significant stage sequences to be tested but the optimal solution is not endogenously generated, while TRANSYT-7F® is able to optimise the stage sequence for each single junction starting from the ring and barrier NEMA (i.e. National Electrical Manufacturers Association) phases. Still, these methods do not allow for stage matrix optimisation; moreover, the effects of stage composition and sequence on network performance are not analysed. Further studies, with respect to the stage sequence optimisation may be found in Park et al. (2000). They carried out a simulation framework made up of a mesoscopic flow simulator and Genetic Algorithm optimiser which showed that the developed simulator provided better results than those obtained using the software TRANSYT-7F®. Other investigations may be found also in Hadi and Wallace (1993; 1994) which studied the possibility of introducing a phase sequence optimisation capability to TRANSYT-7F® using Genetic algorithms and the Cauchy simulated annealing.

Approach-based (or Phase-based) methods address the signal setting as a periodic scheduling problem: the cycle length, and for each approach, the start and the end of the green are considered as decision variables, other variables or constraints are included in order to avoid incompatible approaches having green at the same time (see for instance Improta and Cantarella, 1987). If needed, the stage composition and sequence may easily be obtained from decision variables, thus, the stage matrix is an implicit result of the procedure. Commercial software codes following this methodology are available for single junction control only, for instance, Oscady Pro® (TRL, UK; Burrow, 1987). Once the green timing and scheduling have been carried out for each junction, offsets can be optimised (*coordination*) using the stage matrices and greens obtained from single junction optimisation or using the stage matrices only to optimise both green and offsets (*synchronisation*).

Both methodologies described above share a two-step optimisation structure: decisional variables are first grouped in two sets and then they are sequentially optimised. No one-step optimisation method for the simultaneous optimisation of greens, their schedule, and node offsets, the so-called *scheduled synchronisation*, is available to the authors' knowledge.

In this paper the Stage-based methodology is applied to specify one-step methods for NSSD. These methods can be used by explicitly considering the stage composition and sequence as decisions. Firstly, for each junction a set of candidate stages is defined and then the stage sequence can be optimised, as described in this paper. Resulting methods are simpler than those derived from approach-based methods, but cannot provide the optimal solution in the general cases.

Formally, existing approach-based methods for single junctions may easily be extended to specify one-step methods for NSSD, since the node offsets may be easily obtained from decision variables, say the start and the end of the green of each approach, and if needed, the stage composition and sequence as well. Nonetheless, from a practical point of view the resulting problem may be hard to solve since several equivalent local optima may exist; this condition may quite easily be dealt with for a single junction, but it is rather unclear how it can effectively be circumvented for a network (with loops). Thus, meta-heuristics, or other optimisation techniques, might perform rather poorly unless the features of the space of solution are further exploited.

This paper aims to develop a general stage-based one-step method (CENEO) for Network Signal Setting Design that does not need explicit enumeration of stage sequences, which is only practicable for very small networks. The proposed method is an extension of the synchronisation method and the traffic flow model (ENEO) proposed in Cantarella et al. (2015)⁷.

The paper is organised as follows: section 2 describes the methodological framework whereas within section 3 the optimisation problem is discussed and in particular the solution methods mainly focusing on the proposed stage based approach, CENEO; in section 4 the results of the numerical applications carried out through CENEO and ENEO are shown and compared; in section 5 the final conclusions are discussed.

9.2 Methodological Framework

This section describes the adopted methodological framework, discussing in more detail the variables, the constraints and the objective functions.

9.2.1 Stages and Sequences

A *stage* is a set of *approaches* that have green at the same time. For safe operations all the approaches in a stage must be mutually compatible, namely they may have green without any conflict - *compatibility requirement* -.

Let $n \geq 2$ be the number of candidate stages ($n = 1$ meaning that there is no need for traffic control). To avoid a number of candidate stages which is too great, usually it is also required that each stage be maximal - *completeness requirement* -, meaning that no further approach may be added to any of them without violating

⁷ For the sake of completeness the method proposed in Cantarella et al. (2015) was not yet named ENEO.

the compatibility requirement. All candidate stages satisfying both the compatibility and completeness requirements can easily be generated by Bron & Kerbosch's (1973) algorithm for finding all maximal cliques of a graph, in this case, the adjacency matrix of the graph is the (square symmetric) compatibility matrix with as many rows and columns as the number of approaches and '0/1' entry for each 'incompatible/compatible' pairs of approaches.

For each junction the stages must be put in a *sequence*. To this aim a sequence is called *feasible* if each approach belongs to at least one stage in the sequence - *feasibility requirement* -. Of course this requirement may be fulfilled easily if the set of candidate stages is defined so that each approach belongs to one stage at least. On the other hand, if all the compatible and maximal stages are considered such condition surely holds (in the last case each stage contains only one approach). According to this definition, a candidate stage that must be included in each sequence since it contains an approach not included in any other stage is called *compulsory*, otherwise it is called *optional*. Let n_c and $n_o = n - n_c$ be the number of compulsory and optional candidate stages, respectively.

If there is no optional stage, $n_o = 0$ and $n = n_c$, the number of feasible sequences is given by $n!$, say the number of permutations of the n stages. However, it is worth noting that any periodic rotation of a sequence, for instance such that the sequence (1, 2, 3) becomes (2, 3, 1), then (3,1,2), then (1,2,3,) again it does not affect optimal green and performance indicators, while optimal offsets change in an easily predictable way (see **Fig.9.1**).

Since given a sequence, $(n - 1)$ other equivalent sequences may be generated this way, all the sequences may be grouped into $(n! / n) = (n - 1)!$ equivalency classes, and it suffices to analyse one sequence in each class. Thus,

- if only two stages are available, two possible sequences can be built up {(1,2); (2,1)} which are equivalent, thus, only one equivalent class exists;
- if three stages are available two equivalent classes exist: {(1,2,3), (2,3,1); (3,1,2)} and {(3,2,1); (2,1,3); (1,3,2)}; either may be obtained from sequence (1,2) by positioning stage 3 after stage 1 or after stage 2;
- if four stages are available six equivalent classes exist: {(1,2,3,4); (2,3,4,1); (3,4,1,2); (4,1,2,3)}; {(1,3,2,4); (3,2,4,1); (2,4,1,3); (4,1,3,2)}; {(1,4,2,3); (4,2,3,1); (2,3,1,4); (3,1,4,2)}; {(1,4,3,2); (4,3,2,1); (3,2,1,4); (2,1,4,3)}; {(1,3,4,2); (3,4,2,1); (4,2,1,3); (2,1,3,4)}; {(1,2,4,3); (2,4,3,1); (4,3,1,2); (3,1,2,4)}; each may be obtained by changing the position of stage 4 with respect to sequences (1,2,3) or (3,2,1);
- and so on.

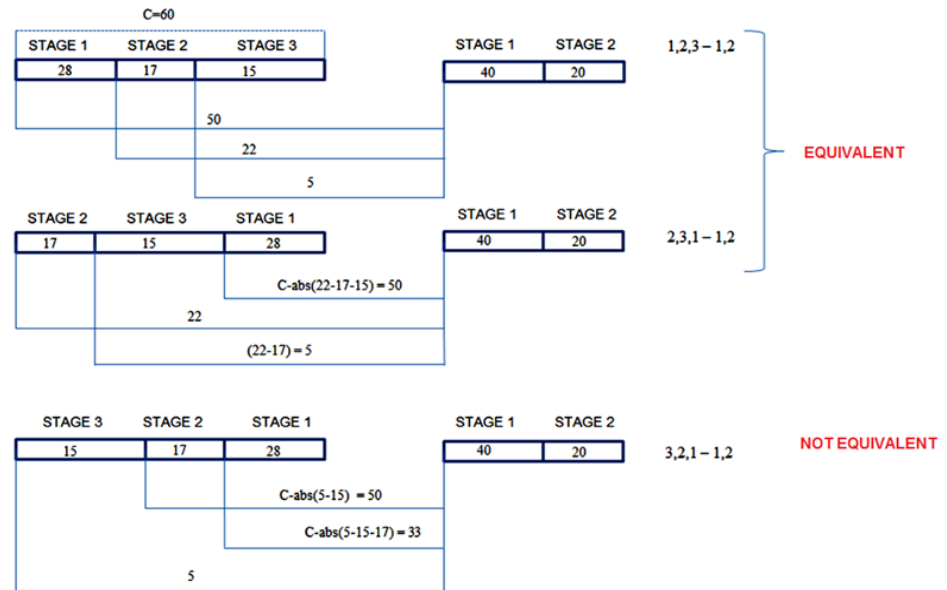
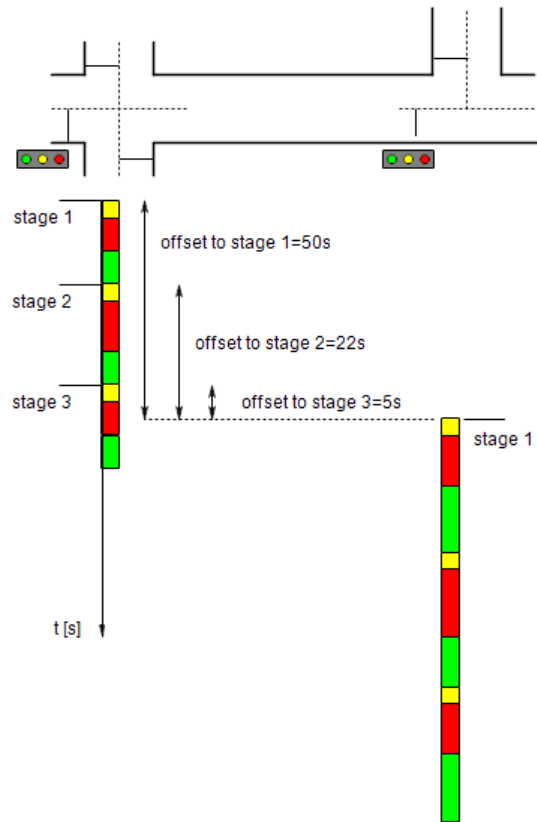


Fig.9.1 - equivalent vs. not equivalent sequences for a simple arterial

If there is at least one optional stage, $n_o > 0$ and $n > n_c$, the stages can be grouped into 2^{n_o} sub-sets, each including all the n_c compulsory stages and some (or none) of the n_o optional stages; stages belonging to any of such sets may be arranged in a number of feasible sequences equal to the factorial of its size minus one, as described above.

Thus,

- if only one compulsory, 1, and one optional, 2, stages are available it implies that stage 2 is a sub-set of stage 1, therefore, stage 2 violates the completeness requirement, even though two equivalent classes exist: $\{(1)\}$; $\{(1,2); (2,1)\}$;
- if two compulsory, 1 and 2, and one optional, 3, stages are available, three equivalent classes exist: $\{(1,2); (2,1)\}$; $\{(1,2,3), (2,3,1); (3,1,2)\}$ and $\{(3,2,1); (2,1,3); (1,3,2)\}$; each may be obtained from sequence (1,2) by not including stage 3 or by positioning it after stage 1 or after stage 2;
- if three compulsory, 1 and 2 and 3, and one optional, 4, stages are available, eight equivalent classes exist: $\{(1,2,3), (2,3,1); (3,1,2)\}$ and $\{(3,2,1); (2,1,3); (1,3,2)\}$; $\{(1,2,3,4); (2,3,4,1); (3,4,1,2); (4,1,2,3)\}$; $\{(1,3,2,4); (3,2,4,1); (2,4,1,3); (4,1,3,2)\}$; $\{(1,4,2,3); (4,2,3,1); (2,3,1,4); (3,1,4,2)\}$; $\{(1,4,3,2); (4,3,2,1); (3,2,1,4); (2,1,4,3)\}$; $\{(1,3,4,2); (3,4,2,1); (4,2,1,3); (2,1,3,4)\}$; $\{(1,2,4,3); (2,4,3,1); (4,3,1,2); (3,1,2,4)\}$; each may be obtained by not including stage 4 or by changing its position within the sequences (1,2,3);
- and so on.

Quite often it is also required that each approach has green in consecutive stages, if more than one - *consecutiveness requirement* -. This requirement is effective for sequences containing four or more stages.

The sequence of stages and their composition can be described by Δ , the approach-stage incidence matrix (or stage matrix for short), with entries $\delta_{kj}=1$ if approach k receives green during stage j and 0 otherwise.

9.2.2 Time and Flow variables and constraints

Assuming that the green scheduling is described by the stage matrix (i.e. the stage matrix composition and sequence), let

c be the cycle length, assumed known or as a decision variable (common to all junctions);

for each junction (not explicitly indicated)

t_j be the duration of stage j as a decision variable;

t_{ar} , be the so-called all red period at the end of each stage to allow the safe clearance of the junction, assumed known (and constant for simplicity's sake);
 Δ be the approach-stage incidence matrix (or stage matrix for short), with entries $\delta_{kj}=1$ if approach k receives green during stage j and 0 otherwise, assumed known;
 l_k be the lost time for approach k , assumed known;
 $g_k = \sum_j \delta_{kj} t_j - t_{ar} - l_k$ be the effective green for approach k ;
 $r_k = c - g_k$ be the effective red for approach k ;
 y_k be the arrival flow for approach k , assumed known;
 s_k be the saturation flow for approach k , assumed known;
 $(s_k \cdot g_k) / (c \cdot y_k)$ be the capacity factor for approach k ;

and for each junction in the network

ϕ_i be the node offset as the time shift between the start of the plan for the junction i and the start of the reference plan, say the plan of the junction number 1, $\phi_1=0$. Given such a reference value, all the other $m-1$ nodes, where m is the number of junctions in the network, are independent variables. As well known, compute total delay through a traffic flow model needs the link offset between each pair of adjacent junctions, $\phi_{ij} = -\phi_{ji}$. Let the junction network be represented by an undirected graph with 1 node for each junction and an edge for each pair of adjacent junctions (the actually traffic directions are irrelevant). According to this representation

- if such a network is loop less, all the $m-1$ link offsets are independent (as many as the independent node offsets) and may be used as decision variables; Arterials are a special case of such a network;
- on the other hand, if the network contains k independent loops, the number of independent link offsets will be equal to $m-k$; in this case it is better to use the $m-1$ independent node offsets as optimisation variables.

Some constraints were introduced in order to guarantee:

stage durations being non-negative

$$t_j \geq 0 \quad \forall j$$

effective green being non-negative

$$g_k \geq 0 \quad \forall k$$

this constraint is usually guaranteed by the non-negative stage duration, but for a too short cycle length with regard to the values of all-red period length and lost times, say

$$\sum_j \text{MAX}_k (\delta_{kj} l_k + t_{ar}) \geq c$$

consistency among the stage durations and the cycle length

$$\sum_j t_j = c$$

the minimum value of the effective green timing

$$g_k \geq g_{min} \quad \forall k$$

A further constraint was included in order to guarantee that the capacity factor must be greater than 1 (or any other value)

$$((s_k \cdot g_k) / (c \cdot y_k) - 1) \geq 0 \quad \forall k$$

Such a constraint may be added only after having checked that the maximum junction capacity factor for each approach k in the junction i is greater than 1, otherwise a solution may not exist whatever the objective function is.

Finally let assume

$$c \geq \phi_i \geq 0.$$

9.2.3 Objective Functions

In this paper the total delay TD was considered as objective function (even though other measure of performances could be introduced). Since generally in a junction network there can be interacting and non-interacting approaches, two cases should be distinguished.

Delay for non-interacting approaches

In this case it was computed using the two term Webster formula (Webster, 1958)⁸ as:

$$TD = \sum_k y_k \cdot (0.45 \cdot c \cdot (1 - g_k/c)^2 / (1 - y_k/s_k) + y_k \cdot 0.45 / (s_k \cdot g_k / c) \cdot ((g_k/c) \cdot (s_k/y_k) - 1))) \quad (\text{Eq.9.1})$$

Delay for interacting approaches

In this case delay computation needs an explicit *traffic flow model*. In this paper the CT&PDM (Cell Transmission and Platoon Dispersion Model), which is described in more detail in Cantarella et al. (2015), was adopted. For the computation of

⁸ Any other formula can be used as well.

delay for an interacting approach the cumulated input, $Cif_k^j(t)$, and output flows, $Cof_k^j(t)$, through the stop line of approach k of junction j, in the subsequent sub-intervals t were compared.

The Deterministic Total Delay (DTD_k^j) cumulated in the interval [0, T] for approach k of junction j was then given by the following expression:

$$DTD_k^j = \sum_{i=1..T/\Delta\tau} (Ciy_k^j(i \times \Delta\tau) - Coy_k^j(i \times \Delta\tau)) \Delta\tau^9 \quad (\text{Eq.9.2})$$

Thus delay experienced at an interacting approach is a function of the offsets between the timing plans. In fact, such a delay depends on the output flow in the downstream junction which is obtained by starting from the input flow in the upstream junction through the phenomenon of dispersion.

Let s_k^j be the saturation flow on approach k of junction j, the Stochastic and Oversaturation component of Total Delay $SOTD_k^j$ for approach k of junction j is computed using the following expression

$$SOTD_k^j = \{ [(y_k^j - s_k^j)^2 + (4y_k^j/T)]^{0.5} + (y_k^j - s_k^j) \} T/4 \quad (\text{Eq.9.3})$$

and considering the average of the values of the cyclic flow profile along the connecting link arriving at approach k of considered junction j, f_k^j as input flow.

Then

$$TD_k^j = DTD_k^j + SOTD_k^j \quad (\text{Eq.9.4})$$

Finally, the total network delay is given by

$$TD = \sum_j \sum_k TD_k^j \quad (\text{Eq.9.5})$$

where TD_k^j is computed through **Eq.9.1** in case of non-interacting junction and through **Eq.9.4** in case of interacting junctions.

⁹ $\Delta\tau$ is the length of the time interval.

9.3 Scheduled Synchronisation

In this section the proposed approach to the scheduled synchronisation problem is discussed. As already noted in the introduction, stage based methods can be used to specify the scheduled synchronisation by explicitly considering the stage composition and sequence as decisions. To solve such optimisation problem meta-heuristic algorithms are usually adopted such as Simulated Annealing (SA). As a matter of fact, such algorithms can effectively address even optimisation problems where objective function not expressed in a closed form, so that derivatives are not easily available, as it occurs for the scheduled synchronisation. The proposed solution approach and the implementation remarks for the scheduling synchronisation, called CENEO (CompleTE NEtwork Optimisation), are outlined below.

First, a set of candidate stages is defined for each junction, usually all the stages satisfying both the compatibility and completeness requirements introduced in **sub-section 9.2.1**, then the stage sequence can be optimised, together with greens and offsets, taking into account the feasibility requirement introduced in **sub-section 9.2.1**, as well as the constraints discussed in **sub-section 9.2.2**, minimising TD computed as described in **sub-section 9.2.3**.

As a matter of fact, as described in **sub-section 9.2.1**, any periodic rotation of a sequence, for instance such that the sequence (1,2,3) becomes (2,3,1), does not affect optimal greens and performance indicators, thus, for each junction i all the sequences are considered grouped into $(n_i! / n_i) = (n_i - 1)!$ equivalence classes, eventually reduced through the consecutiveness requirement (**sub-section 9.2.1**); therefore, only one sequence for each equivalence class is further analysed.

9.3.1 The Solution Algorithm

The proposed SA algorithm, also known as the Metropolis Scheme (Metropolis et al., 1953), starts with an initial solution and goes through a predetermined number of iterations to try to improve upon the objective function. In the scheduled-synchronisation, the algorithm starts with an initial combination of sequences, greens and offsets, randomly provided, and tries to improve the network total delay (or any other measure of performance). The whole framework is composed by an outer loop (external iterations) and an inner loop (internal iterations), as in following described (see **Fig.9.3**).

OUTER LOOP

From the initial or current solution (combination of sequences, greens and offsets), at each external iteration, new solution is randomly generated from the neighbourhood according to a generation scheme (see below for the inner loop). The measure of performance of the new solution is calculated and evaluated against the current one. If the new measure is better than the current one, it is accepted with a probability of one and the new solution is made the current one and the process repeats itself. However, if the new measure is worse than the current

one, then the new solution is still accepted, but with a probability $P(C) = \exp [- (C(n) - C(c)) / T]$ (1), where $C(n)$ and $C(c)$ are the measures of performance of the new and current solutions, respectively, and T is the current temperature of the system. In particular, given a real number r randomly drawn from a uniform distribution over the interval $[0,1]$, if $P(C)$ is smaller than the number r , then the new solution is rejected and the current solution remains in the process while the algorithm repeats itself. Note that as temperature decreases, so decreases the probability of accepting a “hill-climbing move.” If T is very large, then r is likely to be less than $P(C)$ and the new solution is almost always accepted. If T is small, or close to zero, then only the new solution characterised by having very small $\Delta C > 0$ have any realistic chance of being accepted. The temperature T is slowly decreased by a predetermined percentage until it reaches a certain predetermined value where the SA process terminates. The process can also terminate after a certain predetermined number of external iterations.

INNER LOOP

At each external iteration (temperature state), the new solution is carried out as an output of an inner loop which consists of testing the greens and offsets, randomly provided for the current external iteration, for a number (N) of combination of sequences (equalling the number of internal iterations of the inner loop) selected among those representing the $(n_i - 1)!$ equivalence classes of each junction. The size of the set of combinations to be tested is determined by the number of junctions in the considered network. For instance, in the case of a three junction network, with junction 1 having 3 stages and 2 equivalence classes, junction 2 having 4 stages and 6 equivalence classes, junction 3 having 3 stages and 2 equivalence classes, the size of such a set is given by $2 \times 6 \times 2 = 24$ combination of sequences.

The generation scheme is carried out on discrete binary and tertiary variables; in particular, if the considered junction contains a sequence of three stages, two equivalence classes exist thus a binary (b) variable 0/1 is adopted to randomly choose between the sequence (1,2,3), belonging the class $\{(1,2,3), (2,3,1); (3,1,2)\}$ and the sequence (1,3,2), belonging the class $\{(3,2,1); (2,1,3); (1,3,2)\}$. On the other hand, if the considered sequence contains four stages, a binary (b) variable 0/1 and a tertiary (t) variable 1/2/3 is considered and two random generations are applied (see **Fig.9.2**): first a binary random decision is taken as to select between the sequence (1,2,3,4), belonging the class $\{(1,2,3,4); (2,3,4,1); (3,4,1,2); (4,1,2,3)\}$, and the sequence (1,4,3,2), belonging the class $\{(1,4,3,2); (4,3,2,1); (3,2,1,4); (2,1,4,3)\}$, then a tertiary random decision is taken as to select the positions of the last stage in the sequence: (1,4,2,3) xor (1,2,4,3) xor (1,2,3,4), otherwise (1,4,3,2) xor (1,3,4,2) xor (1,3,2,4). And so on for junctions with more than four stages.

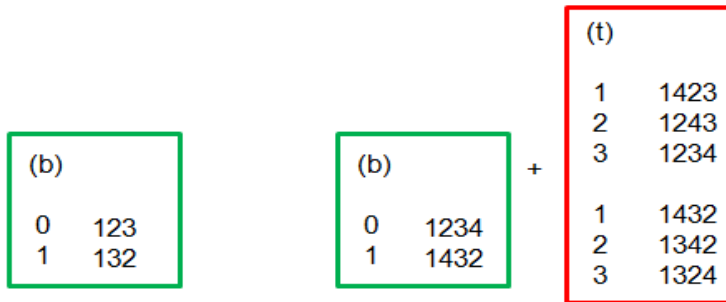


Fig.9.2 - Generation scheme through the adoption of discrete binary (b) and tertiary (t) variables. On the left the case of a three stage junction. On the right the case of a four stage junction.

In order to strengthen the algorithm exploitation, at each external iteration, the current optimum sequence combination (s) is stored and fixed for the successive neighbourhood generation steps (in this way the neighbourhood size will be equal to the total number of feasible combinations minus one) in order to intensify (refine) the search of the optimal greens and offsets (x) in the vicinity of s.

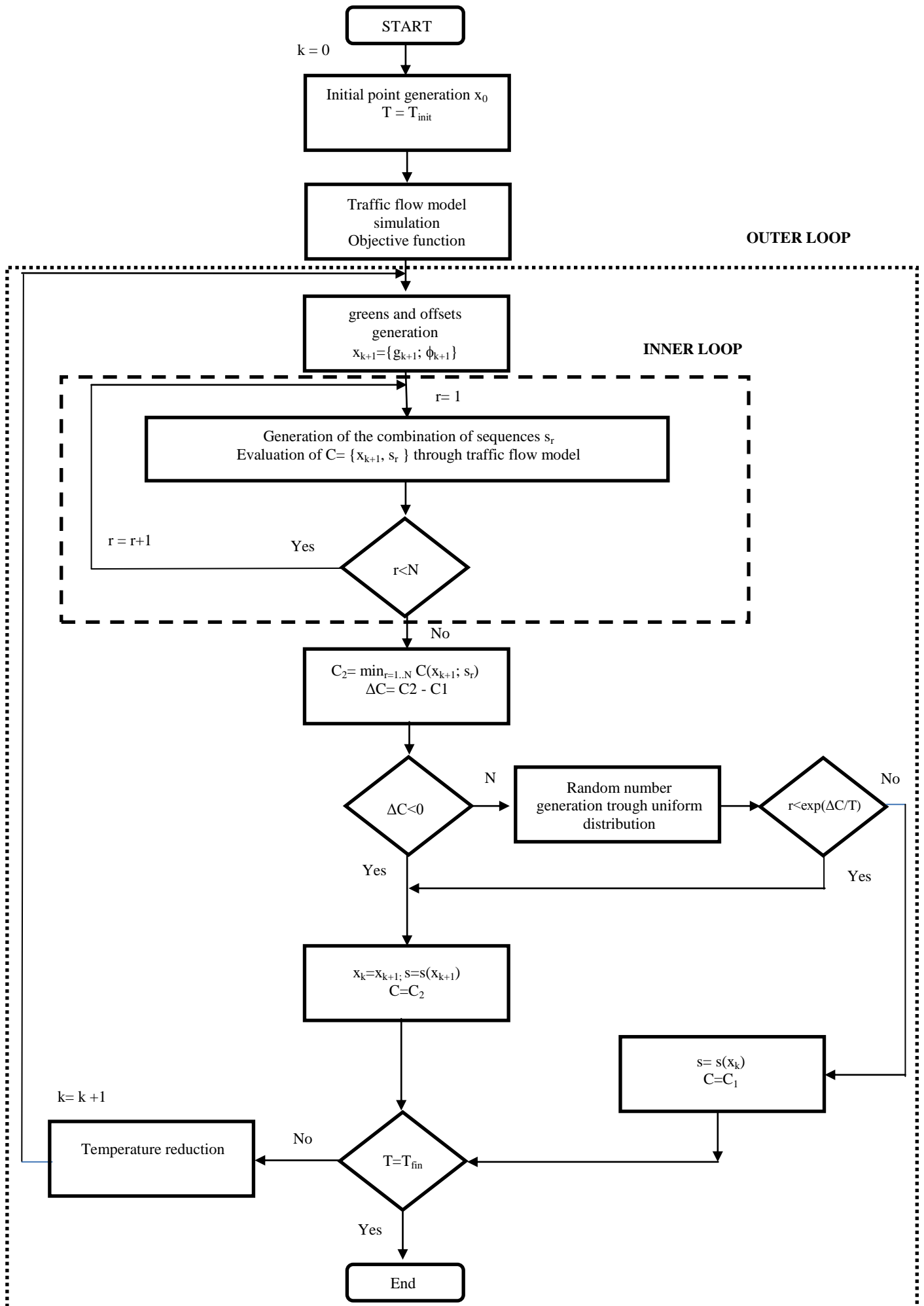


Fig.9.3 - Flow chart representation of CENEO

9.4 Numerical applications

In order to test the proposed method¹⁰, two test networks (case study A and B) were considered.

In particular case study A was introduced in order to

- point out the relevance of the green scheduling on the network performances;
- verify the results of the stage based implicit CENEO by checking the obtained solutions with those of the enumerative explicit ENEO.

whilst an extensive application of CENEO providing further details regarding the optimisation procedure is shown in case study B.

9.4.1 Case study (A)

In this section the scheduled synchronisation problem is applied to a small size triangular network. Both the explicit enumeration and the implicit enumeration stage based method were performed; the solution results are then compared. For the sake of clarity we need to state that the two solution approaches are not completely comparable due to the fact that in the stage based implicit enumeration method, the sequences are simultaneously optimised together with the greens and the offsets whilst in the explicit enumerative method two separate steps are identified: 1) the selection of the feasible combination of sequences; 2) the optimisation of the greens together with the offsets which are constrained to the selected combination of sequences.

It has to be noted that in the following case study we will indicate, with the *number of external iterations* (ei), the number of temperature states adopted to develop the simulated annealing scheme, while with the *number of internal iterations* (ii) the number of combinations (randomly selected from the equivalence classes sets) tested within each external iteration simultaneously with the current greens and offsets.

On the basis of previous considerations it is easy to understand how in the case of the explicit enumerative approach a higher number of traffic flow simulations¹¹ is required, if compared to the implicit approach¹². Nevertheless, such a comparison is performed to investigate the effectiveness of the proposed implicit method

The test network layout is shown in **Fig.9.4**, the entry/exit flows and the turning percentages (obtained through path flows) are shown respectively in

¹⁰ CENEO was implemented through MATLAB code (Release 2013b).

¹¹ The number of traffic flow simulations in case of explicit enumerative approach is equal to the product of the size of the set of feasible combinations of sequences and the number of algorithm iterations during the second step of the procedure, i.e. for the synchronisation.

¹² The number of traffic flow simulations in case of implicit approach is equal to the product of the ii which is always lower than the size of the complete set of combinations and the ei which is fixed as in the explicit approach.

Table 9.1 and **Table 9.2**; finally, in **Fig.9.5** the stage compositions and the equivalence classes of sequences are summarised. In accordance with the previous section which explains the optimisation problem, let m be the number of junctions in the network equal to 3 and n_i the number of feasible stages for each junction i (equal to 3 for junction 1, 4 for junction 2 and 3 for junction 3), and let us assume that all stages are compulsory for simplicity's sake, there are $(n_i - 1)!$ classes of equivalent sequences for each junction i (i.e. 2 classes for junction 1, 6 classes for junction 2 and 2 classes for junction 3), thus, there are $\prod_{i=1, m} (n_i - 1)!$ combinations of classes, equal to 24, to be analysed in order to establish the optimal one. Moreover, in our implementations two equivalent classes were further discarded at junction 2 (i.e. classes 1324 and 1423) since the approach j_{22} had green in two non-consecutive stages, so reducing the number of stage sequence combinations to 16.

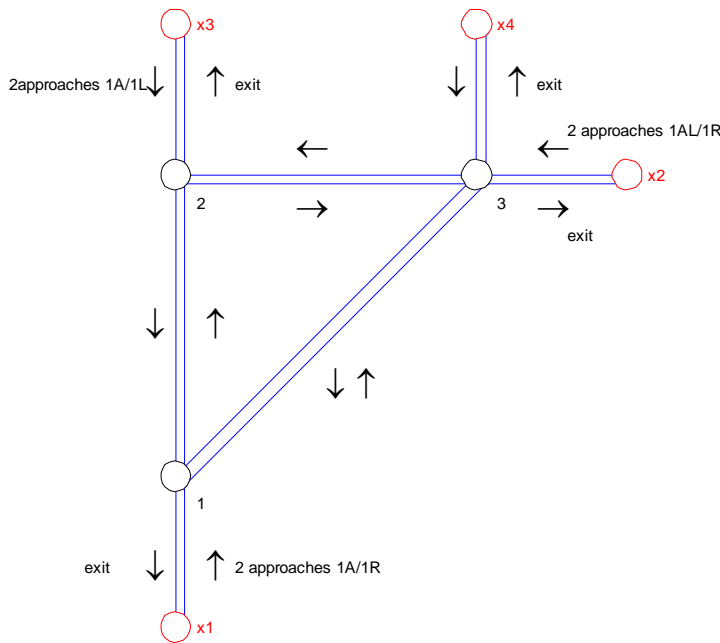


Fig.9.4 - Layout configuration of the “toy” triangular network. Exclusive turn lanes (when occurring) are identified with the corresponding symbol i.e. A=Ahead; L=Left; R=Right.

Table 9.1 - entry-exit flows

entry/exit	x1	x2	x3	x4	Total
x1	-	179	170	225	574
x2	132	-	45	39	216
x3	143	32	-	73	248
x4	302	69	28	-	399
Total	577	280	243	337	

Table 9.2 - turning percentages

Source	turn [%]		
	Ahead	Left	Right
x1-1	*	-	*
2-1	60	40	-
3-1	-	70	30
x2-3	70	30	*
x4-3	60	10	30
1-3	30	30	40
2-3	20	40	40
1-2	70	-	30
3-2	-	70	30
x3-2	*	*	-

*exclusive turn lane

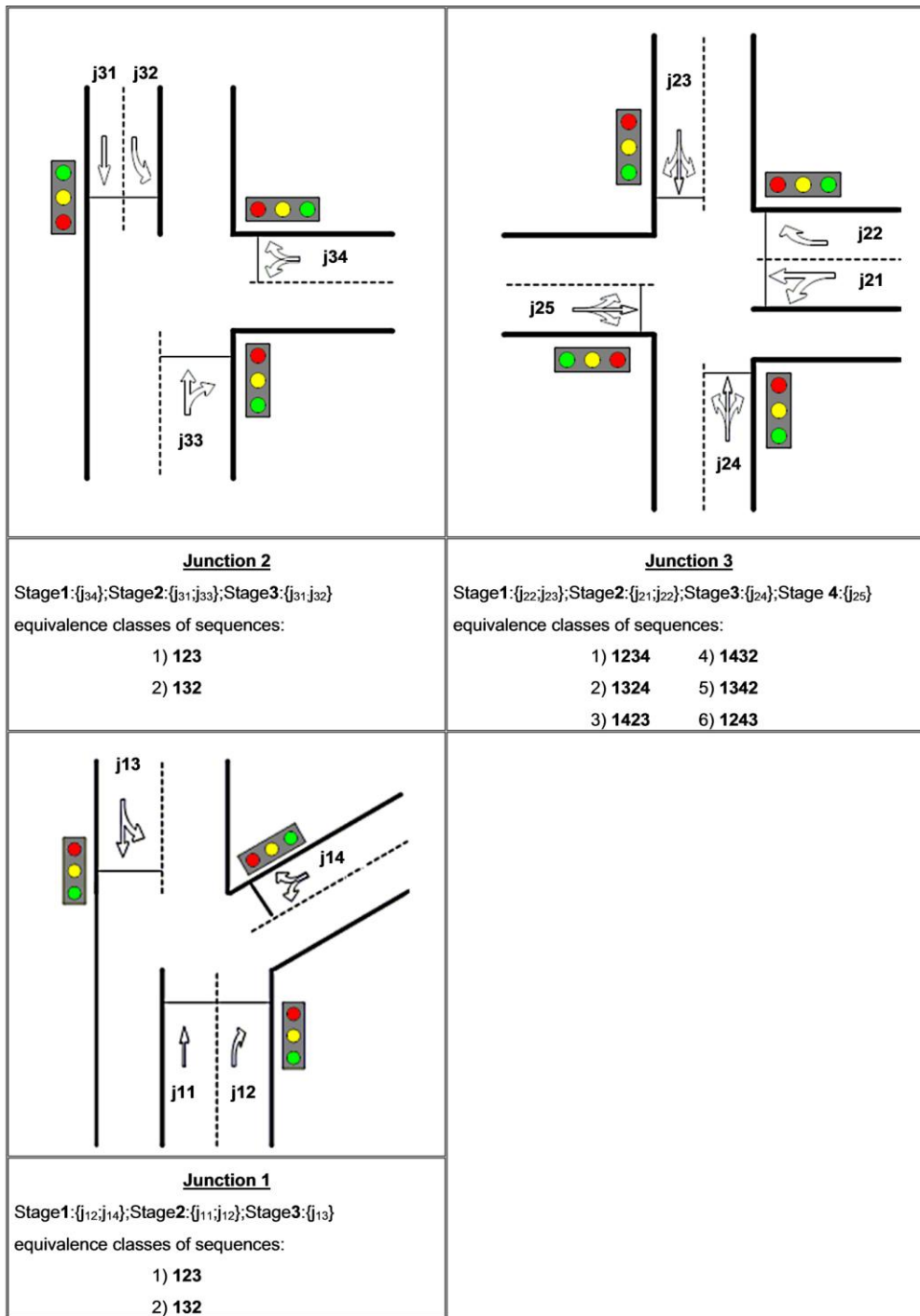


Fig.9.5 - Stage composition and classes of equivalence sequences for each junction in the network.

The results obtained through CENEO have been compared with those achieved by explicitly enumerating and analysing all feasible sequences through the procedure ENEO (Enumerative Network Optimisation), also implemented by the authors in a MATLAB (Release 2013b) code. This latter approach is clearly suitable only for a very small simple network (as occurred in this case study) due to the large number of feasible sequences greatly increasing with the number of junctions and the number of stages (see case study B) and it is presented below for comparison purpose only. In the ENEO procedure, three implementation steps are identified: 1) for each junction the adjacency (or compatibility) matrix is defined then all the equivalence classes of sequences are generated; 2) all the $\prod_{i=1, m} (n_i - 1)!$ combinations of sequences (of all the network junctions) are generated; 3) for each combination, greens and offsets optimising network TD are computed simultaneously (synchronisation) through a Simulated Annealing solution algorithm (see Cantarella et al., 2015). In **Table 9.3** the results obtained through ENEO are shown. As expected the combination of sequences greatly affect the network performances (the worst combination leads to TD close to 50% greater than the best combination) and, thus, support a proper investigation on the problem solution.

Table 9.3 - results computed through ENEO

ENEEO				
junction 1	junction 2	junction 3	TD [PCU/h]	Delta to best [%] External dispersion
Enumerative sequence	Enumerative sequence	Enumerative sequence		
132	1234	132	22.01	3
132	1243	132	23.14	8
132	1342	132	22.93	7
132	1432	132	27.46	28
123	1234	132	21.57	1
123	1243	132	24.85	16
123	1342	132	26.82	25
123	1432	132	21.43	0
123	1234	123	22.70	6
123	1243	123	24.53	14
123	1342	123	25.95	21
123	1432	123	22.01	3
132	1234	123	21.97	3
132	1243	123	23.72	11
132	1342	123	31.44	47
132	1432	123	26.88	25

In the following **Table 9.4** it is shown the internal dispersion of the solutions carried out through ENEO by comparing the results obtained in 8 simulation runs in which the combination of sequences is the best one previously obtained (see **Table 9.3**, sequence 123 for junction 1, 1432 for junction 2, 132 for junction3, which lead to a TD of 21.43 PCU-h/h). It should be stressed that 2 solutions with equal sequence combination may still differ with respect to greens and offsets due to the stochastic nature of the algorithm adopted for approximating the global optimum. Anyhow the fluctuations are less than 6% of the best solution.

Table 9.4 - Results computed trough ENEO over several runs on the best combination of sequences

ENEO					
junction 1	junction 2	junction 3	TD [PCU/h]	Sim run	Delta to best [%] Internal dispersion
constrained sequence	Constrained sequence	Constrained sequence			
123	1432	132	21.43	1	+2.19
123	1432	132	22.13	2	+5.53
123	1432	132	20.97	3	0.00
123	1432	132	21.33	4	+1.72
123	1432	132	21.49	5	+2.50
123	1432	132	21.41	6	+2.10
123	1432	132	22.05	7	+5.15
123	1432	132	21.37	8	+1.91

The analysis of **Table 9.4** might be repeated for all, or at least for the best solutions of **Table 9.3** (for example for the solutions which return TD not higher than 6% of the best one). In **Table 9.5** the outputs obtained through CENEO (ComplEte NEtwork Optimisation) are shown; in particular, the optimum TD and the optimum combination of sequences are reported.

Table 9.5 - results computed through CENEO

CENEO				
junction 1	junction 2	junction 3	TD [PCU/h]	Delta to best [%] ENEO
optimum sequence	Optimum sequence	Optimum sequence		
123	1432	123	22.06	+3.04

In the following **Table 9.6**, as for ENEO (**Table 9.4**) several runs¹³ are performed in order to verify the internal dispersion of solution results which was affected by the stochastic nature of the optimisation algorithm. As **Table 9.6** makes clear, the dispersion among different runs is small and in particular the first, the second and the third minimum are very close. Moreover, the fluctuations are less than the 9% of the best solution.

Table 9.6 - results computed through CENEO over several runs

CENEO					
junction 1	junction 2	junction 3	TD	Sim run	Delta to best [%]
optimum sequence	Optimum sequence	Optimum sequence	[PCU/h]		Internal dispersion
123	1432	123	22.06	1	+3.04
123	1234	132	21.41	2	0.00
123	1432	132	23.32	3	+8.92
123	1432	132	23.07	4	+7.75
123	1432	123	23.33	5	+8.97
123	1234	132	21.58	6	+0.79
132	1234	132	22.47	7	+4.95
123	1234	132	21.63	8	+1.03

It is worth noting that, as expected, the internal dispersion obtained in 8 simulation runs of CENEO (**Table 9.6**) results lower with respect to the internal dispersion in 8 simulation runs of ENEO (**Table 9.4**).

In the following **Table 9.7** an excerpt of the neighbourhood dynamic generation in a SA execution is shown. In particular, internal redundancy of the random solution generated during each SA iteration is tested in order to verify the algorithm exploration capability which consists of probing a much larger portion of the search space with the hope of finding other promising solutions that are yet to be refined. Furthermore, total redundancy is carried out in order to test algorithm exploitation, intensifying and refining the search in the vicinity of the current optimum sequence combination.

¹³ In general there is no indication about the number of runs to be carried out. In our case it was observed that the optimal solution was not significantly affected if the considered number of runs was greater than 8.

Table 9.7 - Dynamic neighbourhood generation for CENEO (an excerpt)

SA iteration	Neighbourhood generation	Junction 1	Junction 2	Junction 3	Internal redundancy #/5	Total redundancy #/120	
1	1	123	1234	123	1	7	
	2	123	1243	132	1	9	
	3	123	1432	123	1	12	
	4	*	123	1243	123	1	8
	5		132	1234	123	1	11
2	1	123	1243	123	1	8	
	2	132	1432	123	1	8	
	3	132	1342	132	2	5	
	4	132	1342	132	2	5	
	5	*	123	1234	123	1	7
3	1	123	1234	123	1	7	
	2	123	1342	132	1	10	
	3	*	132	1243	132	1	11
	4	123	1243	132	1	9	
	5	132	1432	132	1	6	
	~	~	~	~	~	~	
23	1	*	123	1432	123	2	12
	2		132	1432	132	1	6
	3		123	1234	132	1	4
	4		132	1243	132	1	11
	5		123	1432	123	2	12
24	1	*	123	1432	123	1	12
	2		123	1342	132	1	10
	3		132	1342	123	1	4
	4		132	1243	132	2	11
	5		132	1243	132	2	11

* current optimum

Finally, in **Table 9.8** the delta percentage of the optimum solution attained through the several simulation runs of CENEO from the worst and the best solution carried out through ENEO is shown. Therefore, we can appreciate how the solutions of CENEO approximate the optimum achieved through the explicit enumeration strategy.

Table 9.8 - Relative variation [%] among optimal CENEO solutions and best/worst ENEO solutions.

Sim run	CENEO - SA – CT&PDM					
	TD [PCU/h]	Delta to Best ENEO	Delta to Worst ENEO	junction 1	junction 2	junction 3
		[%]	[%]	optimum sequence	Optimum sequence	Optimum sequence
1	22.06	3	-43	123	1432	123
2	21.41	0	-47	123	1234	321
3	23.32	9	-35	123	1432	321
4	23.07	8	-36	123	1432	321
5	23.33	9	-35	123	1432	123
6	21.58	1	-46	123	1234	321
7	22.47	5	-40	321	1234	321
8	21.63	1	-45	123	1234	321

Summing up, considering that each simulation run in ENEO takes about 2 minutes whilst each simulation run in CENEO takes about 15 minutes, it can be observed (considering that only 16 feasible combination of sequences exist) that: 1) ENEO needs 32 minutes plus 16 minutes to get the approximate optimum and refine it with an improvement of 6% (which is affected by the stochastic nature of the adopted algorithm); 2) against, CENEO needs 18 minutes (and so saving 30 minutes with respect to ENEO) to get the approximate optimum, yielding poorer solution with respect to the best of ENEO even though with a discrepancy which is always less than 10%, and better solution with respect to the worst of ENEO, with a discrepancy which is always more than at least 35%.

It should be stressed that such a comparison results pointless when larger scale network are considered and an higher number of feasible sequences are identified (see case study B).

9.4.2 Case study (B)

In this section the application of the scheduled synchronisation problem to a more complex test network is shown. The network layout is shown in **Fig.9.6**, while the stage compositions and the equivalence classes of sequences are summarised in **Fig.9.7**.

As in the previous section, let m be the number of junctions in the network equal to 5 and n_i the number of feasible stages for each junction i (equal to 2 for junction 1,

3 for junction 2 and 4 for junction 3,4,5), and let us assume that all stages are compulsory for simplicity's sake, there are $(n_i - 1)!$ classes of equivalent sequences for each junction i (i.e. 1 class for junction 1, 2 classes for junction 2 and 6 classes for junction 3,4,5), thus, there are $\prod_{i=1, m} (n_i - 1)!$ combinations of classes, equal to 432, to be analysed.

Unlike case study A, no comparison between explicit (ENEO) and implicit (CENEO) enumeration approaches are carried out and only the second one is adopted. That is because in doing so (to match the results attained by CENEO with those of ENEO) we need to explore 432 combinations of sequences, all of which require us to address a synchronisation, thus, a complete simulated annealing implementation. In this way, via the explicit enumeration we would achieve $24 \times 432 = 10368$ traffic flow simulations and so a number of solutions which is too great to explore, which leads to a unpractical pursuable approach. Moreover, as previously stated the two solution approaches are not exhaustively comparable due to the fact that in CENEO the sequences are simultaneously optimised together with the greens and the offsets whilst in ENEO two separate steps are identified: 1) the selection of the feasible combination; 2) the optimisation of the greens together with the offsets which are constrained to the selected combination of sequences.

In order to analyse the effect of the neighbourhood generation a comparative analysis was worked out to handle the number of internal algorithm iterations (and maintaining the number of external iterations to 24) in such a way to investigate only the neighbourhood exploitation (considering that its size increases congruently with ii) which affects the significance of the error percentage obtained within each external iteration. Therefore, (as shown in **Fig.9.8**, **Fig.9.9** and **Fig.9.10**) the trends of the performance indicators (in terms of network TD) achieved considering different numbers of internal iterations were evaluated (in this case $ii=3;5$ and 8) in order to verify the solution range of variation. To this aim, a box plot is represented in which internal (int), maximum (max) and minimum (min) performance indicator, as well as current optimum (c.o.) are shown.

The second analysis aims to equalise the time consuming independently from the neighbourhood generation, thus, the number of external iterations was constrained in order to fix an upper bound for the number of algorithm implementations (as the product of the internal and external iterations). In particular, it was assumed that such a bound was nearly a third of the total number of feasible combinations. On that basis, as shown in **Fig.9.11** we maintain 24 external iterations when $ii=5$ whilst we reduce ei to 15 when $ii=8$ and we extend ei to 40 when $ii=3$. Such an assumption implies that different temperature decrements have to be considered.

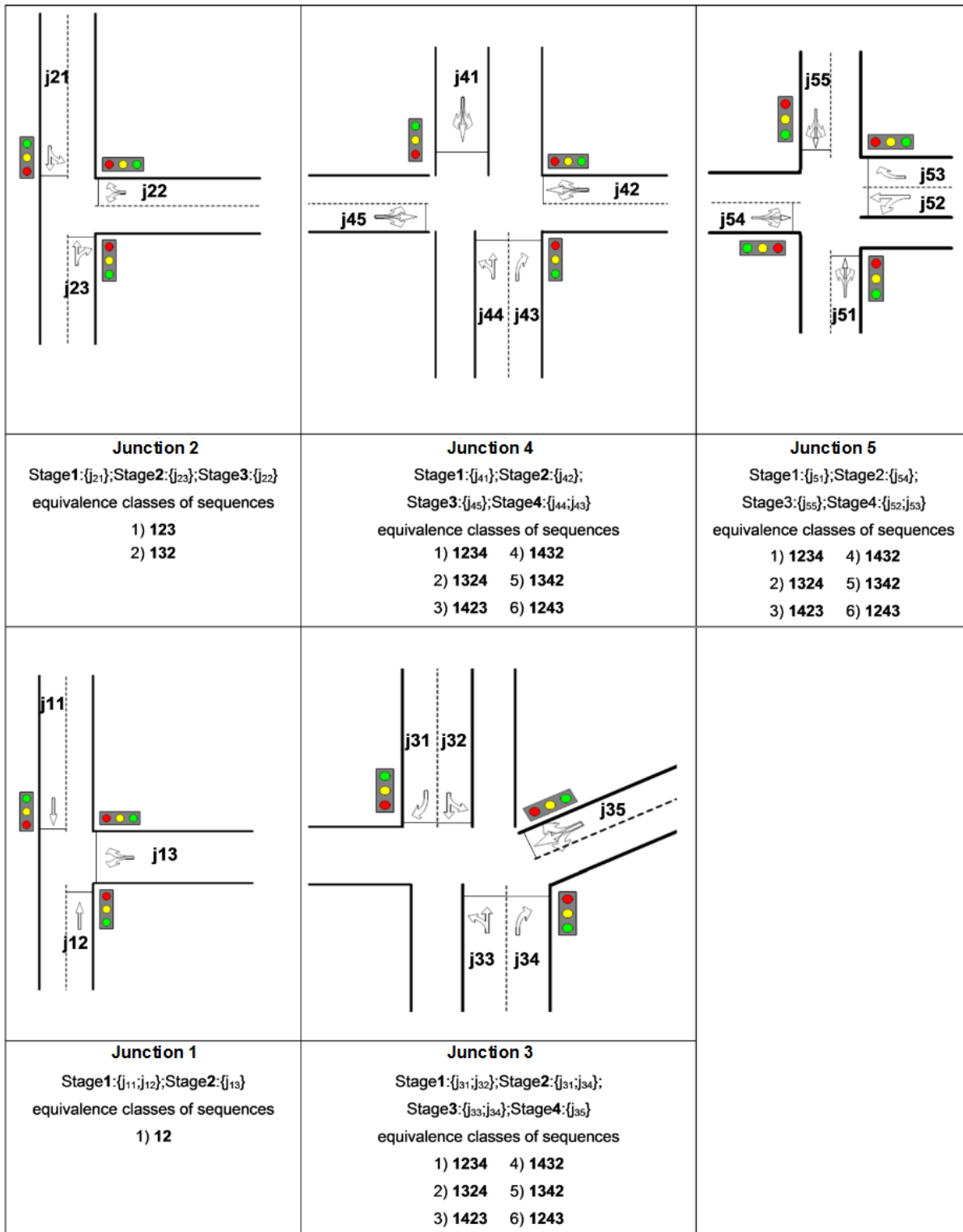


Fig.9.7 - Stage composition and classes of equivalence sequences for each junction in the network.

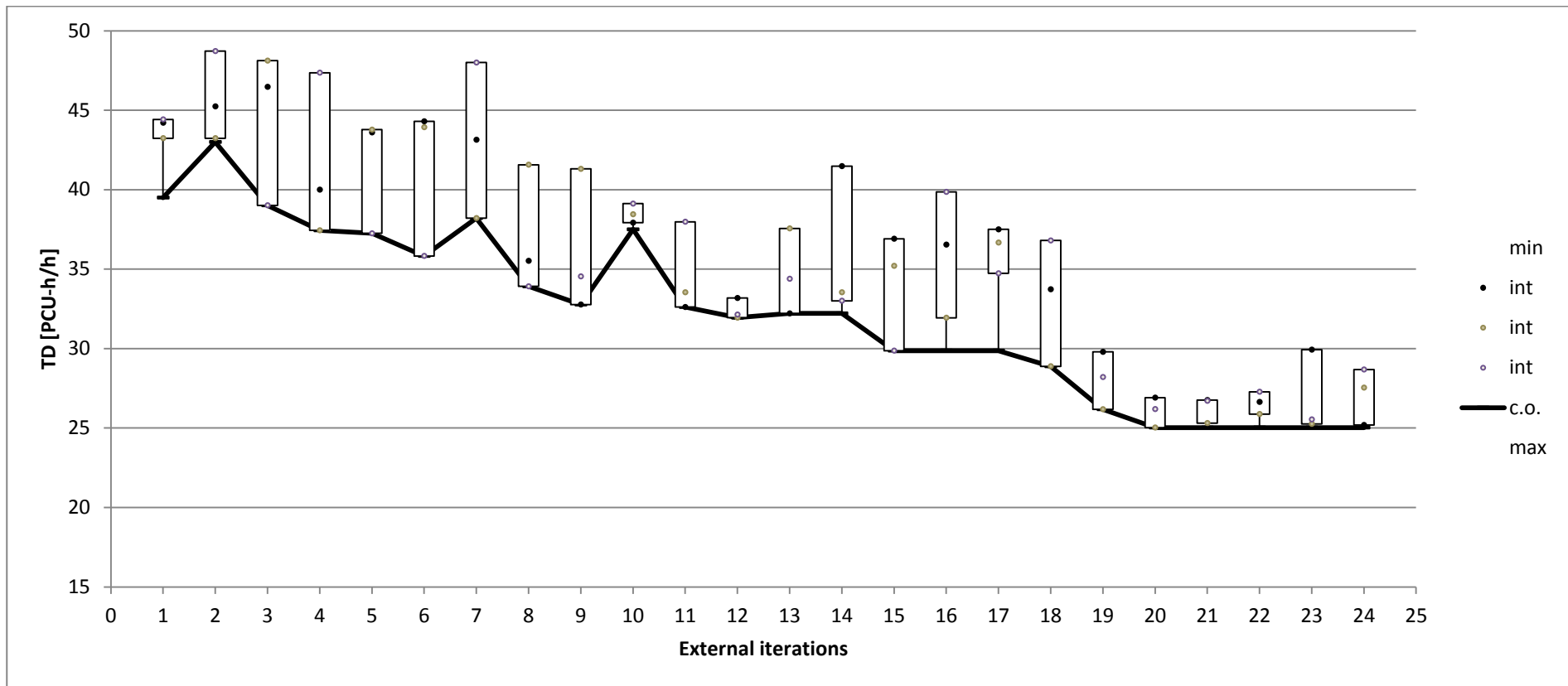


Fig.9.8 - Trends of the performance indicators (in terms of network TD) against external iterations (external iterations constrained to 24; internal iterations constrained to 3).

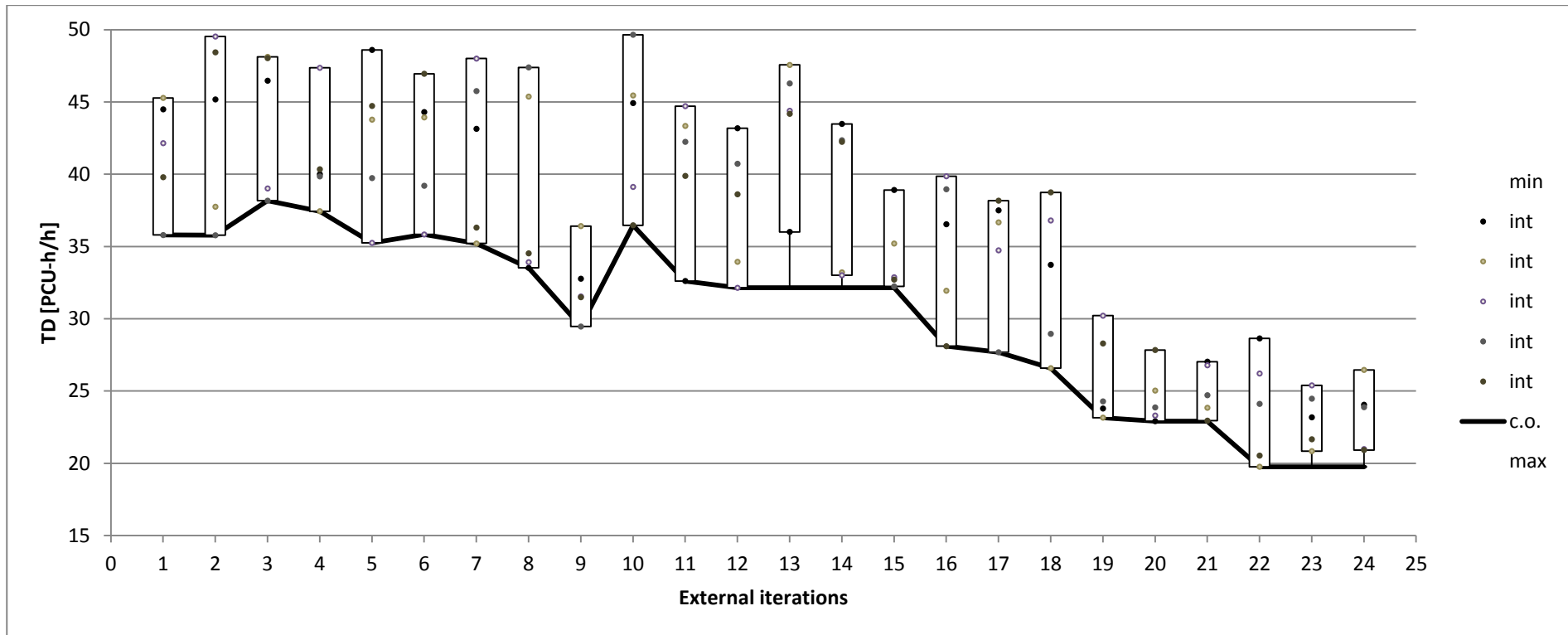


Fig.9.9 - Trends of the performance indicators (in terms of network TD) against external iterations (external iterations constrained to 24; internal iterations constrained to 5).

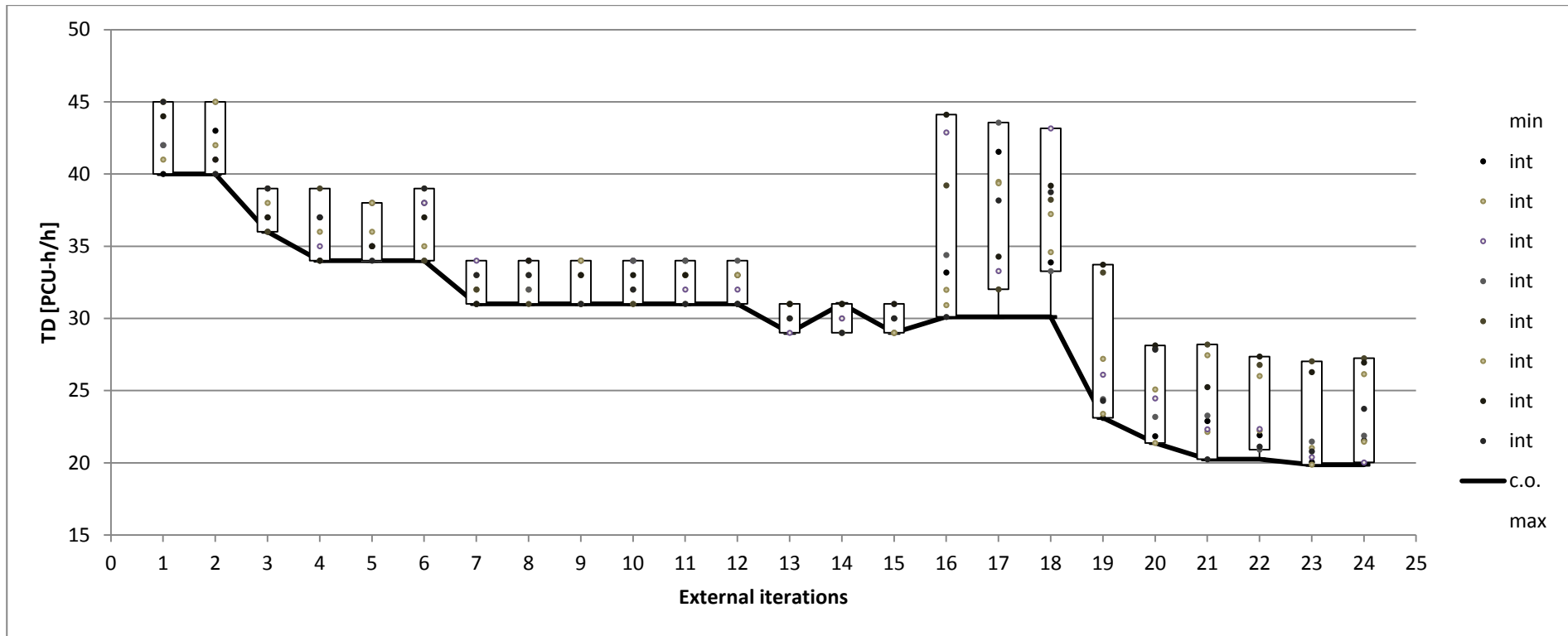


Fig.9.10: Trends of the performance indicators (in terms of network TD) against external iterations (external iterations constrained to 24; internal iterations constrained to 8).

The results obtained for the different classes of internal iterations were compared considering three measures of performance (see **Table 9.9**)

- 1) the optimum carried out in terms of TD;
- 2) the running time of the algorithm;
- 3) the mean range of variation of the solutions carried out during the internal iterations.

Table 9.9 - performance indicators w.r.t. different values of internal iterations ($ii=3, 5$ and 8)

internal iterations	running time [min]	TD [PCU h/h]	mean internal range [%]
3	12	25.03	16.46
5	21	19.76	32.01
8	30	19.87	35.11

It should be observed that the running time grows linearly with respect to the number of internal iterations. Both the optimum achieved by the algorithm (in terms of TD) and the mean internal range of variation (higher values result in a larger portion of the search space being explored) are noticeably better when $ii=5$ and $ii=8$ are considered as input. Such a result is due to the fact that presumably only in few cases a significant effect could be observed by switching the combination of stage sequences and then no large internal refining are necessary. On that basis, in this case study the choice of considering the input parameter as the class of 5 internal iterations results as the best one, allowing a considerably safe computation time and ensuring contemporary promising optimal solutions.

In order to equalise the time consuming without assuming any variation of the number of internal iterations we perform, as previously described, a further analysis (see **Fig.9.11**) which consists in adopting three different temperature decrements, each one derived from a different temperature range. In particular, we assume that the start and the final temperature are fixed but we vary the number of temperature states in order to generate different decrement lengths. The case of 5 internal iterations was considered as reference, thus, we obtain 15 external iterations when $ii=8$ and 40 external iterations when $ii=3$.

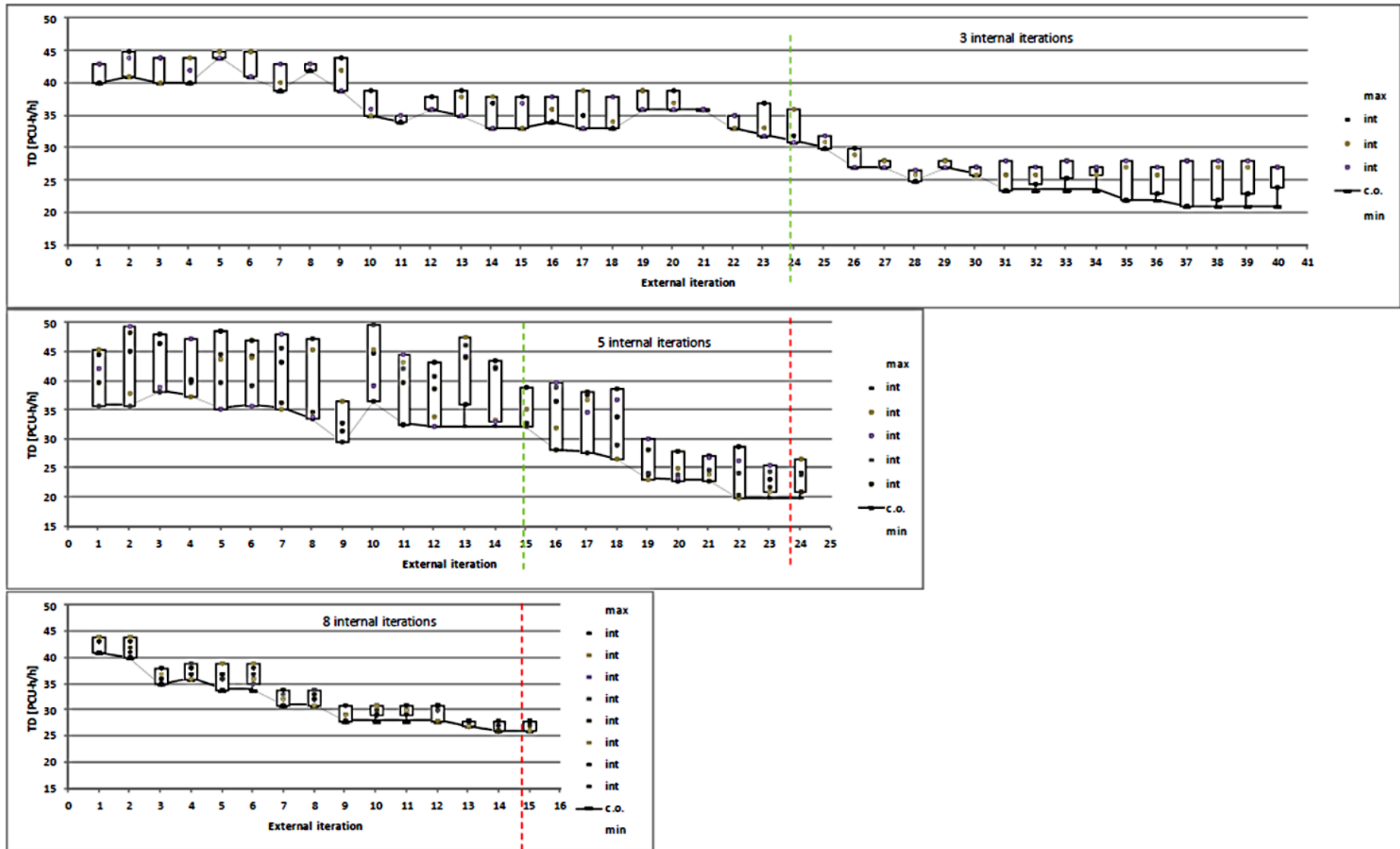


Fig.9.11 - Comparison of three trends of the performance indicators (in terms of network TD) against external iterations (case 1: internal iterations=3; external iterations=40; case 2: internal iterations=5; external iterations=24; case 3: internal iterations=8; external iterations=15;).

As expected when $ii=8$ and $ei=15$, as high temperature decrements are adopted only a few search spaces are explored, thus a proxy of a local search is performed yielding a poor optimum result. However, when $ii=3$ and $ei=40$ low temperature decrements are applied in order to randomise the search for a number of external iterations until the temperature was cool enough to make the algorithm act as a simulated annealing¹⁴; as a matter of fact, the starting point of the cooling process is not good enough to generate a solution which is better with respect to the reference one attained considering $ii=5$ and then $ei=24$. Nevertheless, even though it is quite clear that a large number of internal iterations, on the basis of the trade-off between the results obtained in the two aforementioned analyses, leads to worse global performances (especially in terms of time consuming), no general considerations can be made when lowering the number of external iterations since the results are affected by a certain randomness.

9.5 Conclusions

This paper focuses on a general framework for Network Signal Setting Design (NSSD) including green scheduling, green timing and coordination into the so-called scheduled synchronisation. The proposed method (CENEO- ComplEtE Network Optimisation) optimises stage sequence, thus, it is an extension of the synchronisation method and the traffic flow model (ENEO - Enumerative NETwork Optimisation) proposed in Cantarella et al. (2015) which requires the stage sequence as input data. CENEO is a stage-based method, thus, simpler than approach-based.

In order to discuss the effectiveness of the proposed method two networks have been considered: with three and five nodes. For the three node network the results performed by CENEO are compared with those carried out by ENEO¹⁵ in which all possible stage sequences are explicitly enumerated. This approach is, in fact, clearly suitable only for a very small simple network, due to the large number of feasible solutions greatly increasing with the number of junctions and the number of stages. Indeed, in the case of the three node network only 16 feasible combinations of sequences exist whilst in the case of the five node network (with two extra junctions) ENEO needs to explore 432 feasible combinations of sequences. Moreover, it is worth noting that even though in the three node network only 16 feasible combinations of sequences exist, the explicit and implicit approaches are not completely comparable; indeed, in CENEO the sequence is simultaneously optimised together with the greens and the offsets whilst in ENEO two separate steps are needed: 1) the list of all feasible sequence combinations; 2) the optimisation of the greens together with the offsets for each combination of sequences.

The values of the performance indicators provided by CENEO over 8 simulations are in one case the best feasible solution, in four cases within an error less than 5%

¹⁴ A more detailed description of the Simulated Annealing is given in Appendix A.

¹⁵ The TD has been considered as performance indicator in all cases.

and in three cases within an error less than 10% (see **Table 9.8**) of the optimal solution generated by ENEO through complete enumeration of all combinations.

For the five node network, some operational considerations on CENEO iterations and time consuming were discussed. The iterations were distinguished in external and internal; the former representing the number of temperature states adopted to apply the simulated annealing scheme, the latter representing the number of combinations (randomly selected from the equivalence classes) tested within each external iteration together with the current greens and offsets.

Based on these considerations two kinds of comparisons were carried out: firstly, for a fixed value of external iterations, the effect of different values of internal iterations (3, 5 and 8) were tested (see **Fig.9.8, Fig.9.9, and Fig.9.10**) then the external and internal iterations pairs were obtained by fixing their product. Results of both analyses confirm that higher values of internal iterations are not necessarily required; that is justified, in the first case, due to the fact that presumably only in few circumstances a significant effect by switching the combination of stage sequences could be observed, whilst in the second case, due to the fact that by constraining the number of external iterations the explored search spaces is smaller, thus, a proxy of a local search is performed yielding a poor optimum result.

Future research perspectives that will be discussed in papers under development are:

- the specification of a macroscopic traffic flow model rather than to the link flow pattern, which may greatly affect proper modelling of exiting flow diversion at junctions;
- the investigation of hybrid meta-heuristics (i.e. population based with neighbourhood based methods) presumably more suitable for large scale network.

Further topics worthy of research effort are: NSSD based on reserve capacity maximisation at network level, and the specification of solution algorithms following the approach- (phase-) based methodology.

Acknowledgements

This research has been partially supported by the University of Salerno, Italy, EU under PhD program on transportation (Ph.D. School in Environmental Engineering), local grant n. ORSA132945 financial year 2013 and under 'APPS4SAFETY – PON03PE_00159_3'.

For our friendship policy, the authors' order is organised as in follows: the first author is a PhD Candidate whose research thesis mainly focused on the topic of the paper, the other authors are in alphabetical order.

References

- Binning, J. C., Crabtree, M. R., & Burtenshaw, G. L. (2010) Transyt 14 user guide. Transport Road Laboratory Report nr AG48. APPLICATION GUIDE 65 (Issue F).
- Bron, C., & Kerbosch, J., 1973. Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 16(9), 575-577.
- Burrow, I.J., 1987. OSCADY : a computer program to model capacities, queues and delays at isolated traffic signal junctions: Traffic Management Division, Traffic Group, Transport and Road Research Laboratory, Crowthorne, Berkshire.
- Cantarella, G. E., de Luca, S., Di Pace, R., & Memoli, S., 2015. Network Signal Setting Design: meta-heuristic optimisation methods *Transportation Research C* 10.1016/j.trc.2015.03.032
- Cantarella, G. E., de Luca, S., Di Gangi, M., Di Pace, R., & Memoli, S., 2014. Macroscopic vs. mesoscopic traffic flow models in signal setting design. In *Intelligent Transportation Systems (ITSC)*, 2014 IEEE 17th International Conference 2221-2226
- Daganzo, C.F., 1994. The cell-transmission model. Part 2: Network traffic, University of California, Berkeley, California.
- Hadi, M. A. & Wallace, C. E., 1993. Hybrid genetic algorithm to optimize signal phasing and timing *Transpn.Res.Rec.*, 1421, 94-103.
- Hadi, M. A., & Wallace, C. E. (1994). Optimization of signal phasing and timing using cauchy simulated annealing. *Transportation Research Record*, (1456), 64-71.
- Improta, G. & Cantarella, G. E., 1984. Control system design for an individual signalized junction. *Transportation Research Part B: Methodological*, 18(2), 147-167.
- Metropolis, N. Rosenbluth, A.W. Rosenbluth, M.N. Teller, A.H. Teller, E., 1953 Equation of state calculation by fast computing machines *J. Chem. Phys.*, pp. 1087-1092.
- Park, B., B., Messer, J., C., Urbanik II, T., 2000 Enhanced Genetic Algorithm for Signal-Timing Optimization of Oversaturated Intersections. *Transportation Research record* 1727, no.00.1661.

About the author:

Silvio Memoli achieved in December 2012 his M.Sc title in “Environmental Engineering, Land-conservation” at the Civil Engineering Department, University of Salerno. In January 2013, he became a Ph.D student in Transportation Engineering, scholarship winner, at the Department of Civil Engineering, at the University of Salerno. There he is being mentored by Professor. G.E. Cantarella with the research fellowship topic of "Network Signal Setting Design at urban level". At Salerno he contributes teaching in the courses of “Transportation System Theory ”, “Transportation System Design”, “Transportation Planning” and “Transportation Techniques and Economics ”. Besides he collaborates also supervising the realization of Bachelor theses, in the transport sector. He has participated in several training courses in Transportation Engineering (e.g. the course of prof. Ben-Akiva and prof. Cascetta, held in Naples on may 2014). He is coauthor in some publications in scientific journals, and international conference proceedings. He is also collaborating to professional consulting activities conducted by the Department of Civil Engineering, at the University of Salerno.