Soft Computing Letters 1 (2019) 100001

Contents lists available at ScienceDirect

Soft Computing Letters

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

A hybrid heuristic approach for traffic light synchronization based on the MAXBAND

Xavier Cabezas[a,b,*], Sergio García[a], Santiago D. Salas[b]

[a] College of Science and Engineering, School of Mathematics, The University of Edinburgh, James Clerk Maxwell Building, The King's Buildings, Peter Guthrie Tait Road, Edinburgh, EH9 3FD, United Kingdom

[b] Escuela Superior Politécnica del Litoral, ESPOL, Facultad de Ciencias Naturales y Matemáticas, Campus Gustavo Galindo Km. 30.5 Vía Perimetral, P.O. Box 09-01-5863, Guayaquil, Ecuador

ARTICLE INFO

Keywords:
Traffic light synchronization
Hybrid heuristics
Tabu search
Variable neighborhood search
MAXBAND

ABSTRACT

This study addresses a high resolution model for the synchronization of traffic lights on transport networks. A hybrid heuristic algorithm optimizes the mixed integer linear model, referred as MAXBAND, seeking to achieve maximal bandwidth by setting arterial signals. The assessed algorithm corresponds to a hybrid metaheuristic which combines Tabu Search and Variable Neighborhood Search. The algorithm uses a memory structure within an iterative local search, allowing a broader diversity of solutions. In addition, some adjustments were incorporated to the MAXBAND such as the revision of the constraints of the mixed integer linear model, including those that describe all the cyclic routes in the graph, and some bounds were generalized for integer variables. Extensive computational experiments were carried out evidencing a competitive performance for large instances.

1. Introduction

Over recent years, traffic congestion has become an issue of remarkable importance due to an increase of vehicles on the roads in urban areas. For controlling the flow of vehicles, traffic lights have been used as regulators since the late 19th century. Nevertheless, its use also leads to other problems which include time delays for vehicles moving from one place to another, and an increase in pollution due to the constant changes in the speed of vehicles [1]. In this sense, traffic light synchronization, which embraces the regulation in the timing of traffic lights, is a topic of noteworthy importance. This problem is commonly tackled following two approaches, by minimizing the delays or some other measures which assess the performance of traffic, or by maximizing the time in which vehicles move without stopping at the red lights, referred also as the bandwidth maximization [2].

In terms of traffic flow measures, usually a minimization problem is formulated to reduce the overall delay, or to reduce the number of stops vehicles complete. In a classic study, a convex nonlinear formulation with a piece-wise linearization and an increased number of constraints was proposed by Gartner et al. (1975) [3]. Similarly, an analogous linear model permitted to decide among different signal timing plans while relaxing the assumption of a common red-green period width [4]. However, the latter is not a hard constraint because in that case the least common multiple of all red-green periods could be used as a uniform period, as mentioned in [5].

Concerning the bandwidth maximization, initial formulations with preassigned vehicles velocity deemed a geometric method for a two-way street given a fixed and common red-green time period on each signal [6]. Subsequently, a mixed integer linear program (MILP) was proposed to solve a more complete version of the problem [7]. The selection of the red-green time period occurred between given bounds. Speed variations and bounds on the velocity limits between adjacent signals were included. Additionally, an extension was provided to solve the same problem on general networks, but only very small instances could be solved. This is a more difficult problem because it is necessary to introduce the so-called loop constraints, which permit to model the circular movements that vehicles can do. The introduction of a generalization containing left turns at the junctions led to the MAXBAND model [8]. On its first version, the MAXBAND handled problems on networks with only 3 arteries and up to 17 traffic signals. The MAXBAND was extended to work with a variable bandwidth for each street segment originating the MULTIBAND model [9]. This modification incorporated a traffic factor on the objective function. Recent variations have derived a new version referred as the AM-BAND, which attempts to use better the available green times on both road directions by relaxing the symmetric assumption with respect to the progression line [10]. Finally, the inclusion of variable bandwidths permitted to consider the impact of the speed variation to the model [11,12].

Metaheuristic approaches are flexible and robust strategies that have demonstrated to be computationally efficient in terms of CPU time. They

* Corresponding author.
E-mail addresses: joxacabe@espol.edu.ec (X. Cabezas), sergio.garcia-quiles@ed.ac.uk (S. García), sdsalas@espol.edu.ec (S.D. Salas).

https://doi.org/10.1016/j.socl.2019.100001
Received 9 October 2019; Accepted 6 November 2019

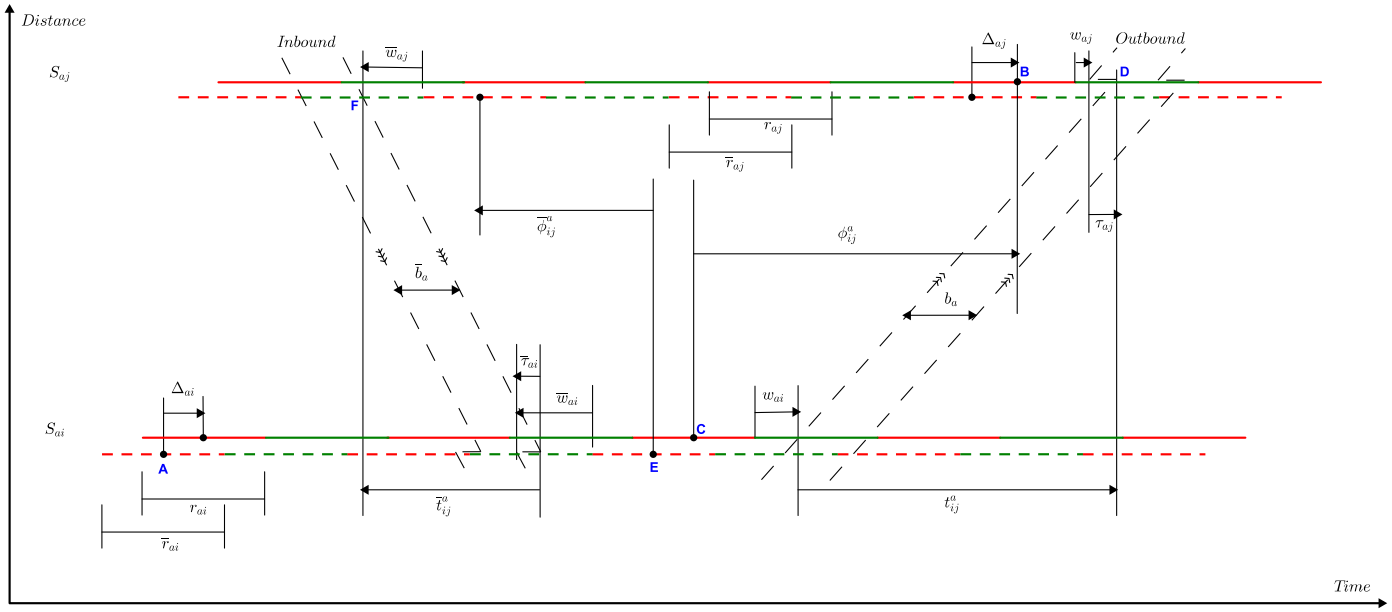

Fig. 1. Geometry for MAXBAND model on artery a.

have the competency of solving large-scale problems such as the traffic light synchronization [13]. For instance, the commercial software for synchronization of traffic signals and traffic management TRANSYT provides a heuristic solution for a complete and robust objective function [14,15]. TRANSYT uses microscopic simulations of traffic behavior ensemble with genetic algorithms. In the case of the MAXBAND model, a heuristic method was implemented for the network problem by initially solving a tree sub-problem which considered measures of interest. The integer variables were fixed to the values obtained and used in a second stage to obtain a solution for the whole problem [16]. In addition, numerous heuristics have been employed in traffic light scheduling including, evolutionary methods [17,18], particle swarm optimization [19], and a hybrid heuristic which combined harmony search with local search [20]. The hybridization of a metaheuristic with local search strategies enhances significantly the success rate in much less CPU time [21,22].

Tabu Search is a global optimization algorithm originally proposed by Glover (1986) [23]. It has proven applicability in operations research applications including scheduling [24,25], vehicle routing problems [26], and health care [27]. In this work, a hybrid heuristic algorithm based on Tabu Search is developed for bandwidth maximization. The algorithm is enhanced by intensifying its search using a sequence of neighborhood solution processes, which follows the idea of the Variable Neighborhood Search (VNS) [28]. The MAXBAND problem is typically solved for very small instances by optimization solvers. In this sense, this contribution aims to adapt a hybrid heuristic algorithm for bandwidth maximization to expand its capabilities and to solve the problem for complete networks. To the best of our knowledge, there are no references in the literature about solving the MAXBAND on a complete network nor using other metaheuristic approaches.

The rest of the paper is structured as follows. Section 2 introduces and reviews the MAXBAND model, including a detailed explanation of its elements and the notation to be used in the manuscript. Section 2.4 explains the approach for modelling the loop constraints with a small number of binary variables. Section 3 includes a generalization of the bounds for arterial integer variables. Section 4 introduces the MILP-based hybrid heuristic algorithm based on Tabu Search and VNS. Later, computational evaluations are carried out to verify the efficiency and effectiveness of the proposed method for large instances. Finally, in Section 5 conclusions and future research are discussed.

2. MAXBAND modelling for a network

Let us consider a group of two-way *arteries* (streets) that meet each other at *junctions* to form a transport *network*. This network has some traffic signals to regulate its traffic. They work with a common *period* that splits into red and green time. Even though it is sometimes referred as *cycle length*, to avoid confusion with the *loops* (also known as cycles), the term to be used through the paper is *period length*. The distances (time units) that allow to measure the relative location between two signals on the same artery and on different arteries are called *internode offset* and *intranode offset*, respectively. A list of offsets for the signals is said to be a *synchronization*.

The MAXBAND model relies on the geometry illustrated in Fig. 1. Information for two signals S_{ai} and S_{aj} is provided on an artery a. The notation is similar to the one proposed in [8].

The parameters in Fig. 1 are defined as follows,

- T: Period length, in seconds.
- n_a: Number of traffic lights (signals) on artery a.
- r_{ai} ($\overline{r}_{ai}$) : Outbound (inbound) red time of signal i on artery a, in periods.
- τ_{ai} ($\overline{\tau}_{ai}$) : An advancement of the outbound (inbound) bandwidth upon leaving S_i, in periods.

The variables in Fig. 1 are defined as follows,

- z: Signal frequency, in periods per second.
- b_a ($\overline{b}_a$) : Outbound (inbound) bandwidth on artery a, in periods.
- t^a_{ij} ($\overline{t}^a_{ij}$) : Travel time from S_{ai} to S_{aj} in outbound (from S_{aj} to S_{ai} in inbound) direction, in periods.
- ϕ^a_{ij} ($\overline{\phi}^a_{ij}$) : Time from the center of red at S_{ai} to the center of red at S_{aj}, in periods. The two reds are chosen so that each is immediately to the left (right) of the same outbound (inbound) green band. ϕ^a_{ij} ($\overline{\phi}^a_{ij}$) is positive if S_{aj}'s center of red lies to the right (left) of S_{ai}'s.
- w_{ai} ($\overline{w}_{ai}$) : Time from the right (left) side of S_{ai}'s red to the left (right) side of green band in outbound (inbound) direction, in periods.
- Δ_{ai}: Time from center of $\overline{r}_{ai}$ to the nearest center of r_{ai}, in periods. It is positive from left to right.

For the complete formulation of the MAXBAND please refer to LM 2.1, which holds additional parameters and variables not included

LM 2.1 MAXBAND, Maximal Bandwidth Formulation.

$$\text{Maximize} \quad \sum_{a\in A}\left(k_a b_a + \overline{k}_a \overline{b}_a\right) \tag{1}$$

subject to :

$$\frac{1}{T_2} \le z \le \frac{1}{T_1}, \tag{2}$$

$$w_{ai} + b_a \le 1 - r_{ai}, \forall a \in A, \forall i = 1, \dots, n_a, \tag{3}$$

$$\overline{w}_{ai} + \overline{b}_a \le 1 - \overline{r}_{ai}, \forall a \in A, \forall i = 1, \dots, n_a, \tag{4}$$

$$\begin{aligned}(w_{ai} + \overline{w}_{ai}) - (w_{a,i+1} + \overline{w}_{a,i+1}) + (t_i^a + \overline{t}_i^a)\\ +(\delta_{ai}\ell_{ai} - \overline{\delta}_{ai}\overline{\ell}_{ai}) - (\delta_{a,i+1}\ell_{a,i+1} - \overline{\delta}_{a,i+1}\overline{\ell}_{a,i+1}) + (r_{ai} - r_{a,i+1})\\ -(\tau_{a,i+1} + \overline{\tau}_{ai}) = m_i^a, \forall a \in A, \forall i = 1, \dots, n_a - 1,\end{aligned} \tag{5}$$

$$\left(\frac{d_i^a}{f_i^a}\right) z \le t_i^a \le \left(\frac{d_i^a}{e_i^a}\right) z, \forall a \in A, \forall i = 1, \dots, n_a - 1, \tag{6}$$

$$\left(\frac{\overline{d}_i^a}{\overline{f}_i^a}\right) z \le \overline{t}_i^a \le \left(\frac{\overline{d}_i^a}{\overline{e}_i^a}\right) z, \forall a \in A, \forall i = 1, \dots, n_a - 1, \tag{7}$$

$$\left(\frac{d_i^a}{h_i^a}\right) z \le \left(\frac{d_i^a}{d_{i+1}^a}\right) t_{i+1}^a - t_i^a \le \left(\frac{d_i^a}{g_i^a}\right) z, \forall a \in A, \forall i = 1, \dots, n_a - 2, \tag{8}$$

$$\left(\frac{\overline{d}_i^a}{\overline{h}_i^a}\right) z \le \left(\frac{\overline{d}_i^a}{\overline{d}_{i+1}^a}\right) \overline{t}_{i+1}^a - \overline{t}_i^a \le \left(\frac{\overline{d}_i^a}{\overline{g}_i^a}\right) z, \forall a \in A, \forall i = 1, \dots, n_a - 2, \tag{9}$$

$$\sum_{(i,j):\, a\in A_\zeta^F} \phi_{ij}^a - \sum_{(i,j):\, a\in A_\zeta^B} \phi_{ij}^a + \sum_{(b,j,i,c,k)\in J_\zeta} \Psi_{S_{bj},S_{ck}}^i = C_\zeta, \forall \zeta \in \mathcal{B}_\zeta, \tag{10}$$

$$C_\zeta \in \mathbb{Z}, \forall \zeta \in \mathcal{B}_\zeta, \tag{11}$$

$$m_i^a \in \mathbb{Z}, \forall a \in A, \forall i = 1, \dots, n_a - 1, \tag{12}$$

$$\delta_{ai}, \overline{\delta}_{ai} \in \{0, 1\}, \forall a \in A, \forall i = 1, \dots, n_a - 1, \tag{13}$$

$$b_a, \overline{b}_a, t_i^a, \overline{t}_i^a, w_{ai}, \overline{w}_{ai}, z \ge 0, \forall a \in A, \forall i = 1, \dots, n_a - 1. \tag{14}$$

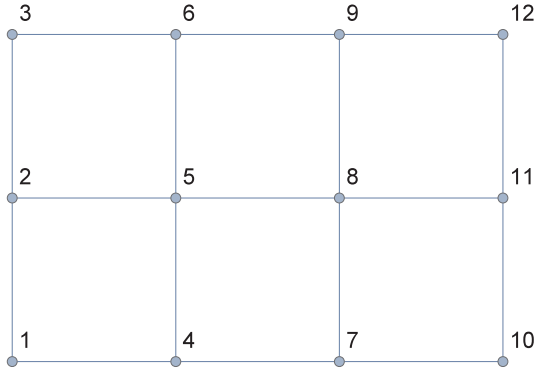

Fig. 2. A grid graph $G_{3\times 4}(V, E)$.

in Fig. 1. In addition, α_i^a for any parameter or variable α, and their corresponding names with bars for the opposite direction on an artery are considered.

The model takes into account only networks that can be represented by two dimensional grid graphs $G_{r\times c}(V, E)$, where r and c are the number of rows and columns, respectively. V is the set of nodes, E is the set of edges, $|V| = n = rc$ and $|E| = m = 2rc - r - c$. Grid graphs are a good representation of many real-world networks. Fig. 2 shows an example with $n = 12$ and $m = 17$.

2.1. Objective function

In our practice, the goal is to maximize the weighted sum of the bandwidths. Let A be the set of arteries on the network; thus, the objective function to be maximized is:

$$\sum_{a\in A}\left(k_a b_a + \overline{k}_a \overline{b}_a\right).$$

Here, k_a and $\overline{k}_a$ are the weights for the outbound and inbound bandwidth, respectively.

2.2. Arterial constraints

As mentioned beforalgorithme, all signals are assumed to work into a common signal period which length is introduced in the model as a decision variable. They must lie on an interval $[T_1, T_2]$, see (2). The decision variable z is the reciprocal of the period length, that is, $z = 1/T$. The inequalities (3)-(4) guarantee that the bandwidth remains within the green time. The velocities v_i^a between each signal on each artery are decision variables that are bounded in between e_i^a and f_i^a, representing the lower and upper limits, and d_i^a is the distance between two consecutive arteries, see (6)-(7). In order to avoid sudden changes in the velocities between consecutive signals, they are limited by imposing lower and upper bounds $1/h_i^a$ and $1/g_i^a$ on changes in reciprocal velocities. The reason for using reciprocal bounds for velocities and changes in velocities is that linear constraints can be formulated in this manner. It is not possible to consider directly the inequalities $e_i^a \le v_i^a \le f_i^a$ because the period length is also a variable. Thus, if v_i^a is converted from meters/second to meters/period, the inequality becomes $e_i^a T \le v_i^a T \le f_i^a T$, which resembles nonlinear constraints. Therefore, we use the reciprocals of e_i^a and f_i^a,

$$e_i^a \le v_i^a \le f_i^a \quad \rightarrow \quad \frac{d_i^a}{f_i^a} \le \frac{d_i^a}{v_i^a} \le \frac{d_i^a}{e_i^a} \quad \rightarrow \quad \frac{d_i^a}{f_i^a} z \le t_i^a \le \frac{d_i^a}{e_i^a} z.$$

The same applies to the changes in the velocities.

Additionally, Fig. 1 denotes that $Time_{\text{A-B}} = \Delta_{ai}$+integer number of periods+ϕ_{ij}^a and that $Time_{\text{A-B}}$ =integer number of periods−$\overline{\phi}_{ij}^a$+integer number of periods+Δ_{aj}. Consequently,

$$\phi_{ij}^a + \overline{\phi}_{ij}^a + \Delta_{ai} - \Delta_{aj} = m_{ij}^a, \tag{15}$$

where m_{ij}^a is an integer decision variable (number of periods). In addition, $Time_{\text{C-D}} = \phi_{ij}^a + \frac{1}{2}r_{aj} + w_{aj} + \tau_{aj} = \frac{1}{2}r_{ai} + w_{ai} + t_{ij}^a$ and $Time_{\text{E-F}} = \overline{\phi}_{ij}^a + \frac{1}{2}\overline{r}_{aj} + \overline{w}_{aj} = \frac{1}{2}\overline{r}_{ai} + \overline{w}_{ai} - \overline{\tau}_{ai} + \overline{t}_{ij}^a$. So, if (15) is substituted,

$$\begin{aligned}t_{ij}^a + \overline{t}_{ij}^a + \frac{1}{2}(r_{ai} + \overline{r}_{ai}) + (w_{ai} + \overline{w}_{ai}) - \frac{1}{2}(r_{aj} + \overline{r}_{aj}) - (w_{aj} + \overline{w}_{aj}) - (\tau_{aj}\\ + \overline{\tau}_{ai}) + (\Delta_{ai} - \Delta_{aj}) = m_{ij}^a.\end{aligned} \tag{16}$$

Eq. (16) is named *arterial loop constraint* for artery a between signals S_{ai} and S_{aj}.

In addition, there are constraints that model left turn decisions if they are allowed by the green lights. The MAXBAND permits to decide among four possible patterns of left turns which are resembled in Fig. 3 (for more details please refer to Little et al. [8]).

Parameters ℓ_{ai} and $\overline{\ell}_{ai}$ in Fig. 3 represent, for a signal i on an artery a, the time (periods) of outbound and inbound left turn phases respectively. R is the common red time. For instance, Fig. 4 shows the three possible movements for vehicles on a main street in three different moments. As can be seen at area 2, the traffic lights are green for outbound and inbound directions, so no car in the horizontal street can cross to the other side. On the common red time R a possible different left turn pattern can be given for cross street.

Furthermore, Δ_{ai} can be expressed as a function of ℓ_{ai} and $\overline{\ell}_{ai}$. For example, if we consider Pattern 1 and calculate the difference between the center of total red time of outbound and the total red of inbound (in that order), the following is obtained

$$\Delta_{ai} = \frac{\overline{\ell}_{ai} + R}{2} - \left(\frac{R + \ell_{ai}}{2} + \overline{\ell}_{ai}\right) = -\frac{\ell_{ai} + \overline{\ell}_{ai}}{2}.$$

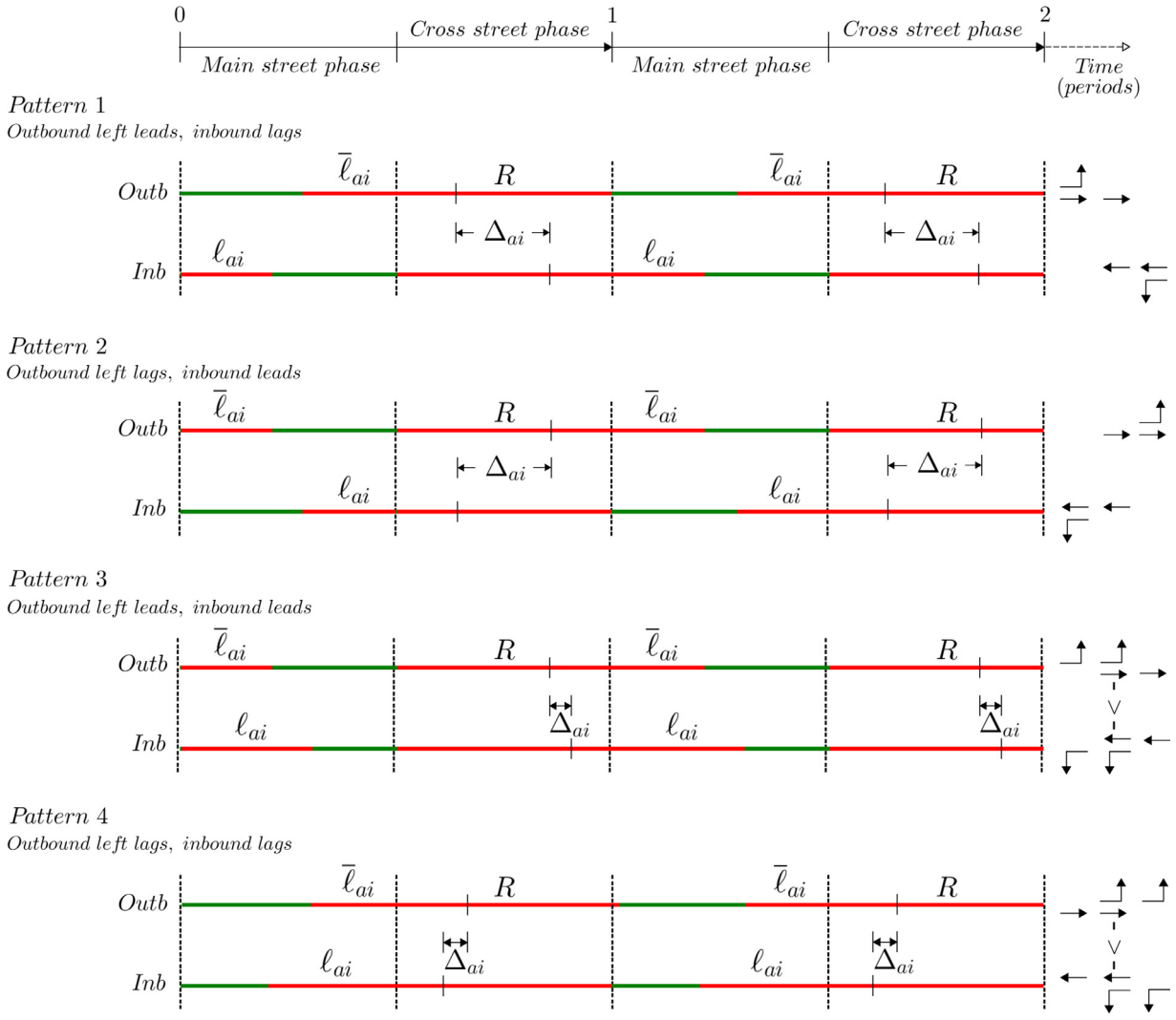

Fig. 3. Patterns of left turn phases.

Table 1
Expressions for Δ_{ai}'s.

Pattern	Δ_{ai}	δ_{ai}	$\bar{\delta}_{ai}$
1	$-\frac{\ell_{ai}+\bar{\ell}_{ai}}{2}$	0	1
2	$\frac{\ell_{ai}+\bar{\ell}_{ai}}{2}$	1	0
3	$-\frac{\ell_{ai}-\bar{\ell}_{ai}}{2}$	0	0
4	$\frac{\ell_{ai}-\bar{\ell}_{ai}}{2}$	1	1

The results for the other left turn phases are listed in Table 1. These expressions can be obtained with the following formula

$$\Delta_{ai} = \frac{1}{2}[(2\delta_{ai}-1)\ell_{ai} - (2\bar{\delta}_{ai}-1)\bar{\ell}_{ai}], \tag{17}$$

where, δ_{ai}, $\bar{\delta}_{ai} \in \{0, 1\}$ are additional binary variables. The decisions on left turns are included in the model by substituting Eq. (16) in (17), as can be seen in the constraint (5).

2.3. Loop constraints

The network case is a natural generalization of the arterial case, and the corresponding model includes all the aforementioned constraints for each artery. The arterial loop constraint (16) can be seen as a cycle for two nodes because it represents the movement of going to and returning from a signal. If this idea is extended to larger cycles, it is clear that the sum of all the offsets in the cycle must be an integer number as well.

For observing how to write the equation of the cycle constraints, let us start with an example. A cycle consisting of 4 arteries $A = \{a, b, c, d\}$ and 4 junctions $J = \{J_1, J_2, J_3, J_4\}$ is exhibited in Fig. 5. Each artery $a \in A$ has signals S_{aj}, where j is the index for signals on a increasing in the outbound direction given by the arrows. Heading in the clockwise direction and starting from junction J_1, the cycle constraint for this example would be

$$\phi^b_{jk} + \Psi^{J_2}_{S_{bk},S_{co}} + \phi^c_{op} + \Psi^{J_3}_{S_{cp},S_{dq}} + \phi^d_{qr} + \Psi^{J_4}_{S_{dr},S_{ah}} + \phi^a_{hi} + \Psi^{J_1}_{S_{ai},S_{bj}} = C_\zeta,$$

where C_ζ is an integer decision variable and $\Psi^i_{S_{aj},S_{bk}}$ is a decision variable expressed as *intranode offset* which represents the time between consecutive centers of reds for signals S_{aj} and S_{bk} that meet at junction i, i.e, it is a link time between arteries a and b.

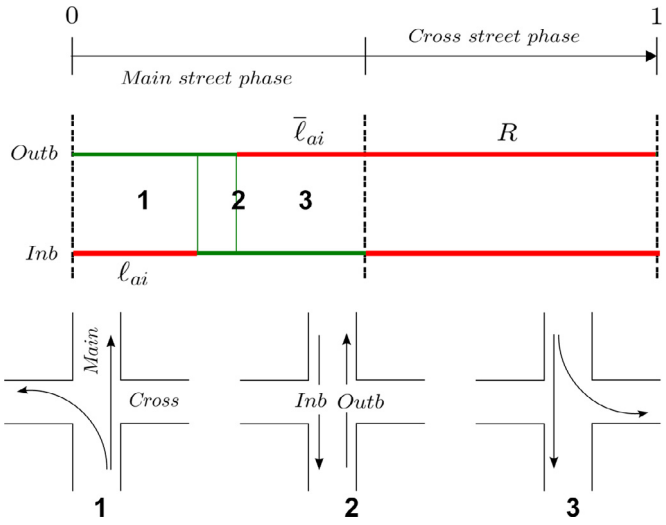

Fig. 4. Left turn phase example with Pattern 1.

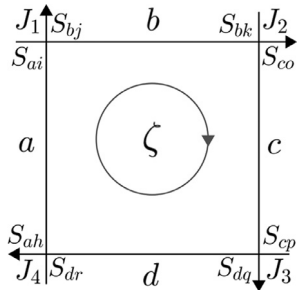

Fig. 5. Clockwise loop with 4 junctions.

To generalize this expression for any cycle, the following sets are defined:

- A^F_ζ (A^B_ζ): Set of all segments of forward (backward) arteries with edges (i, j) in the clockwise direction of cycle ζ,
- J_ζ: All sets of the form (b, j, i, c, k) in ζ, where i is the junction between arteries b and c in the signals S_{bj} and S_{ck},

Afterwards, the *network loop constraint* (cycle constraint) becomes:

$$\sum_{(i,j):a\in A^F_\zeta} \phi^a_{ij} - \sum_{(i,j):a\in A^B_\zeta} \phi^a_{ij} + \sum_{(b,j,i,c,k)\in J_\zeta} \Psi^i_{S_{bj},S_{ck}} = C_\zeta.$$

The number of cycle constraints in the model depend on the quantity of edges and nodes of the network. In fact, this number can be very high, which makes the problem very difficult to solve. The following result permits to alleviate the mentioned problem. It is well known that the set of all cycles ζ on any single graph can be spanned by a basis $\mathcal{B}_\zeta$ with cardinality $m - n + 1$, where m represents the number of edges and n the number of nodes on the underlying undirected graph related to the directed graph, which represents the original network. So, a cycle basis must be found before writing the model. The interested reader is encouraged to consult [29] for more details.

2.4. Computing intranode offset

In general, a grid graph $G_{k\times k}$ requires $(k-1)^2$ network loop constraints and, consequently, several intranodes must be computed for each of these equations. The values of intranode offsets $\Psi^i_{S_{aj},S_{bk}}$'s depend on red time positions of the signals S_{aj} and S_{bk} on a main and cross street, respectively. For instance, if left turns are not permitted, then the red and green times for main and cross streets will have the same length and in this case the intradone offset will be clearly 0.5 periods. Because

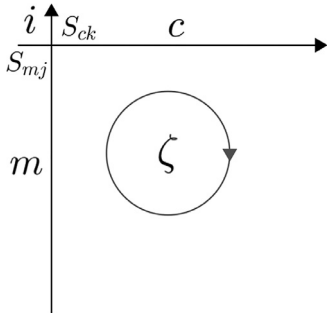

Fig. 6. A junction i of arteries main (m) and cross (c).

Table 2
Expressions for $\psi^{p_m p_c}_{mc}$'s.

		Cross Street			
	Patterns	1	2	3	4
Main Street	1	$\frac{1-\overline{\ell}_{ck}+\overline{\ell}_{mj}}{2}$	$\frac{1+\overline{\ell}_{ck}+\overline{\ell}_{mj}}{2}$	$\frac{1+\overline{\ell}_{ck}+\overline{\ell}_{mj}}{2}$	$\frac{1-\overline{\ell}_{ck}+\overline{\ell}_{mj}}{2}$
	2	$\frac{1-\overline{\ell}_{ck}-\overline{\ell}_{mj}}{2}$	$\frac{1+\overline{\ell}_{ck}-\overline{\ell}_{mj}}{2}$	$\frac{1+\overline{\ell}_{ck}-\overline{\ell}_{mj}}{2}$	$\frac{1-\overline{\ell}_{ck}-\overline{\ell}_{mj}}{2}$
	3	$\frac{1-\overline{\ell}_{ck}-\overline{\ell}_{mj}}{2}$	$\frac{1+\overline{\ell}_{ck}-\overline{\ell}_{mj}}{2}$	$\frac{1+\overline{\ell}_{ck}-\overline{\ell}_{mj}}{2}$	$\frac{1-\overline{\ell}_{ck}-\overline{\ell}_{mj}}{2}$
	4	$\frac{1-\overline{\ell}_{ck}+\overline{\ell}_{mj}}{2}$	$\frac{1+\overline{\ell}_{ck}+\overline{\ell}_{mj}}{2}$	$\frac{1+\overline{\ell}_{ck}+\overline{\ell}_{mj}}{2}$	$\frac{1-\overline{\ell}_{ck}+\overline{\ell}_{mj}}{2}$

the MAXBAND model must decide among four different left turn patterns on each junction i, the computation of each intranode should take into account all the possible values of binary variables involved in such choice. The following result provides a simple expression to compute intranode offsets without requiring extra variables beyond those included in the model LM 2.1.

Theorem 1. *Consider the patterns of left turn phases exhibited in Fig. 3. Let $\psi^{p_m p_c}_{mc}$ be the value of $\Psi^i_{S_{mj},S_{ck}}$ when the arteries m and c meet at the junction i for signals S_{mj} and S_{ck} with left turn phases patterns p_m and p_c, respectively (See Fig. 6). Then, for all possible values of p_m and p_c, it is defined:*

$$\Psi^i_{S_m,S_{ck}} = \frac{1}{2} - \frac{1}{2}\left[(2\overline{\delta}_{ck}-1)\overline{\ell}_{ck} - (2\overline{\delta}_{mj}-1)\overline{\ell}_{mj}\right].$$

Proof. Let us consider Fig. 7, which shows the different forms that $\psi^{p_m p_c}_{mc}$ may acquire for all possible permutations of left turn phases in Fig. 3. The cross street phase takes place during the red time R_m in the main street and vice versa.

All values of $\psi^{p_m p_c}_{mc}$ are summarized in Table 2. Since only the ϕ's in equation (10) of the MAXBAND model are being used, only the outbound directions phases are taken into account.

Moreover, Table 2 reveals that:

$$\psi^{1,1}_{mc} = \psi^{1,4}_{mc} = \psi^{4,1}_{mc} = \psi^{4,4}_{mc} = \frac{1-\overline{\ell}_{ck}+\overline{\ell}_{mj}}{2}, \quad \psi^{2,1}_{mc} = \psi^{2,4}_{mc} = \psi^{3,1}_{mc} = \psi^{3,4}_{mc} = \frac{1-\overline{\ell}_{ck}-\overline{\ell}_{mj}}{2}, \quad \psi^{1,2}_{mc} = \psi^{1,3}_{mc} = \psi^{4,2}_{mc} = \psi^{4,3}_{mc} = \frac{1+\overline{\ell}_{ck}+\overline{\ell}_{mj}}{2}, \quad \text{and}$$

$$\psi^{2,2}_{mc} = \psi^{2,3}_{mc} = \psi^{3,2}_{mc} = \psi^{3,3}_{mc} = \frac{1+\overline{\ell}_{ck}-\overline{\ell}_{mj}}{2}.$$

Table 3 shows all the different values of the binaries variables δ's and $\overline{\delta}$'s for each left turn phase that the model needs to compute Δ's with Eq. (17). The $\psi^{p_m p_c}_{mc}$'s are arranged in four groups determined just for the values of $\overline{\delta}$'s on the signals S_{mj} and S_{ck}.

Now, it is easy to verify that a single expression to compute any $\Psi^i_{S_m,S_{ck}}$ is given by:

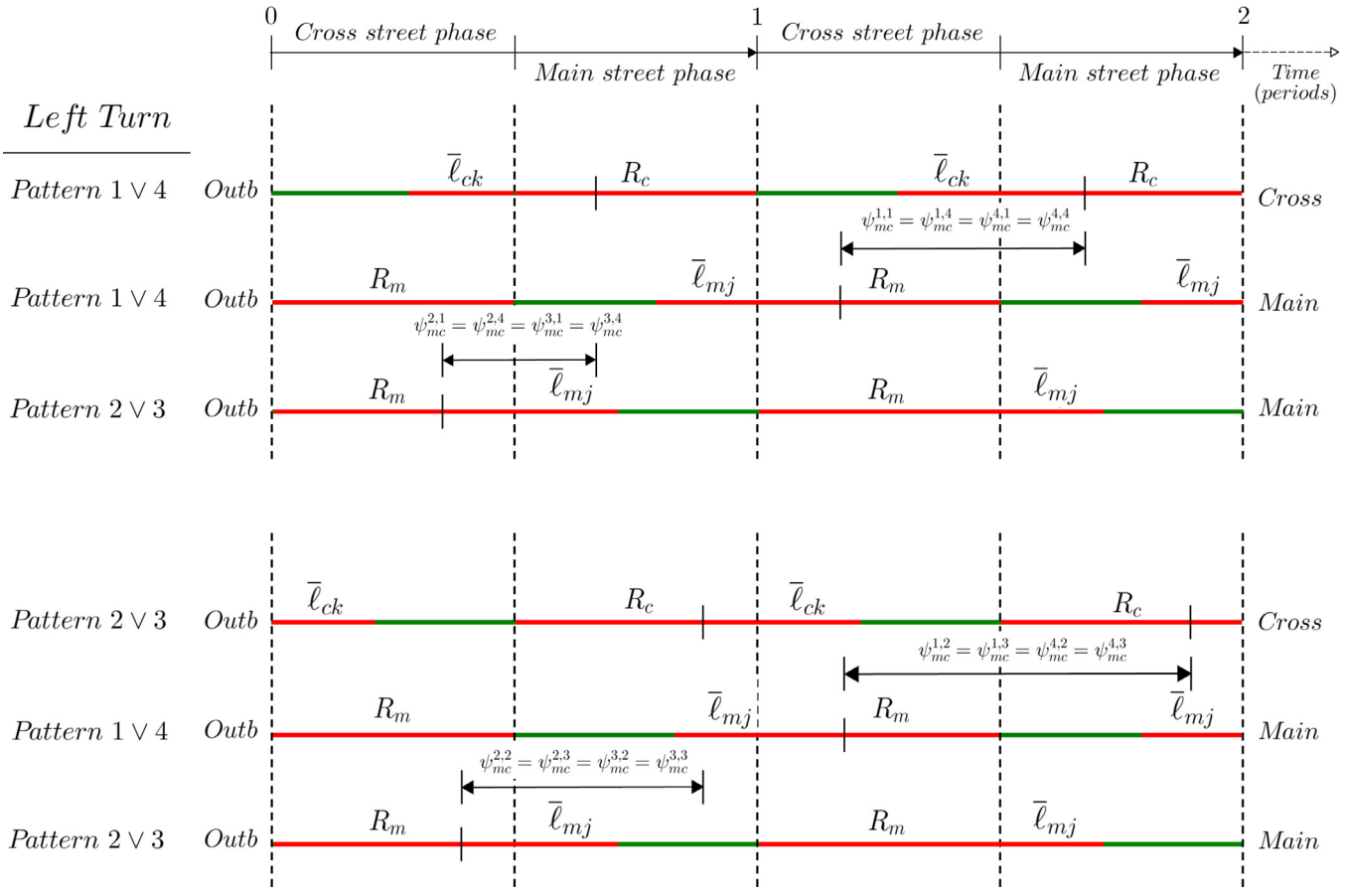

Fig. 7. Geometry for $\psi^{p_m p_c}_{mc}$.

Table 3
Four different groups of $\psi^{p_m p_c}_{mc}$'s on junction i.

Patterns						
p_m	p_c	δ_{mj}	$\overline{\delta}_{mj}$	δ_{ck}	$\overline{\delta}_{ck}$	$\psi^{p_m p_c}_{mc}$
4	4	1	1	1	1	$\psi^{4,4}_{mc}$
4	1	1	1	0	1	$\psi^{4,1}_{mc}$
1	4	0	1	1	1	$\psi^{1,4}_{mc}$
1	1	0	1	0	1	$\psi^{1,1}_{mc}$
3	4	0	0	1	1	$\psi^{3,4}_{mc}$
2	1	1	0	0	1	$\psi^{2,1}_{mc}$
3	1	0	0	0	1	$\psi^{3,1}_{mc}$
2	4	1	0	1	1	$\psi^{2,4}_{mc}$
1	2	0	1	1	0	$\psi^{1,2}_{mc}$
4	2	1	1	1	0	$\psi^{4,2}_{mc}$
4	3	1	1	0	0	$\psi^{4,3}_{mc}$
1	3	0	1	0	0	$\psi^{1,3}_{mc}$
2	2	1	0	1	0	$\psi^{2,2}_{mc}$
2	3	1	0	0	0	$\psi^{2,3}_{mc}$
3	2	0	0	1	0	$\psi^{3,2}_{mc}$
3	3	0	0	0	0	$\psi^{3,3}_{mc}$

$$\Psi^{i}_{S_m,S_{ck}} = \frac{1}{2} - \frac{1}{2}\left[(2\overline{\delta}_{ck} - 1)\overline{\ell}_{ck} - (2\overline{\delta}_{mj} - 1)\overline{\ell}_{mj}\right]. \tag{18}$$

□

A comparable proposition can be found in [30]. However, no attempt to expand it was made.

3. Bounds for integer variables

In this section, the bounds for the integer variables in the arterial loop constraints when left turns phases, given in [7], are extended for the network loop constraints.

From Fig. 1, clearly $0 \le w_{ai} \le 1 - r_{ai}$, and $0 \le \overline{w}_{ai} \le 1 - \overline{r}_{ai}$ for any signal i on artery a. By using the constraints (2), (6) and (7), it is obtained $\frac{d^a_{ij}}{f^a_{ij}T_2} \le t^a_{ij} \le \frac{d^a_{ij}}{e^a_{ij}T_1}$, and $\frac{d^a_{ij}}{\overline{f}^a_{ij}T_2} \le \overline{t}^a_{ij} \le \frac{d^a_{ij}}{\overline{e}^a_{ij}T_1}$. Eq. (17) provides $\Delta_{ai} = \delta_{ai}\ell_{ai} - \frac{\ell_{ai}}{2} - \overline{\delta}_{ai}\overline{\ell}_{ai} - \frac{\overline{\ell}_{ai}}{2}$. Thus, $-(\frac{\ell_{ai}}{2} + \frac{\overline{\ell}_{ai}}{2}) \le \Delta_{ai} \le (\frac{\ell_{ai}}{2} + \frac{\overline{\ell}_{ai}}{2})$, as δ's take values in {0, 1}.

By using the bounds in constraints (5), the following limits for m's in terms of the MAXBAND parameters, $\underline{m^a_{ij}} \le m^a_{ij} \le \overline{m^a_{ij}}$, can be expressed.

$$\overline{m^a_{ij}} = \left\lfloor 2 - \frac{1}{2}(r_{ai} + \overline{r}_{ai}) - \frac{1}{2}(r_{aj} + \overline{r}_{aj}) + \frac{1}{2}(l_{ai} + \overline{l}_{ai}) + \frac{1}{2}(l_{aj} + \overline{l}_{aj}) - (\tau_{aj} + \overline{\tau}_{ai}) + \frac{d^a_{ij}}{e^a_{ij}T_1} + \frac{d^a_{ij}}{\overline{e}^a_{ij}T_1} \right\rfloor, \tag{19}$$

$$\underline{m^a_{ij}} = \left\lceil -2 + \frac{1}{2}(r_{ai} + \overline{r}_{ai}) + \frac{1}{2}(r_{aj} + \overline{r}_{aj}) - \frac{1}{2}(\ell_{ai} + \overline{\ell}_{ai}) - \frac{1}{2}(\ell_{aj} + \overline{\ell}_{aj}) - (\tau_{aj} + \overline{\tau}_{ai}) + \frac{d^a_{ij}}{f^a_{ij}T_2} + \frac{d^a_{ij}}{\overline{f}^a_{ij}T_2} \right\rceil. \tag{20}$$

Therefore, using the notation in LM 2.1, i.e., $m^a_{i,i+1} = m^a_i$:

$$\underline{m^a_i} \le m^a_i \le \overline{m^a_i}, \quad \forall a \in A, \forall i = 1, \dots, n_a - 1. \tag{21}$$

Furthermore, let $a \in A^F_\zeta$ be an artery with signals S_{ai}, where $i \in I^a_\zeta = \{1^a_\zeta, \dots, n^a_\zeta\}$ and increasing in the outbound direction. The time between

Table 4
Sizes of some MAXBAND problems.

	Grid Graph $G_{r\times c}$								
	3x3	5x5	6x6	7x7	8x8	9x9	10x10	15x15	20x20
Equalities	16	56	85	120	161	208	261	616	1121
m's	12	40	60	84	112	144	180	420	760
C's	4	16	25	36	49	64	81	196	361
δ's	36	100	144	196	256	324	400	900	1600
Integer variables	52	156	229	316	417	532	661	1516	2721

the first signal and the last one in the artery segment is:

$$t^a_{1^a_\zeta,n^a_\zeta} = \sum_{i\in I^a_\zeta\setminus\{n^a_\zeta\}} t^a_i - \sum_{i\in I^a_\zeta\setminus/1^a_\zeta/} \tau_{ai}. \tag{22}$$

Hence, $\underline{t^a_{1^a_\zeta,n^a_\zeta}} \le t^a_{1^a_\zeta,n^a_\zeta} \le \overline{t^a_{1^a_\zeta,n^a_\zeta}}$, in which

$$\overline{t^a_{1^a_\zeta,n^a_\zeta}} = \sum_{i\in I^a_\zeta\setminus\{n^a_\zeta\}} \frac{d^a_i}{e^a_i T_1} - \sum_{i\in I^a_\zeta\setminus\{1^a_\zeta\}} \tau_{ai}, \tag{23}$$

$$\underline{t^a_{1^a_\zeta,n^a_\zeta}} = \sum_{i\in I^a_\zeta\setminus\{n^a_\zeta\}} \frac{d^a_i}{f^a_i T_2} - \sum_{i\in I^a_\zeta\setminus\{1^a_\zeta\}} \tau_{ai}. \tag{24}$$

By using the same means mentioned before and considering $\phi^a_{ij} + \frac{1}{2}r_{aj} + w_{aj} + \tau_{aj} = \frac{1}{2}r_{ai} + w_{ai} + t^a_{ij}$, it is obtained $\underline{\phi^a_{i^a_\zeta,n^a_\zeta}} \le \phi^a_{i^a_\zeta,n^a_\zeta} \le \overline{\phi^a_{i^a_\zeta,n^a_\zeta}}$, for,

$$\overline{\phi^a_{i^a_\zeta,n^a_\zeta}} = -\frac{1}{2}(r_{a,1^a_\zeta} + r_{a,n^a_\zeta}) + \overline{t^a_{1^a_\zeta,n^a_\zeta}} + 1, \tag{25}$$

$$\underline{\phi^a_{i^a_\zeta,n^a_\zeta}} = \frac{1}{2}(r_{a,1^a_\zeta} + r_{a,n^a_\zeta}) + \underline{t^a_{1^a_\zeta,n^a_\zeta}} - 1. \tag{26}$$

Following Eq. (18), we also have $\underline{\Psi^i_{bc}} \le \Psi^i_{bc} \le \overline{\Psi^i_{bc}}$, where

$$\overline{\Psi^i_{bc}} = \frac{1}{2}(1 + \overline{\ell}_{ck} + \overline{\ell}_{bj}), \tag{27}$$

$$\underline{\Psi^i_{bc}} = \frac{1}{2}(1 - \overline{\ell}_{ck} - \overline{\ell}_{bj}). \tag{28}$$

Finally, the bounds for C_ζ, in the set of constrains (10), become

$$\overline{C_\zeta} = \left\lfloor \sum_{a\in A^F_\zeta} \overline{\phi^a_{i^a_\zeta,n^a_\zeta}} - \sum_{a\in A^B_\zeta} \underline{\phi^a_{i^a_\zeta,n^a_\zeta}} + \sum_{(b,i,c)\in J_\zeta} \overline{\Psi^i_{bc}} \right\rfloor, \tag{29}$$

$$\underline{C_\zeta} = \left\lceil \sum_{a\in A^F_\zeta} \underline{\phi^a_{i^a_\zeta,n^a_\zeta}} - \sum_{a\in A^B_\zeta} \overline{\phi^a_{i^a_\zeta,n^a_\zeta}} + \sum_{(b,i,c)\in J_\zeta} \underline{\Psi^i_{bc}} \right\rceil. \tag{30}$$

The next set of constraints can be added to the linear model:

$$\underline{C_\zeta} \le C_\zeta \le \overline{C_\zeta}, \quad \forall\zeta \in \mathcal{B}_\zeta. \tag{31}$$

For reducing the space of search of the integer variables, the above-mentioned limits are utilized in the coming computational experiments.

4. Hybrid heuristic for MAXBAND

4.1. A MILP-based heuristic with tabu search for MAXBAND

On a network, the MAXBAND model requires a large number of initial data points as well as several variables defined on each segment of an artery, signals and cycles in the cycle basis. Even though the instances presented in Table 4 may not seem to be very large, the formulation has many equalities which contain integer and binary variables, making the problem difficult to solve.

Because the proposed algorithm includes Tabu Search [23], it requires an initial solution for later exploring its neighbourhood. Despite many attempts to find a systematic procedure to generate this solution, due to the high number of equalities involved, the optimization solver XPRESS provided the first feasible solution [31].

Moreover, the set of the variables m's ($2rc - r - c$), δ's ($4rc$) and C's ($r(c-1) - c + 1$) require to take integer values in the optimal solution. A number of *rm, rd* and *rC* variables are chosen randomly from them respectively to be modified later with one of the following procedures. The rest of the integer variables are fixed to the values they have in the initial solution.

- TSILP-LSF: The values of *rm, rd* and *rC* variables are fixed to values within the inequality Eqs. (21), {0, 1} and (31), respectively.
- TSILP-LSU: The *rm, rd* and *rC* variables with values given by a solution become variables again (i.e., unfixed).
- TSILP-LSVNS: TSILP-LSF and TSILP-LSU are applied one after the other.

The current problem is solved using XPRESS to attain a new feasible neighbour solution. This is repeated to generate a set of solutions (*candidate list*) of size *SizeList*. The list of candidates may contain many unfeasible solutions when using TSILP-LSF. Thus, if some variables are unfixed, the number of unfeasible solutions reduces considerably. In this sense, TSILP-LSU generates the candidate list. In our experience, if a memory structure is applied, releasing (unfixing) variables to solve the problem, it generates a diverse set of solutions. TSILP-LSU is applied on the selected variables *rm, rd* and *rC* only if it is not forbidden by a *tabu list*. The list is an array which contains *tt* values for each variable representing the number of iterations that cannot be modified. This is sorted by decreasing in order the objective value (i.e., the best first), and the first solution becomes the *current solution*. After saving the *tabu list* and updating each integer variable, the *tt* values decrease if different from zero, otherwise they are fixed to a value *maxtt*. Subsequently, a greedy local search is applied to the *current solution*. This is simply an iterative application of TSILP-LSVNS (Algorithm 1), TSILP-LSF (Algorithm 2) or TSILP-LSU (Algorithm 3). The termination criteria is met when a maximum number of iterations occur (see Algorithm 4).

4.2. Computational results

The algorithms performance is tested using artificial data. All random data hold the same values for outbound and inbound directions. Notice that even small grid graphs are very dense.

Let $U(a, b)$ be a continuous uniform distribution on interval (a, b),

- The lengths of the arcs of the grid follow a distribution $U(140, 600)$ (meters).
- Red times r follow a distribution $U(0.4, 0.6)$ (periods).
- Times to turn left ℓ follow a distribution $U(0.25r, 0.38r)$ (seconds).
- Min/max common period $T_{\min}/T_{\max}$, follows a distribution $U(40, 60)/U(90, 110)$ (seconds).
- Limits of velocities lower/upper e/f follow a distribution $U(12, 14)/U(15, 16)$ (meters/second).
- Limits on changes in reciprocal speed lower/upper $1/h/1/g$ = $0.012/-0.012$ (meters/second)$^{-1}$.

Table 5
Computational Results for TSILP procedure (small instances).

		Exact		Global			Global/LS				TSILP-LSF				TSILP-LSU				TSILP-LSVNS			
size	#	OF*	t	iter	sl	tt	iLS	rm	rd	rC	avg	worst	best	avgt	avg	worst	best	avgt	avg	worst	best	avgt
3x3	1	3.22	0	10	5	3	5	2	2	2	2.73	2.71	2.91	6	3.01	3.01	3.02	7	3.00	2.83	3.02	9
				10	5	3	10	2	2	2	2.72	2.71	2.81	8	3.01	3.01	3.02	10	3.02	3.01	3.02	15
				30	10	3	10	4	4	4	2.71	2.71	2.76	33	3.02	3.02	3.02	45	3.02	3.02	3.02	59
				50	10	3	20	4	4	4	2.71	2.71	2.71	73	3.02	3.02	3.02	123	3.02	3.02	3.02	174
	2	3.80	0	10	5	3	5	2	2	2	2.61	1.24	3.76	5	3.48	1.38	3.80	6	3.79	3.67	3.80	8
				10	5	3	10	2	2	2	3.24	1.24	3.77	10	3.78	3.72	3.80	9	3.80	3.76	3.80	13
				30	10	3	10	4	4	4	3.69	3.69	3.69	31	3.69	3.69	3.69	39	3.69	3.69	3.69	52
				50	10	3	20	4	4	4	3.69	3.69	3.69	72	3.69	3.69	3.69	102	3.69	3.69	3.69	145
5x5	3	4.77	7	10	5	3	5	2	2	2	3.57	2.63	4.16	11	4.05	3.43	4.31	14	4.11	3.73	4.29	17
				10	5	3	10	2	2	2	3.86	3.29	4.06	14	4.08	3.52	4.30	17	4.17	3.83	4.35	19
				30	10	3	10	4	4	4	4.16	4.01	4.41	55	4.31	4.18	4.73	96	4.36	4.18	4.73	117
				50	10	3	20	4	4	4	4.14	3.94	4.18	145	4.33	3.95	4.71	257	4.42	4.23	4.72	332
	4	5.20	6	10	5	3	5	2	2	2	3.88	3.75	3.98	7	4.09	3.84	4.43	11	4.24	3.98	4.49	11
				10	5	3	10	2	2	2	3.91	3.74	4.11	14	4.25	3.78	4.77	22	4.21	4.11	4.26	29
				30	10	3	10	4	4	4	4.47	3.97	4.61	58	4.77	4.66	4.86	84	4.74	4.46	4.86	120
				50	10	3	20	4	4	4	4.45	3.97	4.61	145	4.78	4.66	4.86	260	4.83	4.77	4.86	326
6x6	5	5.18	116	10	5	3	5	2	2	2	3.11	2.01	3.52	15	3.51	3.28	3.94	17	3.54	3.18	3.97	20
				10	5	3	10	2	2	2	3.12	2.49	3.48	21	3.49	3.07	3.79	33	3.48	3.07	3.86	39
				30	10	3	10	4	4	4	3.58	3.38	3.73	85	4.07	3.57	4.24	128	4.16	3.91	4.24	152
				50	10	3	20	4	4	4	3.73	3.49	4.18	190	4.23	3.93	4.46	345	4.31	4.12	4.46	420
	6	4.74	1270	10	5	3	5	2	2	2	3.35	1.63	4.16	12	4.18	3.92	4.31	17	4.16	3.92	4.31	19
				10	5	3	10	2	2	2	3.57	1.63	4.15	17	4.19	3.74	4.35	25	4.26	4.10	4.35	36
				30	10	3	10	4	4	4	4.24	4.17	4.33	92	4.32	4.14	4.37	128	4.36	4.35	4.37	152
				50	10	3	20	4	4	4	4.28	4.22	4.33	193	4.36	4.35	4.37	274	4.36	4.35	4.37	295

Algorithm 1 VNS Local Search Procedure (LSVNS).

Let:
S : A problem with best objective function value on a *candidate list*.

Step 1. Choose *rm*, *rd* and *rC* from m's, δ's and C's on S.
Step 2. If $tt = 0$, fix their values with random numbers within their bounds.
Step 3. Solve the instance using an LP solver (XPRESS) and set this solution as a *current solution*.
Step 4. Choose other integer variables *rm*, *rd* and *rC* from m's, δ's and C's on the *current solution*.
Step 5. If $tt = 0$, these variables are unfixed.
Step 6. Solve the instance using branch and bound (XPRESS).
Step 7. Update the *current solution* only if it is better than the previous one.
Step 8. Repeat the process until the maximum number of iterations is reached.

Algorithm 2 Fix local search procedure (LSF).

Let:
S : A problem with best objective function value on a *candidate list*.

Steps 1 …3 and 7 …8 are as in LSVNS.

- All τ_{ai}'s and $\overline{\tau}_{ai}$'s were set to 0.
- All weights on the objective function were set to 1.

Computational experiments are carried out using a PC Intel(R) Xeon(R) 3.40GHz 16.0 (RAM). The fundamental cycle basis [29,32] for each graph $G_{r\times c}$ were obtained using *Mathematica* version 10.1. The algorithms were coded with *Xpress Mosel* version 3.4.2, and the solver used was Xpress Optimizer version 24.01.04.

Algorithm 3 Unfix local search procedure (LSU).

Let:
S : A problem with best objective function value on a *candidate list*.

Steps 4 …6 and 7 …8 are as in LSVNS.

Algorithm 4 Tabu Search for MAXBAND (TSILP-LS/F/U/VNS).

Let:
P: A complete MAXBAND problem on a grid graph.

Step 1. For all m, δ and C, $tt = 0$ in a *tabu list*.
Step 2. Find the first integer solution for P by XPRESS and set it as *current solution*.
Step 3. Create a *candidate list* of size *SizeList*:
1. Choose randomly *rm*, *rd* and *rC* from m's, δ's and C's in *current solution*.
2. If $tt = 0$, unfix these variables.
3. Solve the problem using branch and bound (XPRESS).
4. Repeat *SizeList* times.

Step 4. Sort the *candidate list* in decreasing order of the objective function value.
Step 5. Apply procedure LSF, LSU or LSVNS with the best first in the candidate list as input.
Step 6. Set the *current solution* as the best solution in the *candidate list*.
Step 7. Update the *tabu list* for each m, δ and C on the *current solution*:
1. If current $tt = 0$ then $tt = maxtt$, else
2. $tt = tt - 1$.

Step 8. Repeat Steps $3-7$ until a maximum number of iterations is reached.

Table 5 presents results for different small grid graph instances generated randomly, considering all parameters and variables in the model 1 including the bounds (21) and (31).

Table 6
Computational Results for TSILP procedure (large instances).

		Global			Global/LS				TSILP-LSF				TSILP-LSU				TSILP-LSVNS			
size	#	iter	sl	tt	iLS	rm	rd	rC	avg	worst	best	avgt	avg	worst	best	avgt	avg	worst	best	avgt
7x7	7	30	10	5	20	5	5	5	4.33	4.24	4.34	292	4.51	4.35	4.66	484	4.63	4.56	4.68	546
		30	10	5	30	5	5	5	4.34	4.27	4.46	213	4.51	4.35	4.72	440	4.58	4.36	4.72	520
		50	10	5	30	5	5	5	4.34	4.34	4.35	300	4.56	4.35	4.70	625	4.66	4.53	4.72	815
		50	20	5	30	10	10	10	4.37	4.24	4.55	623	4.67	4.56	4.72	1469	4.70	4.66	4.72	1700
	8	30	10	5	20	5	5	5	3.11	2.88	3.19	175	3.28	3.19	3.46	347	3.40	3.19	3.55	417
		30	10	5	30	5	5	5	3.15	2.88	3.19	162	3.31	3.12	3.48	329	3.41	3.19	3.83	425
		50	10	5	30	5	5	5	3.21	3.19	3.38	399	3.35	3.19	3.55	613	3.63	3.23	3.99	817
		50	20	5	30	10	10	10	3.29	3.19	3.82	581	3.52	3.22	4.30	1113	3.81	3.36	4.39	1323
8x8	9	30	10	5	20	5	5	5	4.08	3.85	4.28	219	4.22	4.20	4.28	398	4.23	4.20	4.28	483
		30	10	5	30	5	5	5	3.99	3.85	4.19	281	4.21	4.20	4.24	715	4.24	4.20	4.28	910
		50	10	5	30	5	5	5	4.04	3.85	4.19	472	4.23	4.20	4.33	806	4.25	4.20	4.33	900
		50	20	5	30	10	10	10	4.22	4.19	4.28	778	4.28	4.20	4.33	1710	4.30	4.24	4.33	1760
	10	30	10	5	20	5	5	5	2.25	2.08	2.84	167	3.07	2.54	3.37	397	3.10	2.75	3.53	524
		30	10	5	30	5	5	5	2.25	2.01	2.84	218	3.03	2.38	3.49	601	3.12	2.77	3.49	727
		50	10	5	30	5	5	5	2.33	2.08	2.96	382	3.35	2.84	3.72	710	3.42	2.84	3.61	1186
		50	20	5	30	10	10	10	3.01	2.68	3.22	567	3.54	3.27	3.92	1345	3.72	3.29	4.10	1688
9x9	11	30	10	5	20	5	5	5	3.33	3.00	3.70	282	4.15	3.61	4.35	524	4.15	3.61	4.31	620
		30	10	5	30	5	5	5	3.40	3.00	4.02	324	4.04	3.61	4.31	620	4.20	3.61	4.49	755
		50	10	5	30	5	5	5	3.43	3.00	3.65	522	4.25	3.61	4.88	893	4.93	4.75	5.01	928
		50	20	5	30	10	10	10	4.14	3.79	4.42	479	4.36	4.26	4.49	1131	4.80	4.38	5.20	1206
	12	30	10	5	20	5	5	5	4.17	4.07	4.31	270	4.31	4.24	4.71	541	4.36	4.24	4.71	646
		30	10	5	30	5	5	5	4.18	4.07	4.31	316	4.25	4.22	4.31	699	4.28	4.24	4.42	848
		50	10	5	30	5	5	5	4.18	4.07	4.42	539	4.26	4.24	4.31	901	4.39	4.24	4.74	1223
		50	20	5	30	10	10	10	4.27	4.12	4.42	658	4.43	4.24	4.56	1324	4.45	4.31	4.61	1698
10x10	13	30	10	5	20	5	5	5	1.63	1.55	1.91	9937	1.85	1.76	1.91	10,123	1.86	1.76	1.91	10,259
		30	10	5	30	5	5	5	1.75	1.55	1.91	10,007	1.84	1.57	1.91	10,278	1.89	1.72	1.91	10,488
		50	10	5	30	5	5	5	1.71	1.55	1.91	10,218	1.86	1.66	1.91	10,714	1.91	1.90	1.91	11,058
		50	20	5	30	10	10	10	1.89	1.72	1.91	10,553	1.91	1.91	1.91	11,341	1.94	1.73	2.01	11,729
	14	30	10	5	20	5	5	5	2.36	2.30	2.46	474	2.86	2.56	2.97	778	2.91	2.75	2.97	918
		30	10	5	30	5	5	5	2.45	2.30	2.90	554	2.85	2.52	3.03	769	2.94	2.79	3.05	958
		50	10	5	30	5	5	5	2.40	2.30	2.55	648	2.93	2.84	3.05	1407	2.95	2.79	3.05	2016
		50	20	5	30	10	10	10	2.98	2.77	3.05	876	3.13	3.04	3.21	2082	3.20	3.05	3.24	3238

Table 7
10 × 10 instances vs XPRESS.

		Exact			TSILP_LSF		TSILP_LSU		TSILP_LSVNS	
size	#	f_OF	t_fs	t > 3h	worst	best	worst	best	worst	best
10x10	13	0.14	9657	1.55	1.55	1.91	1.57	1.91	1.72	2.01
	14	0.54	152	2.19	2.30	3.05	2.52	3.21	2.75	3.24

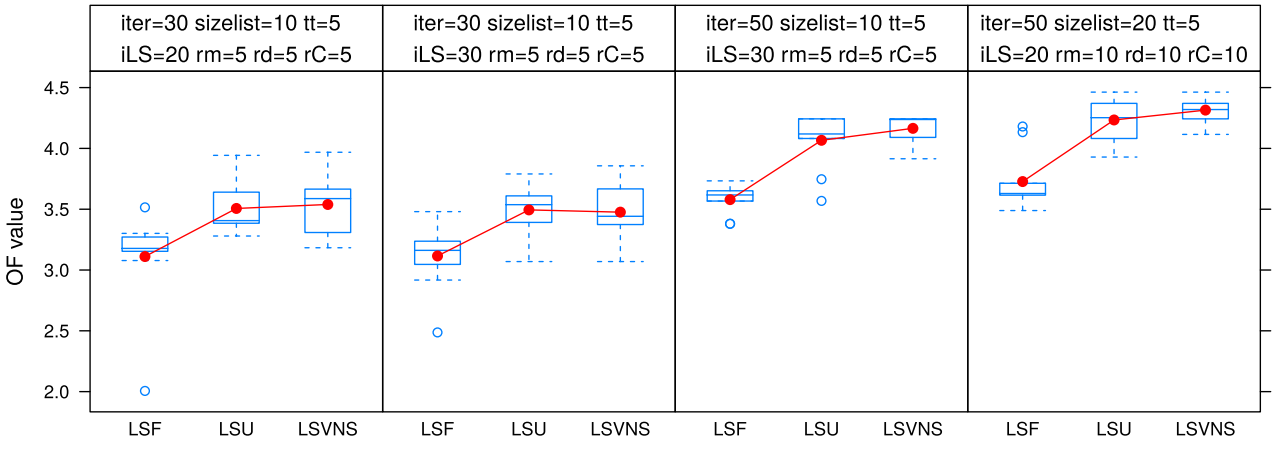

Fig. 8. Box-Plot of 10 runs for experiments TSILP-LS on $G_{6\times 6}$ (instance 5).

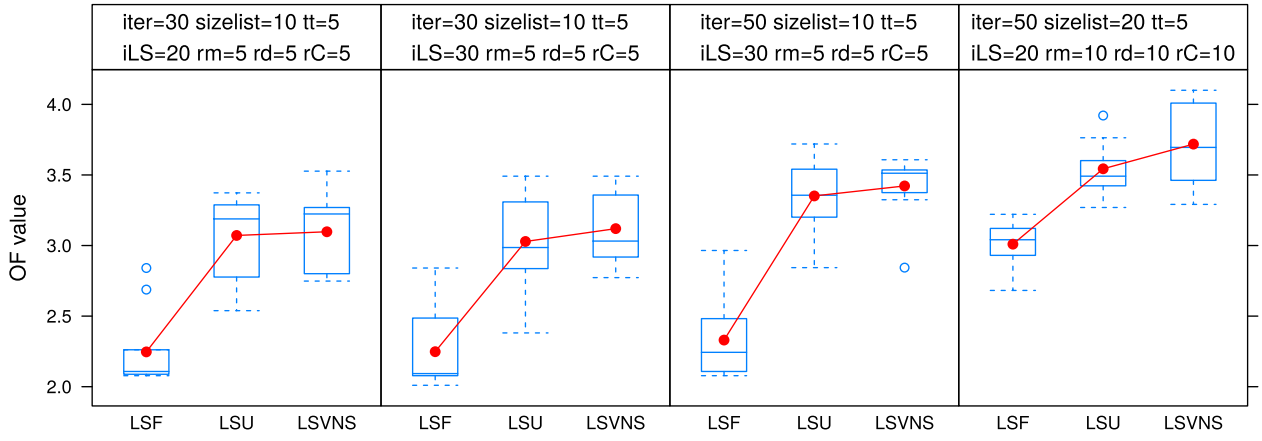

Fig. 9. Box-Plot of 10 runs for experiments TSILP-LS on $G_{8\times 8}$ (instance 10).

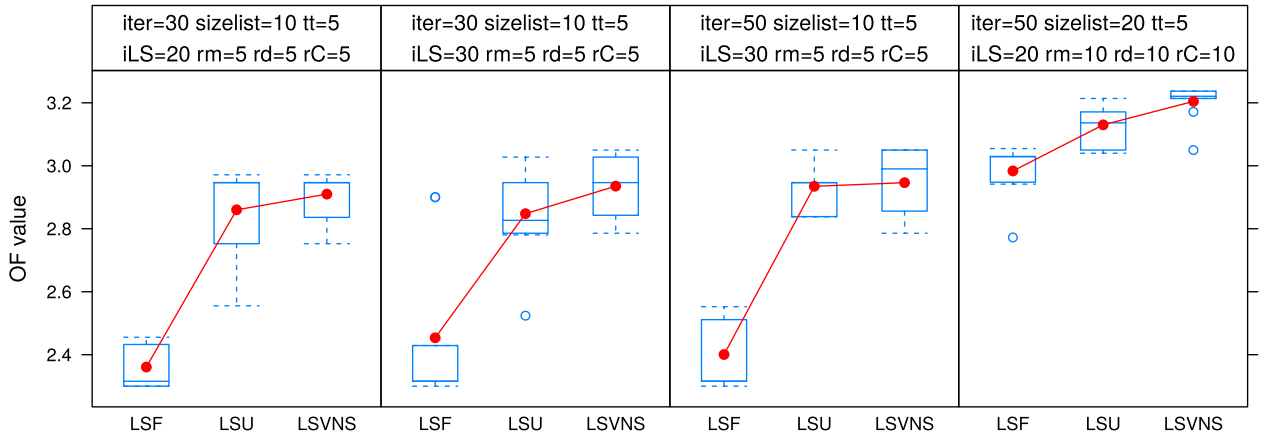

Fig. 10. Box-Plot of 10 runs for experiments TSILP-LS on $G_{10\times 10}$ (instance 14).

The headers are as follows:

- size: size of the problem.
- #: instance number.
- Exact (XPRESS branch and bound).
  - OF*: optimal value of the objective function.
  - t: running time (seconds).
- Global (stands for whole procedure).
  - iter: number of tabu search iterations.
  - sl: size list.
  - tt: tenure time in the memory list (iterations).
- Global/LS (stands for whole and local search procedures).
  - iLS: number of iterations of local search.
  - rm, rd, rC: number of variables m's, δ's and C's chosen for the candidate list and the local search.
- TSILP-(LSF, LSU, LSVNS).
  - avg, worst, best: Average, worst and best case objective function value.
  - avgt: average time (seconds).

Each problem is solved ten times using four different parameter settings. An equal number of integer variables *rm, rd* and *rC* were used in both cases to create the candidate list and to run the local search procedure. The best objective function values obtained among the different heuristic algorithms are underlined. For each instances, XPRESS permmited to obtain optimal solutions in short time.

TSILP-LSU and TSILP-LSVNS met the optimal for problem #2 and both algorithms performed almost the same in most of the evaluated cases. Fig. 8 resembles, as an example, the mean OF values represented in red dots.

For problems of 5×5 and 6×6 size, the increase in the number of iterations, size list and selected random variables generated better results. Clearly, as the problem size increases, the heuristics times are more competitive.

Table 6 presents results for larger instances. In this case, XPRESS was not able to find the optimal solution for any of them within a time limit of 3 h. On the other hand, TSILP-LSVNS showed better results as it is portrayed in Figs. 9 and 10.

The running times, which include the time required to achieve the first initial feasible solution using standard branch and bound in XPRESS, were as explained next. For instance #13, the CPU time was more or less 9657 s, 152 for instance #14, and the remainder took less than 11 s.

In most of the cases, the average solution of TSILP-LSVNS improved the average solutions obtained with the first two methods. TSILP-LSF took less time, but almost never found the best objective function value, which was obtained by the other algorithms.

Due to the vast amount of initial data, and the need of having the integer model ready and before applying any algorithm, requires some time which was not included in the running times. Nevertheless, the largest instance tested had an average of 20.73 s, which is time needed by XPRESS to generate the model to be solved.

The best results was found for the largest instances. In this case, the three algorithms, even in the worst cases, reached much better solution than those obtained using XPRESS. Table 7 lists the first integer solution (f_OF), the time to meet that solution (t_fs) and the best integer solution after 3 h of running time (t > 3h) of the XPRESS branch and bound.

5. Conclusions

In this contribution, a complete review of the MAXBAND model and a generalization of the integer variable bounds proposed in [7] were performed to establish the network case study. Cycle integer variable bounds were provided as well.

Additionally, a hybrid heuristic algorithm based on Tabu Search and VNS took advantage of the mixed integer linear MAXBAND model to obtain feasible solutions. During the search of optimal solutions, the problems were solved in a reduced feasible space due to the initial solution. The proposed algorithm begins with a feasible solution for later obtaining integer values to be used in the subsequent iterations. Results were better when a serial application of LSF and LSU was included by local search using VNS. As attested in the computational experiments, the best results were obtained for large instances.

Future work in this subject embraces to include further attention in prioritizing certain arteries with high traffic, and other real-life traffic aspects that were not included. The incorporation of such real aspect would modify the weights composing the objective function.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Xavier Cabezas acknowledges the support of SENESCYT-Ecuador (National Secretary for Higher Education, Science, Technology and Innovation of Ecuador).

Sergio García acknowledges the support of Fundación Séneca (project 19320/PI/14, Region of Murcia, Spain).

References

[1] M. Péres, G. Ruiz, S. Nesmachnow, A.C. Olivera, Multiobjective evolutionary optimization of traffic flow and pollution in montevideo, uruguay, Appl. Soft Comput. 70 (2018) 472–485.
[2] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, Y. Wang, Review of road traffic control strategies, Proceed. IEEE 91 (12) (2003) 2043–2067.
[3] N. Gartner, D. Little, H. Gabbay, Optimization of traffic signal settings by mixed-integer linear programming; part I: the network coordination problem; part II: The network synchronization problem, Transport. Sci. 9 (1975) 321–363.
[4] G. Wünsch, Coordination of traffic signals in networks, Ph.D. thesis, Technische Universität Berlin, 2008.
[5] E. Köhler, M. Strehler, Traffic signal optimization using cyclically expanded networks, Wiley Period. Netw. 65 (2015) 244–261.
[6] J. Morgan, D. Little, Synchronizing traffic signals for maximal bandwidth, Opera. Res. Spec. Transport. Sci. Issue 12 (6) (1964) 896–912.
[7] D. Little, The synchronization of traffic signals by mixed-integer linear programming, Oper. Res. 14 (4) (1966) 568–594.
[8] D. Little, M. Kelson, N. Gartner, MAXBAND: a versatile program for setting signal on arteries and triangular networks, Transport. Res. Rec. 795 (1980) 40–46.
[9] t Gartner, S. Assmann, F. Lasaga, D. Hou, A multi-band approach to arterial traffic signal, Transport. Res. 25B (1) (1991) 55–74.
[10] C. Zhang, Y. Xie, N. Gartner, C. Stamatiadis, T. Arsava, AM-BAND: an asymmetrical multi-band model for arterial traffic signal coordination, Transport. Res. Part C: Emerg. Technol. 58 (2015) 515–531.
[11] X. Wu, Z. Tian, P. Hu, Z. Yuan, Bandwidth optimization of coordinated arterials based on group partition method, Procedia-Soc. Behav. Sci. 43 (2012) 232–244.
[12] W. Xianyu, H. Peifeng, Y. Zhenzhou, Link-based signalized arterial progression optimization with practical travel speed, J. Appl. Math. (2013).
[13] K. Gao, Y. Zhang, R. Su, F. Yang, P.N. Suganthan, M. Zhou, Solving traffic signal scheduling problems in heterogeneous traffic network by using meta-heuristics, IEEE Trans. Intell. Transport. Syst. (2018).
[14] D. Robertson, TRANSYT, a Traffic Network Study Tool. Technical Report, Crowthorne, Berkshire, 1969.
[15] S. Cohen, Concurrent use of MAXBAND and TRANSYT signal timing programs for arterial signal optimization, Transport. Res. Rec. 906 (1983) 81–84.
[16] t Gartner, C. Stamatiadis, Arterial-based control of traffic flow in urban grid networks, Math. Comput. Model. 35 (2002) 657–671.
[17] R. Braun, F. Weichenmeier, Automatic offline-optimization of coordinated traffic signal control in urban networks using genetic algorithms, in: Proceedings of the 12th World Congress on Intelligent Transport Systems, 2005.
[18] L. Singh, S. Tripathi, H. Arora, Time optimization for traffic signal control using genetic algorithm, Expert Syst. Applica. 2 (2) (2009).
[19] J. García-Nieto, E. Alba, A.C. Olivera, Swarm intelligence for traffic light scheduling: application to real urban areas, Eng. Applica. Artif. Intell. 25 (2) (2012) 274–283.
[20] K. Gao, Y. Zhang, A. Sadollah, R. Su, Optimizing urban traffic light scheduling problem using harmony search with ensemble of local search, Appl. Soft Comput. 48 (2016) 359–372.
[21] H. Yi, Q. Duan, T.W. Liao, Three improved hybrid metaheuristic algorithms for engineering design optimization, Appl. Soft Comput. 13 (5) (2013) 2433–2444.
[22] Q. Duan, T.W. Liao, H. Yi, A comparative study of different local search application strategies in hybrid metaheuristics, Appl. Soft Comput. 13 (3) (2013) 1464–1477.
[23] F. Glover, Future paths for integer programming and links to artificial intelligence, Comput. Opera. Res. 13 (1986) 533–549.
[24] J. Brandao, A. Mercer, A tabu search algorithm for the multi-trip vehicle routing and scheduling problem, Eur. J. Opera. Res. 100 (1) (1997) 180–191.
[25] O. Shahvari, R. Logendran, An enhanced tabu search algorithm to minimize a bi-criteria objective in batching and scheduling problems on unrelated-parallel machines with desired lower bounds on batch sizes, Comput. Opera. Res. 77 (2017) 154–176.
[26] L. Berbotto, S. García, F.J. Nogales, A randomized granular tabu search heuristic for the split delivery vehicle routing problem, Ann. Oper. Res. 222 (1) (2014) 153–173.
[27] P. Demeester, W. Souffriau, P. De Causmaecker, G.V. Berghe, A hybrid tabu search algorithm for automatically assigning patients to beds, Artifi. Intell. Med. 48 (1) (2010) 61–70.
[28] N. Mladenović, P. Hansen, Variable neighbourhood search, Comput. Oper. Res. 24 (1997) 1097–1100.
[29] C. Liebchen, R. Rizzi, Classes of cycle bases, Disc. Appl. Math. 155 (3) (2007) 337–355.
[30] N. Chaudhary, A mixed integer linear programming approach for obtaining an optimal signal timing plan in general traffic networks, Texas A&M University, 1987 Ph.D. thesis.
[31] FICO, Xpress optimization suite, January 19, 2017, Retrieved from https://www.fico.com/en/products/fico-xpress-optimization.
[32] T. Kavitha, C. Liebchen, K. Mehlhorn, D. Michail, R. Rizzi, T. Ueckerdt, K. Zweig, Cycle bases in graphs: characterization, algorithms, complexity and applications, Comput. Sci. 3 (4) (2009) 199–243.