TRANSPORTATION SCIENCE
Vol. 43, No. 1, February 2009, pp. 98–116
ISSN 0041-1655 | EISSN 1526-5447 | 09 | 4301 | 0098

DOI 10.1287/trsc.1080.0254

A Mathematical Logic Approach for the Transformation of the Linear Conditional Piecewise Functions of Dispersion-and-Store and Cell Transmission Traffic Flow Models into Linear Mixed-Integer Form

Yannis Pavlis
Department of Civil Engineering and Institute of Transportation Studies, University of California, Irvine, California 92697, and Centre for Research and Technology Hellas—Hellenic Institute of Transport (CERTH—HIT), 15231 Chalandri-Athens, Greece, ipavlis@certh.gr

Will Recker
Department of Civil Engineering and Institute of Transportation Studies, University of California, Irvine, California 92697, wwrecker@uci.edu

The modeling of traffic control systems for solving such problems as surface street signalization, dynamic traffic assignment, etc., typically results in the appearance of a conditional function. For example, the consistent representation of the outflow discharge at an approach of a signalized intersection implies a function that is conditional on the signal indication and the prevailing traffic conditions. Representing such functions by some sort of constraint(s), ideally linear, so as to be considered in the context of a mathematical programming problem, is a nontrivial task, most often resolved by adopting restrictive assumptions regarding real-life process behavior. To address this general problem, we develop two methodologies that are largely based on analogies from mathematical logic that provide a practical device for the transformation of a specific form of a linear conditional piecewise function into a mixed integer model (MIM), i.e., a set of mixed-integer linear inequality constraints. We show the applicability of these methodologies to transforming into a MIM virtually every possible conditional piecewise function that can be found when one is modeling transportation systems based on the widely adopted dispersion-and-store and cell transmission traffic flow models, as well as to analyzing existing MIMs for identifying and eliminating redundancies.

Key words: mathematical logic; conditional piecewise functions; dispersion-and-store model; cell transmission model; mathematical programming
History: Received: March 2006; revised: December 2006, October 2007; accepted November 2008.

1. Introduction

The modeling of traffic control operations almost inevitably requires specification of conditional functions in the mathematical representation, the most common example being the general relationship describing the outflow discharge at a signalized intersection that is conditional on the signal indication and the prevailing traffic conditions. Within the context of their formulation as a mathematical programming problem, such conditional functions are represented by some sort of constraint(s). Notable instances of such formulations are found in the traffic dynamics described by the so-called store-and-forward—or the similar dispersion-and-store—model, as well as in the cell transmission model. In the following paper we present various approaches for this issue and analyze the process consistency implications.

The store-and-forward model has been adopted for determining the time instant switching of green-red that simultaneously dissipates queues at the end of a rush-hour period during which oversaturated conditions prevail; see, e.g., Singh and Tamura (1974), D'Ans and Gazis (1976). Papageorgiou (1995) provides a generalization in the form of a linear program for mixed-surface street/freeway networks by further considering that: (a) the sample time interval is greater than or equal to the (known) cycle duration and (b) feasible solutions for queues are not determined by traffic flow relationships but rather by signal control variables, e.g., queues on surface streets remain within an upper bound by allowing all-red cycles. However, the formulation ignores oversaturated conditions and determines the outflow discharge by a single constraint, as the product of the saturation flow rate and the green per cycle ratio (i.e., the outflow is independent of the signal indication and the prevailing conditions, which could lead to negative queues). Chang, Wu, and Lieu (1994), and Wu and Chang (1999) develop signal control strategies for mixed networks by manipulating the outflow discharge function, based on the values of the current state and a previously determined control variable, with the solution algorithm assigning the current minimum of the two arguments to the link outflow. A linear problem is generated at each time instant, and the overall solution is determined by heuristically selecting the linear programs to be solved from the various possible combinations, a procedure that, while practical, violates the general principle of optimality.

De Schutter and De Moor (1998) formulate a signal control problem in terms of the green-yellow-red switching time instants. Constant arrival and departure rates are assumed to prevail during the optimization horizon, and the case of a variable cycle with two conflicting movements is considered. In the formulation, the outflow discharge depends only on the signal indication and is described by a pair of linear constraints; consequently, the model determines the maximum between the current queue value and zero to prevent the generation of negative queues in a storage segment while queues are forced inside upper bounds by the control variables (i.e., the model overestimates the stored vehicles). As formulated, the problem is NP-hard and generally intractable for more than seven time–switching instants during the optimization horizon. However, suboptimal solutions are shown to be efficiently found by formulating linear programs under the simplifying assumptions that (a) oversaturated conditions prevail and (b) the duration of yellow is zero.

A more general case for a "storage" type of model occurs when the outflow discharge is an explicit function not only of the upstream density (or occupancy) but also of the downstream spare capacity, so that when the spare capacity at the adjacent downstream link is zero, the outflow discharge is annulled despite a green signal—a phenomenon that is referred to as de facto Red. The importance of this phenomenon in modeling transportation networks is mentioned in Algadhi and Mahmassani (1990), Sheffi, Mahmassani, and Powell (1982), and Chang, Mahmassani, and Herman (1985).

Further instances of the representation of conditional functions as minimum and medium point operators appear in the cell transmission model. In a widely addressed example (see, e.g., Stephanedes and Chang 1993; Ziliaskopoulos 2000), traffic flow at the boundaries of two adjacent cells (segments) is determined as the minimum between the demand of the upstream cell, the supply of the downstream cell, and the flow capacity at the cell boundary (all expressed in flow rate units), i.e., by a function of the form $\omega(\cdot) = \min(\xi, \psi, \zeta)$, which is approximated by the set of relaxed constraints $\omega(\cdot) \leq \xi$, $\omega(\cdot) \leq \psi$, and $\omega(\cdot) \leq \zeta$. Clearly, this constraint set is not an exact transformation of the aforementioned minimum function, because the range of $\omega(\cdot)$ in the latter expressions is not restricted solely to the (discrete) values that the expressions ξ, ψ, or ζ could take (obviously, it could be less than the value of the minimum operator in the former). Indeed, as shown in Ziliaskopoulos (2000), an optimal decision for dynamic traffic assignment could be such that it holds vehicles in cells for some time period (i.e., the corresponding outflow is zero) despite there being spare capacity at the adjacent downstream cell that allows a transfer flow that is at least equal to the minimum of the three upper bounds; this is sometimes referred to as the *vehicle hold back* phenomenon. Neither do attempts to force $\omega(\cdot)$ to fall on the upper boundaries of the set by adding an appropriately weighted term in the objective function (see, e.g., Lin and Wang 2004), guarantee the desired result.

Another example is presented by the merging of outflow from two cells into a single downstream cell. According to the theory, if the receiving downstream cell has sufficient spare capacity, then all traffic from upstream cells enters; otherwise, the spare capacity (supply) is allocated according to the amount of competing traffic at the upstream cells so that as much of the upstream demand as possible moves forward; this is described by a medium point operator. This operator is manipulated for a ramp metering problem by Gomes and Horowitz (2006) as follows. First, the spare capacity is separately allocated in advance to each merging cell (mainline and on-ramp); then, the outflow from the upstream cells is calculated separately by a minimum operator, i.e., by assuming ordinary cell connections. Clearly, this approach could generate a *virtual bottleneck*: The competing demand from an upstream cell could not be satisfied by the corresponding allocated spare capacity while there is enough complementary spare capacity to satisfy not only the corresponding demand but also a portion of the competing demand. To cope with the resulting nonlinear optimal control problem, a suboptimal linear problem is formulated by assuming, additionally, that (a) the minimum operator is relaxed, (b) no on-ramp flows should be restricted by a lack of space on the mainline whenever the freeway is optimally metered, (c) the controller is allowed to completely shut down the on ramps, and (d) the conditions that prevail at the end of the optimization horizon approach an empty freeway. A similar approach regarding the midpoint and minimum operators for the formulation of a dynamic traffic assignment problem is taken by Ukkusuri and Waller (2007).

The aforementioned examples of manipulating conditional functions in the context of a mathematical programming problem reveal the following. First, in modeling traffic control systems, such functions are not associated with unnecessarily detailed aspects of the system; rather, they represent fundamental principles of the traffic process. Second, problems that incorporate such functions are inherently difficult to solve. Third, attempts to simplify the model and problem complexity lead to questionable approaches regarding the mathematical representation of the real-life process attributes and/or the solution procedure for the corresponding problem that restrict the insight of the results. For example, the traffic behavior predicted by modifications of the cell transmission model that allow for vehicle hold back and virtual bottlenecks is similar to that of high-order traffic flow models that can predict backward moving traffic (Daganzo 1995c).

In recognition of these difficulties, we have developed two logic-based methodologies for the transformation of a specific generic form of a conditional piecewise function into a mixed integer model (MIM) that is a set of linear constraints in mixed-integer variables. In the sections following, we present the methodologies and show their applicability by transforming into a MIM virtually every possible conditional function that can be found when one is developing mathematical representations for traffic control systems based on either the dispersion-and-store or the cell transmission traffic flow model. Moreover, the methodologies provide the means for identifying redundant binary variables, 0-1 combinations, and constraints; we provide such examples, including identifying redundancies in a consistent MIM representation for the minimum operator in the cell transmission model described in Lo (2001).

2. Specifying a Theoretical Methodological Framework

Consider a piecewise linear function $y = \mathrm{f}(x)$ defined in the domain $[x_0, x_q]$ containing q branches as shown below:

$$y = \begin{cases} \mathrm{f}_1(x) & \text{when } x \in [x_0, x_1] \\ \mathrm{f}_2(x) & \text{when } x \in [x_1, x_2] \\ \vdots & \quad\vdots \qquad\qquad \vdots \\ \mathrm{f}_q(x) & \text{when } x \in [x_{q-1}, x_q]. \end{cases} \tag{1}$$

Assume that the function is continuous at all bounds between successive domain intervals, i.e., $\mathrm{f}_1(x_1) = \mathrm{f}_2(x_1), \ldots, \mathrm{f}_{q-1}(x_{q-1}) = \mathrm{f}_q(x_{q-1})$. Apparently, Dantzig (1963) was among the first to transform specific examples of a nonconvex linear piecewise function in the form of (1) into an equivalent MIM representation in binary variables as an objective function in a linear programming problem. He also used binary variables for representing by mixed-integer linear models such other nonlinearities as dichotomies (in general, k-fold alternatives), simple conditional constraints (i.e., if a single constraint is satisfied, then another single constraint is implied), and other forms that define either disconnected or nonconvex feasible regions by disjunctive linear constraint sets. Mayer (1981) also developed mixed-integer formulations for linear piecewise functions in the form of (1) that describe economies of scale. Further practical application examples that show how binary (and, in general, integer) variables are used in the equivalent representation of complex nonlinear functions by a MIM for the formulation of such combinatorial optimization problems as resource allocation, machine scheduling, etc., can be found in Ibaraki (1976), Williams (1988, 1995), and Nemhauser and Wolsey (1998).

In modeling conditional functions as MIMs, we observe that they can all be formalized under a generic form. Specifically, function (1) can be viewed as expressing a relationship between some premises and some conclusions. Thus, the premise "$x \in [x_{i-1}, x_i]$" leads to the conclusion "$y = \mathrm{f}_i(x)$," for $i = 1, \ldots, q$. Note that the functional form of (1) is such that (1) the premises express that the decision variable x is bounded by constraints in some range of values, i.e., $x_{i-1} \le x \le x_i$; (2) the conclusions express that the decision variable corresponding to the function range y is equal to a linear expression involving the decision variable x, i.e., $y = \mathrm{f}_i(x)$; and (3) an equivalence relationship ("if and only if") is implied between the premises and the conclusion in all branches. Consider now a more general functional form of (1) that has the following additional properties: (1) The premises may express the division of the feasible range of decision variables other than x, say, x', i.e., $x'_{i-1} \le x' \le x'_i$, and/or logical propositions; (2) the conclusions may express that the decision variable y is equal to a discrete value (real or integer), i.e., $y = \mathrm{f}_i(x_\mathrm{d})$ with $x_\mathrm{d} \in [x_{i-1}, x_i]$ or, in other words, for $x_{i-1} \le x \le x_i$ the decision variable can be either $y = \mathrm{f}_i(x)$ in certain circumstances or $y = \mathrm{f}_i(x_\mathrm{d})$ in other; and (3) the premises and the corresponding conclusion may be connected by an implication ("if-then") relationship. We define the latter functional form as a conditional piecewise function (CPF).

The CPFs that are transformed into an equivalent MIM representation in this paper are conformable to the definition above. This generic functional form allows us to develop two methodological procedures (meta-algorithms) that are a practical device (a set of guidelines) for its transformation into an equivalent form that is a MIM in binary variables. The methodologies are based on fundamental concepts from the sentential calculus of mathematical logic and the corresponding set-theoretic relationships. For the theory of mathematical logic Shoenfield (1967), and for binary variables and functions, Hromkovic (2004) are used as general references.

For methodology, we adopt the axiomatic framework of predicate logic; because the theory of mathematical logic lacks a universal system of terminology and notation, some definitions are necessary. An elementary sentence is simply a sentence that cannot be further analyzed (sometimes also called an atomic proposition). Generally speaking, the elementary sentences of predicate logic can be viewed as functions expressing some conditions that take two values: true (denoted as T) if the conditions are satisfied and false (denoted as F) otherwise. In the context of a mathematical programming problem, their truth or falsehood is reflected in the values of a binary variable. In our case, we distinguish between two elementary sentence types. The first type of elementary sentence is that in which its truth or falsehood does not depend on any conditions; we denote as P_i the i-th elementary sentence of this type. As an example, the sentence "The signal is red" can be either T or F, regardless of any conditions. The second type of elementary sentence is that in which its truth or falsehood depends on conditions and, specifically, on the satisfiability of constraints (inequality or equation); this we denote as $P_j(\cdot)$, corresponding to the j-th elementary sentence. As an example, the sentence that "expresses" the inequality constraint $x \le x_e$, $\forall x \in [x'_e, x''_e]$, with x_e, x'_e, and x''_e any real parameters, corresponding to the j-th elementary sentence, will be denoted as $P_j(x, x \in [x'_e, x''_e]; x_e; x \le x_e)$, i.e., a function of x and x_e that "expresses" $x \le x_e$. The truth or falsehood of $P_j(\cdot)$ depends on the values of the (decision) variable x that satisfy it; it is T for these values of x that are less than or equal to x_e, i.e., in symbolic form: $P_j(x, x \in [x'_e, x''_e]; x_e; x \le x_e) = \mathrm{T}$, $\forall x \in [x'_e, x''_e]$: $x \le x_e$. For notational simplicity, this second type of elementary sentence will be denoted simply as P_j (instead of $P_j(\cdot)$), with the understanding that truth or falsehood depends on the corresponding conditions (as described in the above example).

For any elementary sentence P, the elementary sentence $\bar{P}$ reads "not P" and represents the logical opposite of the sentence P, i.e., a sentence that is true when P is false, and vice versa. The combination $P_1 \wedge P_2$ (the conjunction of P_1 and P_2) reads "P_1 and P_2" and represents a sentence that is true if and only if both P_1 and P_2 are true, and false otherwise. The combination $P_1 \vee P_2$ (the disjunction of P_1 and P_2) reads "P_1 or P_2" and is true if either of the elementary sentences is true, and false otherwise. The sentence $P_1 \mapsto P_2$ reads "if P_1 then P_2" (or, equivalently, "P_1 implies P_2") and is false if and only if P_1 is true and P_2 is false, and true otherwise; its logical equivalence is the sentence $\bar{P}_1 \vee P_2$. The sentence $P_1 \Leftrightarrow P_2$ reads "P_1 if and only if P_2" and is true if and only if either both P_1 and P_2 are true or both false, and false otherwise; its logical equivalence is the sentence $(P_1 \mapsto P_2) \wedge (P_2 \mapsto P_1)$.

In presenting the methodologies, we first describe the way a CPF can be transformed into its literal analogue—a multiargument disjunctive logical sentence that is a tautology, in which each argument states a logical relationship (e.g., implication, etc.) between premise(s) and a conclusion. We next determine a group of truth value combinations that render valid the logical relationship in every argument. The truth-falsehood is reflected in the 0-1 values of appropriate binary variable(s). The two methodologies are differentiated by two different (but equivalent) ways that a logical sentence can be transformed into a MIM—either by first establishing the truth (or falsehood) of the conclusions and then that of the premises (according to their inner logical relationship), or vice versa. Then, the constraints of each argument are appropriately modified so that their activation-relaxation pattern reflects the truth-falsehood of the corresponding elementary sentence. We then check for redundancies both in the number of binary variables and the corresponding 0-1 combinations, as well as in the constraints. Finally, we verify that the resulting MIM is indeed an equivalent representation of the original CPF and determine how appropriate values can be specified for the relaxation parameters.

The MIMs resulting from application of each methodology have different structural properties, i.e., differences in the number of: (1) binary variables, (2) allowable 0-1 combinations, and (3) constraints that reflect the different, though equivalent, angles from which the problem of transforming a logical sentence into a MIM can be attacked. Nonetheless, all satisfy two obvious fundamental properties, namely, that at any time (1) a single conclusion is true and (2) all logical relationships in the arguments of the corresponding sentence are valid (true).

We characterize one methodology as "backward model building" (BMB), because the properties of a MIM are established by following a backward procedure, i.e., from the truth or falsehood of the conclusions to that of the corresponding premises; the other methodology is characterized as "forward model building" (FMB), i.e, by establishing premises that lead to a corresponding conclusion.

3. The FMB and BMB Methodological Procedures

Each of the methodologies can be formalized as a step-by-step list of guidelines (meta-algorithm) as follows.

Step 1. *Introduce elementary sentences.* Introduce as many elementary logical sentences as necessary for the notation of the ultimate constituents in the CPF. The ultimate constituents are either a relationship ($\leq$, $=$, or $\geq$) between all (decision) variables (or between decision variables and constants) or sentences.

Step 2. *Compose the literal analogue of the CPF.* Based on the elementary sentences, compose a logical sentence in disjunctive form—the literal analogue of the CPF—in which each disjunct represents an argument, and the disjunction of all arguments is a tautology. In other words, the arguments in this sentence correspond to all possible combinations that can be simultaneously valid between its elementary sentences (i.e., the premises and the conclusions), so that their disjunction gives a universally true formula, or a tautology.

Step 3. *Determine the group of truth-value combinations and a sufficient set of 0-1 combinations.* Consider each argument of the sentence and determine the group of truth value combinations between the premises and the corresponding conclusion (after possible simplifications) that renders true (establishes) their corresponding logical relationship. Introduce as many binary variables as needed, so that they are in 1-to-1 correspondence with either the conclusions (in the case of BMB) or the premises (in the case of FMB) of the tautology under consideration. Determine the set of the allowable 0-1 combinations. In the case of BMB, if the cardinality of the set of allowable 0-1 combinations is at least equal to the maximum of the cardinality of the truth value combinations corresponding to each argument, then the allowable 0-1 set is sufficient. Otherwise, the 0-1 combinations must be enhanced by introducing additional binary variables corresponding to appropriate premise(s). In the case of FMB, from the definitional property of the tautology, the set of allowable 0-1 combinations of the binary variables that are introduced is always sufficient.

Step 4. *Modify the constraints of each argument.* Transform the constraints that correspond to the elementary sentences in an argument of the logical sentence so that their right-hand side is equal to 0. When a constraint is activated, the corresponding elementary sentence is true, and when relaxed, it is either true or false. We assume, however, that when a constraint is relaxed, then the corresponding elementary sentence is false, and this truth-"falsehood" is subsequently reflected in the values of a binary variable.[1] The constraints of an argument must be activated-relaxed according to the sufficient 0-1 combinations in order to—when considered together with the values of the binary variables corresponding to literals—establish such a group of truth value combinations that ensures the validity of both the conclusion and the logical relationship of the specific argument. The activation-relaxation is accomplished by a trial-and-error procedure that adds to the right-hand side of the constraints term(s) expressed as (1) the product of a linear combination of the binary variable(s) introduced in Step 3 to an M^+ parameter (large positive real number) if the constraint is of the form $\leq$ (at least one M^+ must exist), or to an M^- parameter (large negative real number) if the constraint is of the form $\geq$ (at least one M^- must exist), and/or (2) considers further modified constraints to the existing set. Repeat the process for all arguments.

Step 5. *Verify the representability of the MIM.* That the MIM built in Step 4 is an equivalent representation of the CPF is verified as follows. First, check that the solutions that make false the logical relationship between the premises and the conclusion in all arguments of the logical sentence are infeasible (here, ignore any irregularities in which values of decision variables make two or more conclusions coincide; these values are considered in the second phase of Step 5). For this, show that truth value combinations for which the premises are true and the conclusion is false (and vice versa when the premises and the conclusion are related by an equivalence relationship) could not occur (the truth of the elementary sentences corresponding to constraints is expressed by the satisfiability of the constraints per se, i.e., not by the values of the corresponding binary variable). Second, consider any irregularities in which values of decision variables make two or more conclusions coincide. Set the decision variable(s) equal to the value(s) that makes two or more conclusions have the same content. Then, check that the MIM is solvable with respect to the remaining decision variables and that the feasible solutions reflect all of the corresponding truth value combinations of the premises that make the conclusions coincide. If the verification[2] fails go to Step 2 and search for errors in the form of the logical sentence and/or go to Step 4 and search for errors in the activation-relaxation pattern of an argument and make appropriate corrections. Otherwise, continue with Step 6.

[1] The reason for adopting this practice is that when establishing the truth of a logical relationship in a specific argument, it is necessary to impose an appropriate group of truth value combinations between the premises and the corresponding conclusion; it seems particularly convenient to impose such a group by considering that the activation-relaxation of the constraint(s) of a specific argument resembles the truth-falsehood of the corresponding premises and conclusions, rather than considering the activated constraint(s) only that usually belong to many different arguments.

[2] An alternative way for verifying that a MIM built is an equivalent representation of a corresponding CPF is to examine whether the truth value combinations that are implied by the activated constraints and the binary variables corresponding to literals for each argument satisfy the two fundamental properties of a MIM. However, empirical evidence shows that the two-phase process described in Step 5 is particularly convenient, because it directly leads to the erroneous constraint(s), as well as to the appropriate term(s) that must be added to its right-hand side.

Step 6. *Eliminate possible redundancies.* The 0-1 combinations developed in Step 3 are sufficient in the sense that they are greater than or equal to the number necessary for establishing the truth value combinations for each argument, e.g., the same truth value combination could be established by (correspond to) more than one 0-1 combinations. Moreover, it is possible that either the same number of 0-1 combinations could be obtained by fewer binary variables or that some constraints are redundant, e.g., when the relaxation of a specific constraint is the weakest among all other constraint relaxations for all sufficient 0-1 combinations. Thus, redundancies in the MIM could appear in three ways: (1) in the number of sufficient 0-1 combinations and/or (2) in the number of binary variables (introduced for developing a sufficient number of 0-1 combinations), and/or (3) in the number of modified constraints. Eliminate possible redundancies by (1) introducing further constraints among the existing binary variables that restrict the sufficient 0-1 combinations to a necessary number of combinations and/or (2) establishing the same number of necessary 0-1 combinations (note that these are not the same combinations) by fewer binary variables (and an appropriate relationship among them) and/or (3) removing redundant modified constraints, respectively. In cases (1) and (3), continue with Step 7, while in case (2) go to Step 4 (because new 0-1 combinations are created).

Step 7. *Determine appropriate values for the big-M parameters.* The MIMs built after Step 6 are valid for any sufficiently large value of the big-M parameters, i.e., they represent a family of models. Consider the constraints of a MIM in its final form. Substitute M^+ and M^- in every constraint by their supremum and infimum, respectively, so that in their new form the activation-relaxation pattern for the allowable 0-1 combinations of the binary variables is preserved.

A schematic representation of the procedure is shown in Figure 1. Based on our experience, the most challenging task is the trial-and-error procedure of Step 4; for this, some empirical rules of thumb that might prove useful follow. Consider the constraints corresponding to the conclusions (BMB) or premises (FMB). In the case of BMB, at any time the constraints corresponding to a single conclusion must be activated and the remaining relaxed, and this truth-"falsehood" is reflected in the values of the corresponding binary variables. In the case of FMB, the activation-relaxation of the constraints of each premise indicates the truth-"falsehood" of the premise, which is reflected in the values of the corresponding binary variable. In either case, add to their right-hand side a single term that is the product of the corresponding binary variable (or its difference from one) and a large positive number (big-M) if the constraint is of the form ≤, and negative otherwise.

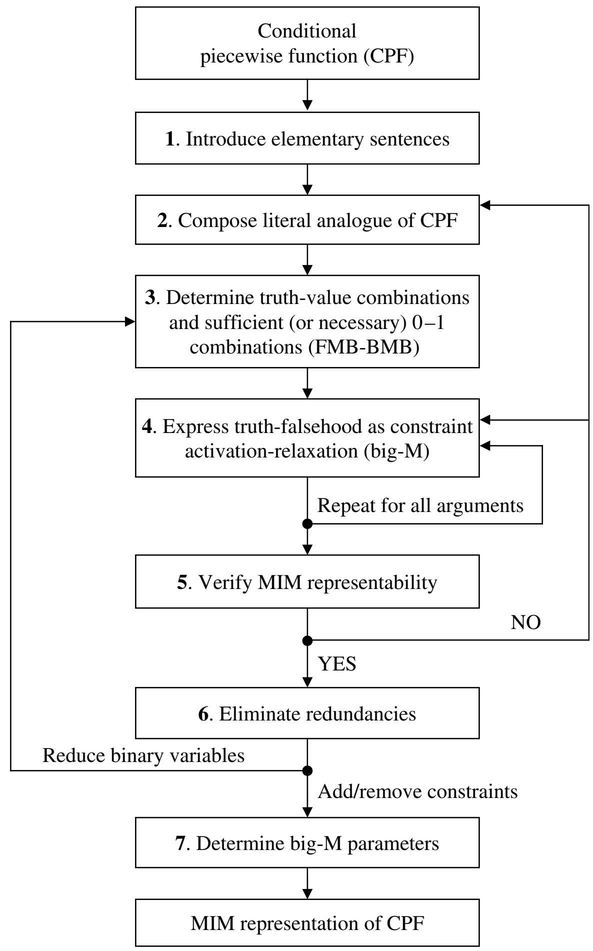

Figure 1 **A Schematic Representation of the Methodological Procedures**

The truth-falsehood of those conclusions (BMB) or premises (FMB) that do not correspond to a constraint is directly reflected in the values of the corresponding binary variable. Consider the set of the allowable 0-1 combinations and continue as described below with the activation-relaxation of the constraint(s) corresponding to the premises (BMB) or to the conclusion (FMB) of the argument according to the corresponding logical relationship.

In the BMB case, usually, the constraints for any premise will be activated (or relaxed) for more than one 0-1 combination. For their activation-relaxation pattern add to their right-hand side a term that is the product of the corresponding binary variable (or its difference from one) and a large positive number (big-M) if the constraint is of the form ≤, and negative otherwise. In the FMB case, setting up the activation relaxation pattern for the constraints corresponding to conclusions involves adding to their right-hand side (at most) as many terms as the number of premises in the corresponding argument. These terms are the multiple of the binary variables for the premises in this argument (or their difference from 1), and a number (positive if the inequality is of the form $\leq$, and negative otherwise). In either case, if the constraint is of the form $\leq$ there must be at least one term with a large positive number (big-M), while in case of a constraint of the form $\geq$ there must be at least one term with a large negative number. This is repeated for all arguments. Finally, for augmented 0-1 combinations in the BMB case, the truth of a conclusion may be indicated by more than one 0-1 combination, which further means that the corresponding constraints must be activated and relaxed accordingly. In this case, add to appropriate constraints (corresponding to a premise or to a conclusion) an additional term that is the product of a linear combination of any of the binary variable(s) introduced (i.e., corresponding either to a conclusion or a premise) and a number. This number is not necessarily large, and it can be either positive or negative, regardless of the form of the constraint (note that constraints of the form $\leq$ already have a large positive number, and constraints of the form $\geq$ have a large negative number in their right-hand sides). In either case, the activation-relaxation pattern of the constraints of an argument must be in compliance with that of the constraints of the remaining arguments.

4. MIM Representations for the Dispersion-and-Store Model

4.1. Expressing the Signal-Controlled Outflow Discharge Function as a Literal

We now demonstrate the applicability of the BMB and FMB methodologies by transforming into MIM representations the CPFs based on the dispersion-and-store model that describe the outflow discharge at the approach of an intersection controlled by a 3-band (green-yellow-red) signal, as well as for the case of the de facto red. For the sake of simplicity we consider the traffic dynamics described by the dispersion-and-store model on a unidirectional stretch of surface street that accommodates only through movements (see Figure 2); here the incoming flow is dispersed, and the dispersed platoon is stored in a storage segment if the signal at the end approach of the surface street is red. Both the dispersion and the conservation of vehicles are described by linear constraints, see, e.g., Cremer and Schoof (1989). However, a CPF describes the corresponding outflow discharge that it is equal to: (1) zero if the signal is red, (2) the minimum between the vehicles in the storage segment (appropriately expressed in flow rate units) and the saturation flow rate if green, and (3) a fraction of the saturation flow rate if yellow (that reflects the tail end of the phase lost time). We note that the same assumptions hold for the similar store-and-forward model, with the exception that the link inflow travels

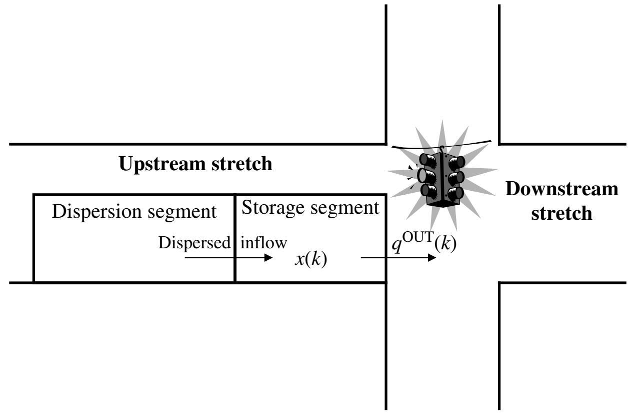

Figure 2 Schematic Representation of the Dispersion-and-Store Model

as a packet with a constant travel time toward the storage segment (i.e., a linear constraint).

We adopt a discrete time approach with the sample time interval denoted as T and the discrete time index as k. Consider a link as described above. Let C denote the capacity of the storage segment (expressed in number of vehicles) and $x(k)T$ represent the number of vehicles in the storage segment at time instant k, so that $x(k)$ is expressed in vph.

Let $q^{\text{OUT}}(k)$ be the traffic volume exiting the link (in vph), i.e., the number of exiting vehicles during the kth time interval $kT \leq t \leq (k+1)T$ divided by T, whereas $q^{\max}$ denotes the saturation flow rate of that link (in vph). By definition, $0 \leq x(k)T \leq C$, and $0 \leq q^{\text{OUT}}(k) \leq q^{\max}$ for the continuous variables $x(k)$ and $q^{\text{OUT}}(k)$, respectively. The signal indications during the kth time interval are modeled by a pair of binary variables $\tilde{g}(k)$ and $\tilde{y}(k)$, with the resulting four 0-1 combinations translated as follows: The combination $(\tilde{g}(k), \tilde{y}(k)) = (1, 0)$ indicates that "The signal is green," and the combination $(\tilde{g}(k), \tilde{y}(k)) = (1, 1)$ indicates that "The signal is yellow" (or "The yellow is on"); the combination $(\tilde{g}(k), \tilde{y}(k)) = (0, 0)$ or $(\tilde{g}(k), \tilde{y}(k)) = (0, 1)$ indicates that "The signal is red."

The CPF that describes the storage segment outflow discharge is constructed as:

$$q^{\text{OUT}}(k) = \begin{cases} 0 & \text{when } x(k) \in [0, C/T] \text{ and} \\ & (\tilde{g}(k), \tilde{y}(k)) = (0,0) \text{ or } (\tilde{g}(k), \tilde{y}(k)) = (0,1) \\ x(k) & \text{when } x(k) \in [0, q^{\max}) \text{ and} \\ & (\tilde{g}(k), \tilde{y}(k)) = (1,0) \text{ or} \\ & (\tilde{g}(k), \tilde{y}(k)) = (1,1) \\ \pi q^{\max} & \text{when } x(k) \in [q^{\max}, C/T] \text{ and} \\ & (\tilde{g}(k), \tilde{y}(k)) = (1,1) \\ q^{\max} & \text{when } x(k) \in [q^{\max}, C/T] \text{ and} \\ & (\tilde{g}(k), \tilde{y}(k)) = (1,0), \end{cases} \tag{2}$$

where the real parameter π satisfies $0 < \pi < 1$ and accounts for the possibility that only a fraction of the yellow change interval may be effectively used.

Transforming (2) into an MIM by either the BMB or the FMB methodology involves its translation into a literal expression by identifying the elementary sentences that correspond to its literal analogues and constraints. Because an elementary sentence takes only two values, two elementary sentences are necessary to describe the signal indications. Specifically, let the elementary sentence P_1 be "The signal is red" so that $P_1 = \mathrm{T}$ is expressed as $\tilde{g}(k) = 0$. In addition, we introduce the atomic proposition P_6, so that $P_6 = \mathrm{T}$ reads "The signal is yellow," and it is quantified by the variable $\tilde{y}(k)$ so that $\tilde{y}(k) = 1$ indicates that $P_6 = \mathrm{T}$. Understandably, the truth-falsehood of the combined sentences P_1 and P_6 must be considered in the earlier defined context of signal indications, i.e., that (1) $P_1 = \mathrm{F}$ and $P_6 = \mathrm{F}$ means that "The signal is green"; (2) $P_1 = \mathrm{F}$ and $P_6 = \mathrm{T}$ means that "The signal is yellow"; (3) $P_1 = \mathrm{T}$ and $P_6 = \mathrm{T}$ means that "The signal is red"; and (4) $P_1 = \mathrm{T}$ and $P_6 = \mathrm{F}$ means that "The signal is red." From these combinations, it is easy to see that a red indication is expressed by a single sentential valuation, i.e., $P_1 = \mathrm{T}$.

In a similar manner, we define that P_2 is the elementary sentence $x(k) < q^{\max}$. We further define that the elementary sentence P_3 corresponds to $q^{\mathrm{OUT}}(k) = 0$; the elementary sentence P_4 corresponds to $q^{\mathrm{OUT}}(k) = x(k)$; and the elementary sentence P_5 expresses that $q^{\mathrm{OUT}}(k) = q^{\max}$. Moreover, the elementary sentence $P_7 = \mathrm{T}$ is the literal analogue of the equation $q^{\mathrm{OUT}}(k) = \pi q^{\max}$, with $0 < \pi < 1$. In other words, the elementary sentence P_7 expresses that the outflow $q^{\mathrm{OUT}}(k)$ is equal to a fraction of the saturation flow rate, with the fraction being represented by the parameter $0 < \pi < 1$.

The negation of the previous elementary sentences has an obvious meaning, e.g., $\bar{P}_1$ expresses that "the signal is green" $\bar{P}_2$ corresponds to $x(k) \geq q^{\max}$, etc. Finally, it is noted that it is not necessary to introduce any elementary sentence for the constraints $0 \leq x(k) \leq C/T$ and $0 \leq q^{\mathrm{OUT}}(k) \leq q^{\max}$ that define the feasible range of the decision variables $x(k)$ and $q^{\mathrm{OUT}}(k)$ respectively, in the context of a mathematical programming problem.[3]

We now compose the elementary sentences into the literal analogue of the CPF (2). The sentence must be in disjunctive form, in which each disjunct (or argument) corresponds to each of the branches of (2). Expressing (2) as a compound sentence:

$$(P_1 \mapsto P_3) \vee ((\bar{P}_1 \wedge P_2) \Leftrightarrow P_4) \vee ((\bar{P}_1 \wedge \bar{P}_2 \wedge \bar{P}_6) \Leftrightarrow P_5) \vee ((\bar{P}_1 \wedge \bar{P}_2 \wedge P_6) \Leftrightarrow P_7). \tag{3}$$

(It is noted that by omitting the sentences P_6 and P_7 and the fourth argument, we obtain the literal expression for the outflow discharge in the case of a 2-band signal.)

4.2. Identifying Irregularities and Transforming the Literal into a MIM

Proposition (3) is a tautology that expresses all possible combinations that could occur regarding vehicle discharge at the approach of a signalized intersection at each discrete time instant, depending on (1) the signal indication, (2) the system state (i.e., the number of vehicles in the storage segment), (3) the saturation flow rate, and (4) the permitted outflow. Thus, the four-argument sentence (3) is read as "either a zero outflow is implied by a red signal (first disjunct), or the outflow is equal to the demand if and only if the signal is green and the demand at the storage segments is less than the saturation flow rate (second disjunct), or the outflow is equal to the saturation flow rate if and only if the signal is green and the demand is greater than or equal to the saturation flow rate (third disjunct), or the outflow is equal to some specified fraction of the saturation flow rate if and only if the signal is yellow and the demand is greater than or equal to the saturation flow rate." From the perspective of mathematical logic, (3) states that at each discrete time instant, any of the four conclusions is true and the others false, and that their truth or falsehood is determined according to their logical relationship (implication or equivalence) with the corresponding premise(s). Among the elementary sentences in proposition (3), P_1, P_2, and P_6 (or their negation) correspond to its premises, while P_3, P_4, P_5, and P_7 correspond to its conclusions. The definition (establishment) of a logical relationship between the premise(s) and the conclusion is determined by the set of possible events between them or, alternatively, by the set of truth valuations between the premise(s) and the conclusion (truth value combinations). Thus, for the first disjunct an implication relationship holds, i.e., "a red signal implies zero outflow" because "a green signal implies that the outflow can be either zero or greater than zero."

Note that in (3) the conclusions P_3 and P_4 can take on the value true simultaneously, i.e., their content coincides. This irregularity arises when $q^{\mathrm{OUT}}(k) = x(k) = 0$, i.e., it indicates the case in which a phase gaps out (the signal is green and the outflow is zero). Similarly, the conclusions P_4 and P_7 can be simultaneously true. Moreover, it is noted that the equivalence

[3] Throughout the paper the feasibility constraints are not incorporated as part of the corresponding MIM, because they are considered explicitly in defining an admissible region for the mathematical programming problem.

relationship between the conjunction of premises $\bar{P}_1 \wedge \bar{P}_2 \wedge P_6$ and the conclusion P_7 in the fourth argument of (3) holds only when $x(k) \neq \pi q^{\max}$. This is because in the fourth argument of (3) an irregularity arises when $x(k) = \pi q^{\max}$ and the conjunction of premises is false for the truth value combinations $(\bar{P}_1, \bar{P}_2, P_6) = (\mathrm{T,F,F})$ and $(\bar{P}_1, \bar{P}_2, P_6) = (\mathrm{T,F,T})$. With the conclusion $P_7 = \mathrm{T}$, the corresponding couple of truth value combinations (between the premises and the conclusion) is not allowed by the equivalence relationship of the fourth argument. In other words, a possible event occurs when: (1) "The signal is green or yellow," (2) the desired demand is bounded as $x(k) < q^{\max}$, and (3) the desired demand is given by $x(k) = \pi q^{\max}$, which results in $q^{\mathrm{OUT}}(k) = x(k)$. However, because $x(k) = \pi q^{\max}$, this further implies that $q^{\mathrm{OUT}}(k) = \pi q^{\max}$, which subsequently means that both conclusions $P_4 = \mathrm{T}$ and $P_7 = \mathrm{T}$ hold. Therefore, we impose the appropriate truth value combinations on the premises and the conclusion of the fourth argument in the case $x(k) \neq \pi q^{\max}$, knowing that in the case $x(k) = \pi q^{\max}$ (i.e., when $x(k) < q^{\max}$ or $\bar{P}_2 = \mathrm{F}$), the conclusion $P_7 = \mathrm{T}$ if $P_4 = \mathrm{T}$. As a final comment, note that P_4 and P_7 are both true (i.e., $q^{\mathrm{OUT}}(k) = x(k) = \pi q^{\max}$) if and only if $x(k) < q^{\max}$, because $\pi q^{\max} < q^{\max}$. This observation indicates that when developing a MIM for the logical sentence (3), we must develop a model by considering either $x(k) < q^{\max}$ (i.e., a strict inequality) or $x(k) \geq q^{\max}$ (note that this has been taken into account in (2); also, this is not necessary in the case of a 2-band signal).

Proposition (3) can be transformed into two different (though equivalent) MIM representations by following either the BMB or the FMB methodology. According to the BMB, it is necessary to introduce five binary variables (out of which two correspond to the signal indication) that allow for eight sufficient 0-1 combinations (five are necessary), and the representation consists of one equation and ten inequality constraints (in addition to the four feasibly constraints for the corresponding decision variables). Alternatively, according to the FMB methodology, three binary variables are necessary (out of which two correspond to the signal indication) that give rise to eight sufficient 0-1 combinations (six are necessary), and the corresponding MIM consists of eight inequality constraints (in addition to the four feasibility constraints for the corresponding decision variables).

A third option, which is particularly convenient because of space limitations, for demonstrating how to apply the methodologies on transforming (2) into a MIM comes by combining the BMB and FMB. A close examination of both (2) and (3) reveals the following. The decision variable $q^{\mathrm{OUT}}(k)$ should take one of the following values: (1) $0 \leq q^{\mathrm{OUT}}(k) = x(k) < q^{\max}$, (2) $q^{\mathrm{OUT}}(k) = 0$, (3) $q^{\mathrm{OUT}}(k) = q^{\max}$, and (4) $q^{\mathrm{OUT}}(k) = \pi q^{\max}$. Note that the values of 0 and $q^{\max}$ are at the boundaries of the feasible range for the variable $q^{\mathrm{OUT}}(k)$, i.e., $0 \leq q^{\mathrm{OUT}}(k) \leq q^{\max}$, and only the value $\pi q^{\max}$ lay in the interior of this interval. The symmetry of the discrete values 0 and $q^{\max}$ regarding the interval $[0, q^{\max}]$ provides the foundation for BMB model building by introducing three binary variables with the following properties: (1) $\beta_k^1 = 1$ indicates that $P_3 = \mathrm{T}$ whose quantification corresponds to $q^{\mathrm{OUT}}(k) = 0$; (2) $\beta_k^2 = 1$ indicates that $P_4 = \mathrm{T}$, whose quantification corresponds to $0 \leq q^{\mathrm{OUT}}(k) = x(k) < q^{\max}$; and (3) $\beta_k^3 = 1$ indicates that $P_5 = \mathrm{T}$ whose quantification corresponds to $q^{\mathrm{OUT}}(k) = q^{\max}$.

Alternatively, for the modeling of the truth or falsehood of the conclusion P_7 we can follow FMB methodological procedure. In other words, we can express the truth or falsehood of the conclusion P_7 according to its logical relationship with the corresponding conjunction of premises. The conclusion P_7 is the literal analogue of the equation $q^{\mathrm{OUT}}(k) = \pi q^{\max}$ with $\pi q^{\max} < q^{\max}$. Because the value $\pi q^{\max}$ lay in the interior of interval $[0, q^{\max}]$, our approach is to restrict the bounds of the variable $q^{\mathrm{OUT}}(k)$ as $\pi q^{\max} \leq q^{\mathrm{OUT}}(k) \leq \pi q^{\max}$, so as to impose that $q^{\mathrm{OUT}}(k) = \pi q^{\max}$ at any time instant the corresponding conjunction of the premises is true, and to relax the bounds of these constraints otherwise. For this, we use the binary variables β_k^1, β_k^2, β_k^3 that were introduced earlier and, in addition, the binary variable $\tilde{y}(k)$.[4]

The reader can easily verify that the constraints set corresponding to relationships (4a)–(4h) offers an equivalent representation of the CPF (2), which we will refer to as MIM-STORAGE-3BAND.

MIM-STORAGE-3BAND: An Equivalent Representation of the CPF (2)

Constraint for determining the sufficient 0-1 combinations (when combined with $\tilde{y}(k)$):

$$\beta_k^1 + \beta_k^2 + \beta_k^3 = 1, \tag{4a}$$

Transformed constraints corresponding to elementary sentence:

$$P_3: \quad q^{\mathrm{OUT}}(k) \leq (1 - \beta_k^1)\mathrm{M}^+; \tag{4b}$$

$$P_2: \quad x(k) - q^{\max} \leq (1 - \beta_k^2)\mathrm{M}^+ - \varepsilon; \tag{4c}$$

$$P_4: \quad q^{\mathrm{OUT}}(k) - x(k) \leq (1 - \beta_k^2)\mathrm{M}^+; \tag{4d}$$

[4] For brevity, the binary variable corresponding to P_7 was directly symbolized as $\tilde{y}(k)$ because of the meaning of P_7; even if another variable was introduced, e.g., β_k^4, it is immediately obvious from the valuations of the corresponding constraints that coincides (i.e., it has the same meaning) with $\tilde{y}(k)$. Also, from the valuation of MIM-STORAGE-3BAND that follows it becomes evident that the signal indications are described by the 0-1 values of the pair of binary variables $(\beta_k^1, \tilde{y}(k))$, i.e., the variable β_k^1 coincides with $\tilde{g}(k)$ defined above, except that they have opposite valuations for the same signal indication.

Table 1 Valuating MIM-STORAGE-3BAND for the Sufficient 0-1 Combinations

0-1 combination	Constraints feasible region	
$(\beta_k^1, \beta_k^2, \beta_k^3, \tilde{y}(k)) = (1, 0, 0, 0)$	$0 \le x(k) \le C/T$	$q^{\text{OUT}}(k) = 0$
$(\beta_k^1, \beta_k^2, \beta_k^3, \tilde{y}(k)) = (1, 0, 0, 1)$	$0 \le x(k) \le C/T$	$q^{\text{OUT}}(k) = 0$
$(\beta_k^1, \beta_k^2, \beta_k^3, \tilde{y}(k)) = (0, 1, 0, 0)$	$0 \le x(k) < q^{\max}$	$q^{\text{OUT}}(k) = x(k)$
$(\beta_k^1, \beta_k^2, \beta_k^3, \tilde{y}(k)) = (0, 1, 0, 1)$	$0 \le x(k) < q^{\max}$	$q^{\text{OUT}}(k) = x(k)$
$(\beta_k^1, \beta_k^2, \beta_k^3, \tilde{y}(k)) = (0, 0, 1, 0)$	$q^{\max} \le x(k) \le C/T$	$q^{\text{OUT}}(k) = q^{\max}$
$(\beta_k^1, \beta_k^2, \beta_k^3, \tilde{y}(k)) = (0, 0, 1, 1)$	$q^{\max} \le x(k) \le C/T$	$q^{\text{OUT}}(k) = \pi q^{\max}$

$$P_4\colon\quad q^{\text{OUT}}(k) - x(k) \ge (1-\beta_k^2)\text{M}^-; \tag{4e}$$

$$\bar{P}_2\colon\quad x(k) - q^{\max} \ge (1-\beta_k^3)\text{M}^-; \tag{4f}$$

$$P_5\colon\quad q^{\text{OUT}}(k) - q^{\max} \ge (1-\beta_k^3)\text{M}^- + \tilde{y}(k)(1-\pi)(-q^{\max}); \tag{4g}$$

$$P_7\colon\quad q^{\text{OUT}}(k) - q^{\max} \le (1-\beta_k^3)\text{M}^+ + \tilde{y}(k)(1-\pi)(-q^{\max}). \tag{4h}$$

In Table 1 below, MIM-STORAGE-3BAND is valuated for all six sufficient 0-1 combinations of the binary variables β_k^1, β_k^2, β_k^3, and $\tilde{y}(k)$ for determining the feasible region of the constraints (4a)–(4h). In Table 2 we provide the group of inferred truth value combinations between the premises and the conclusions in each argument of (3) corresponding to each 0-1 combination by considering the pair of binary variables β_k^1 and $\tilde{y}(k)$ (that correspond to signal indications), as well as the activated constraints of MIM-STORAGE-3BAND. The totality of these combinations (keeping in mind the irregularities in which the content of two conclusions coincides and become simultaneously true) makes evident that MIM-STORAGE-3BAND is an equivalent representation of sentence (3), i.e., at each time instant it ensures the validity of a single conclusion and of all logical relationships in its arguments. Regarding MIM-STORAGE-3BAND, it is noted that ε denotes an infinitesimal number so that (4c) expresses a strict inequality, whereas M^+ and M^- correspond to a large positive and negative value, respectively.

Table 2 Verifying the Representability of MIM-STORAGE-3BAND

	Truth values combination			
0-1 combination	(P_1, P_3)	$(\bar{P}_1, P_2, P_4)$	$(\bar{P}_1, \bar{P}_2, \bar{P}_6, P_5)$	$(\bar{P}_1, \bar{P}_2, P_6, P_7)$
(1, 0, 0, 0)	(T, T)	(F, T or F, F)	(F, T or F, T, F)	(F, T or F, F, F)
(1, 0, 0, 1)	(T, T)	(F, T or F, F)	(F, T or F, F, F)	(F, T or F, T, F)
(0, 1, 0, 0)	(F, T or F)	(T, T, T)	(T, F, T, F)	(T, F, F, T or F)
(0, 1, 0, 1)	(F, T or F)	(T, T, T)	(T, F, F, F)	(T, F, T, T or F)
(0, 0, 1, 0)	(F, F)	(T, F, F)	(T, T, T, T)	(T, T, F, F)
(0, 0, 1, 1)	(F, F)	(T, F, F)	(T, T, F, F)	(T, T, T, T)
Implied logical relationship	$P_1 \mapsto P_3$	$(\bar{P}_1 \wedge P_2) \Leftrightarrow P_4$	$(\bar{P}_1 \wedge \bar{P}_2 \wedge \bar{P}_6) \Leftrightarrow P_5$	$(\bar{P}_1 \wedge \bar{P}_2 \wedge P_6) \Leftrightarrow P_7$

Three brief comments on possible redundancies in MIM-STORAGE-3BAND follow. First, the three 0-1 combinations generated by the constraint $\beta_k^1 + \beta_k^2 + \beta_k^3 = 1$ could also be generated by two binary variables, say $\hat{\beta}_k$ and $\tilde{\beta}_k$, and the constraint $\hat{\beta}_k + \tilde{\beta}_k \le 1$ (and as before, combined with 0-1 values of $\tilde{y}(k)$). Thus, the corresponding new MIM has one fewer binary variable than does MIM-STORAGE-3BAND; however, this is counterbalanced by the new signal indication corresponding to the 0-1 combinations of three binary variables, which subsequently increases both the number of constraints and the complexity of the control strategy model. Second, the combination $(\beta_k^1, \beta_k^2, \beta_k^3, \tilde{y}(k)) = (1, 0, 0, 1)$ is redundant (i.e., five 0-1 combinations are necessary) and can be eliminated either by considering the constraint $\beta_k^1 + \tilde{y}(k) \le 1$, or, by simply developing a control strategy model that prohibits the occurrence of this combination. Third, another constraint corresponding to P_7 is needed to restrict the bounds of the variable $q^{\text{OUT}}(k)$ as $\pi q^{\max} \le q^{\text{OUT}}(k) \le \pi q^{\max}$, which amounts to

$$q^{\text{OUT}}(k) \ge (1-\beta_k^3)\text{M}^- + \tilde{y}(k)(\pi q^{\max}). \tag{4i}$$

However, (4i) has the weakest relaxation for all 0-1 combinations, and it is thus redundant.

4.3. Considering the Case of a De Facto Red

Consider next the case in which the storage capacity of the downstream stretch cannot accommodate the outflow discharge of the upstream stretch; i.e., the de facto red phenomenon. The same traffic flow parameters and variables described above are used in this case, with "DS" used as subscript to indicate the downstream stretch (see Figure 2). To describe this case, it is necessary to add the following premise in the first argument of logical sentence (3):

$$\text{IF } x(k)_{\text{DS}} > \frac{C_{\text{DS}}}{T} - q^{\max} \text{ (regardless the signal indication), THEN } q^{\text{OUT}}(k) = 0. \tag{5}$$

The premise $x(k)_{\text{DS}} > C_{\text{DS}}/T - q^{\max}$ ensures that in transition from a de facto red to green, the reserve capacity in the downstream stretch can accommodate an outflow discharge equal to $q^{\max}$. Depending on the application, another possible assumption is to consider that $x(k)_{\text{DS}} > C_{\text{DS}}/T - \text{minimum}\,\{x(k), \pi q^{\max}, q^{\max}\}$, with $0 < \pi < 1$. However, the consistency of this approach outweighs that of the "storage" type of model (in any case, the representation of the min operator by constraints is provided in §5.1).

To represent the premise in (5) by constraints we introduce the binary variable $\tilde{s}_k$, so that $\tilde{s}_k = 1$ indicates that the condition $x(k)_{\text{DS}} \le C_{\text{DS}}/T - q^{\max}$ is valid;

we then combine the big-M and the ε-modeling methods to obtain the set of inequalities (6a) and (6b) shown below:

$$x(k)_{\mathrm{DS}} - \left(\frac{C_{\mathrm{DS}}}{T} - q^{\max}\right) \le (1 - \tilde{s}_k)\mathrm{M}^{+} \tag{6a}$$

$$x(k)_{\mathrm{DS}} - \left(\frac{C_{\mathrm{DS}}}{T} - q^{\max}\right) \ge \tilde{s}_k\mathrm{M}^{-} + \varepsilon. \tag{6b}$$

The MIM that describes the de facto red conclusion in (5) is developed on the existing MIM-STORAGE-3BAND as follows. First, it is necessary to consider the linear inequality (7) shown below:

$$q^{\mathrm{OUT}}(k) \le \tilde{s}_k\mathrm{M}^{+}. \tag{7}$$

Second, we need to appropriately modify specific constraints of MIM-STORAGE-3BAND. Consider the constraints with the variable $q^{\mathrm{OUT}}(k)$ on their left-hand side. For these, it is necessary to add to their right-hand side the term $(1-\tilde{s}_k)\mathrm{M}^{+}$ if the inequality is of the form $\le$, and the term $(1-\tilde{s}_k)\mathrm{M}^{-}$ if the inequality is of the form $\ge$. The (partially) new MIM-STORAGE-3-BAND plus constraints (6a), (6b), and (7) comprise what we refer to as MIM-STORAGE-3BAND-SPILLBACK. It is easy to verify that this MIM is an equivalent representation of the outflow discharge function that describes the de facto red phenomenon at the approach of a signalized intersection. The determination of appropriate values for the parameters M^{+} and M^{-} in this model is a trivial task.

5. MIM Representations for the Cell Transmission Model

5.1. MIM Representations for Signal-Controlled Link Traffic

In this section we present the equivalent representation by a MIM of the CPFs in the cell transmission model (CTM) of Daganzo that describe the traffic dynamics in a single stretch of roadway (1994, 1995b), and when traffic signal control is present. The CTM is an elegant finite approximation of the so-called LWR model proposed by Lighthill and Whitham (1955) and Richards (1956), based on a fundamental diagram (i.e., flow-density relationship) of trapezoidal form. The model has been extensively applied in traffic control schemes; see, e.g., Lo (2001), and in dynamic traffic assignment applications, see, e.g., Ziliaskopoulos (2000). In this approximation, the conservation principle of vehicles is already a linear equation. However, the relationship that determines the uninterrupted or signal-controlled traffic flow transferred at adjacent (i.e., ordinary), merging, and diverging cells is a CPF.

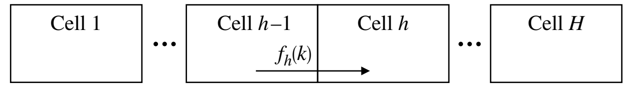

Figure 3 **A Roadway Stretch Subdivided into Cells (Segments)**

We start by considering a single-lane, unidirectional stretch of road, for example, part of a surface street or a freeway without entry or exit points. Assume further that this stretch is divided into homogeneous cells (segments), h, so that their length is equal to the distance traveled by the free-flowing traffic during the sample time interval T. The cells are numbered consecutively, starting with the upstream end of the road from $h = 1, \ldots, H$ in the direction of advancing traffic (see Figure 3), and are observed every T time units. In agreement with our previous notation, k corresponds to the discrete time index. The variable $n_h(k)$ describes the number of vehicles contained in cell h at time instant k, i.e., the cell occupancy. The following parameters are defined for each cell.

$N_h(k)$ denotes the storage capacity of cell h, i.e., the maximum number of vehicles that can be present in cell h at time instant k (equal to the product of the cell length and the jam density). Also, $q_h^{\max}(k)$ denotes the inflow capacity of cell h, i.e., the maximum number of vehicles that can flow from cell $h-1$ to cell h during the kth time interval $kT \le t \le (k+1)T$ (equal to the product of the sample time interval to the cell capacity). The variable $f_h(k)$ represents the inflow to cell h, i.e., the number of vehicles allowed to flow from cell $h-1$ to cell h during the kth time interval $kT \le t \le (k+1)T$. Finally, the parameters w and $u^{\max}$ represent the speed of the backward shock wave and the free flow speed, respectively.

Then, the CTM is described by the recursive pair of relationships shown below:

$$n_h(k+1) = n_h(k) + f_h(k) - f_{h+1}(k), \tag{8a}$$

$$f_h(k) = \min\left\{n_{h-1}(k), q_h^{\max}(k), \frac{w}{u^{\max}}(N_h(k) - n_h(k))\right\}. \tag{8b}$$

Clearly, in (8b) the difference $N_h(k) - n_h(k)$ represents the reserve (spare) capacity of cell h at time instant k. The parameters $q_h^{\max}(k)$ and $N_h(k)$ vary with time to allow the modeling of such transient traffic conditions as those resulting by application of traffic control measures, occurrence of incidents, etc. All quantities except w and $u^{\max}$ in (8b) are expressed in number of vehicles units. Also, for reasons described in Daganzo (1994), the accuracy of (8b) is enhanced when the ratio $w/u^{\max}$ holds if $n_{h-1}(k) > q_h^{\max}(k)$ and takes on a value of unity if $n_{h-1}(k) \le q_h^{\max}(k)$; this simple either-or condition has already been prescribed above as constraints (6a) and (6b).

We make the following notational simplifications: (1) $f_h(k) \stackrel{d}{\longrightarrow} q^{\text{OUT}}(k)$, (2) $n_{h-1}(k) \stackrel{d}{\longrightarrow} n(k)$, (3) $q_h^{\max}(k) \stackrel{d}{\longrightarrow} Q(k)$, and, (4) $(w/u^{\max})(N_h(k)-n_h(k)) \stackrel{d}{\longrightarrow} h(k)$, where the letter "d" over the arrow means "denoted as." According to this notation, the function (8b) is simplified into

$$q^{\text{OUT}}(k) = \min\{n(k), Q(k), h(k)\}. \tag{9}$$

The three-place minimum operator (9) above is expressed by the logical sentence (10) below (note that for brevity we have not introduced any notational symbolism for the corresponding elementary sentences):

$$\begin{aligned}&\{(h(k)\le Q(k))\wedge(h(k)\le n(k))\Leftrightarrow(q^{\text{OUT}}(k)=h(k))\}\\ &\vee\{(Q(k)\le n(k))\wedge(Q(k)\le h(k))\Leftrightarrow(q^{\text{OUT}}(k)=Q(k))\}\\ &\vee\{(n(k)\le Q(k))\wedge(n(k)\le h(k))\Leftrightarrow(q^{\text{OUT}}(k)=n(k))\}.\end{aligned} \tag{10}$$

The sentence (10) can be transformed into a MIM by either the BMB or the FMB procedure by introducing three binary variables corresponding to the three conclusions or the three premises, respectively. For either methodology, it is necessary to establish a group of three 0-1 combinations. For this, we introduce the binary variables[5] δ_1 and δ_2, and we further consider the inequality $\delta_1+\delta_2\le 1$, which gives rise to the following three combinations: (1) $(\delta_1,\delta_2)=(0,0)$; (2) $(\delta_1,\delta_2)=(1,0)$; and (3) $(\delta_1,\delta_2)=(0,1)$. Further application of the FMB methodology leads to the transformation of sentence (10) to MIM-FMB-CELL shown in (11a)–(11d) below.

MIM-FMB-CELL: An Equivalent Representation of the Minimum Operator (9)

$$\delta_1+\delta_2\le 1; \tag{11a}$$

$$(\delta_1+\delta_2)\mathrm{M}^- \le q^{\text{OUT}}(k)-n(k)\le 0; \tag{11b}$$

$$(1-\delta_1+\delta_2)\mathrm{M}^- \le q^{\text{OUT}}(k)-h(k)\le 0; \tag{11c}$$

$$(1+\delta_1-\delta_2)\mathrm{M}^- \le q^{\text{OUT}}(k)-Q(k)\le 0. \tag{11d}$$

Valuating MIM-FMB-CELL for each of the 0-1 combinations is shown in Table 3. From Table 3, by examining the constraints for each 0-1 combination we can infer a group of truth value combinations that establishes the validity of: (1) a single conclusion and (2) all logical relationships in sentence (10).

We consider now the traffic flow bottleneck phenomenon, commonly observed because of lane drops or the lane closure, either from such unpredictable factors as incidents or from such scheduled events as road work. The result is a reduction of the capacity flow in the vicinity upstream of the lane closure. Suppose that the bottleneck occurs in cell h and that the flow capacities of cells $h-1$ and h are denoted as $Q_{h-1}(k)$ and $Q_h(k)$, respectively, corresponding to the maximum possible flow through cells $h-1$ and h as, e.g., an indicator of the cell widths.[6] Then, the bottleneck phenomenon is modeled by explicitly expressing the parameter $q_h^{\max}(k)$ as

$$q_h^{\max}(k) = \min\{Q_{h-1}(k), Q_h(k)\}. \tag{12}$$

For (12) we assume that, for some time instances, $Q_h(k) < Q_{h-1}(k)$. By application of either the BMB or the FMB methodology, the function (12) is transformed into the MIM shown below:

$$\delta_3\mathrm{M}^- \le q_h^{\max}(k)-Q_{h-1}(k)\le 0; \tag{13a}$$

$$(1-\delta_3)\mathrm{M}^- \le q_h^{\max}(k)-Q_h(k)\le 0. \tag{13b}$$

In (13a) and (13b), δ_3 denotes a binary variable so that $\delta_3=1$ indicates that $q_h^{\max}(k)=Q_h(k)$ and $q_h^{\max}(k)\le Q_{h-1}(k)$ (i.e., $q_h^{\max}(k)=Q_h(k)\le Q_{h-1}(k)$), whereas $\delta_3=0$ indicates that $q_h^{\max}(k)=Q_{h-1}(k)$ and $q_h^{\max}(k)\le Q_h(k)$ (i.e., $q_h^{\max}(k)=Q_{h-1}(k)\le Q_h(k)$). Obviously, there is an equivalence relationship between the premise and the corresponding conclusion.

In transforming (8b) and (12) into a MIM for the description of the traffic flow bottleneck, we introduce three binary variables, namely, δ_1, δ_2, and δ_3, and the variable $q_h^{\max}(k)$; the model consists of a set of nine inequality constraints, namely, (11a)–(11d), (13a), and (13b) (in addition to the feasibility constraints). Note, however, that a better MIM representation with only two binary variables can be obtained if (12) is substituted in (8b) to yield the equivalent expression that contains a four-place minimum operator:

$$f_h(k) = \min\left\{n_{h-1}(k), Q_{h-1}(k), Q_h(k), \frac{w}{u^{\max}}(N_h(k)-n_h(k))\right\}. \tag{14}$$

This is shown in the next section during the treatment of the flow transferred between ordinary cells in the case of network representation.

Table 3 Valuating MIM-FMB-CELL for the 0-1 Combinations

0-1 combination	Constraints feasible region		
$(\delta_1,\delta_2)=(0,0)$	$q^{\text{OUT}}(k)=n(k)$	$q^{\text{OUT}}(k)\le h(k)$	$q^{\text{OUT}}(k)\le Q(k)$
$(\delta_1,\delta_2)=(1,0)$	$q^{\text{OUT}}(k)\le n(k)$	$q^{\text{OUT}}(k)=h(k)$	$q^{\text{OUT}}(k)\le Q(k)$
$(\delta_1,\delta_2)=(0,1)$	$q^{\text{OUT}}(k)\le n(k)$	$q^{\text{OUT}}(k)\le h(k)$	$q^{\text{OUT}}(k)=Q(k)$

[5] For notational simplicity, in the transformation examples that follow the discrete time index k is omitted from the binary variables.

[6] This revised definition of the maximum possible flow through cell h is also more convenient for network modeling that is described in the following.

5.2. MIM Representations for Signal-Controlled Network Traffic

We proceed with the description of the CTM for network traffic (Daganzo 1995a). Consider a general transportation network described by a directed graph of nodes and arcs and suppose that each arc is subdivided into cells as described before. Clearly, for general graphs it is not possible to number the cells consecutively and specify that vehicles should always proceed to the next numbered cell. Instead, we will describe the system of cells as the nodes of a more detailed graph and the possible vehicle transfers by a set of links. Cells will be denoted by capital characters and links by lowercase characters. The topology of the detailed network is defined by specifying for each link l a "beginning" cell (denoted as Bl) and an "ending" cell (denoted as El).

According to the previous (single roadway) definition, the parameter $q_h^{\max}(k)$ expresses the maximum number of vehicles that could enter cell h per unit time, assuming that there is room to store them; i.e., it represents the maximum possible flow on the link from cell $h-1$ to h. Alternatively, $q_h^{\max}(k)$ could have been defined to be the maximum possible flow through cell h, and then the minimum of $q_{h-1}^{\max}(k)$ and $q_h^{\max}(k)$ could have been defined to be the maximum flow from cell $h-1$ to h (i.e., in a way similar to the traffic bottleneck phenomenon). Links play a minor role in the revised cell formulation; they simply define the connection of the cells, ensuring that vehicles are transferred among appropriate nodes. The permissible network topologies are those in which the maximum number of links entering and/or leaving a cell is three. Thus, cells can be classified into "ordinary" if one link enters the cell and one leaves it, "merge" if two links enter the cell and one leaves it, and "diverge" if only one link enters the cell and two leave it.

Origins and destinations are modeled as ordinary cells (Daganzo 1995a). In the following case we develop MIMs for the CPFs that describe the link flow(s) in the cases of (a) ordinary cells, (b) ordinary cells controlled by traffic signals, (c) merge cells, and (d) diverge cells, as shown in Figure 4 (the letter C or c stands for "complementary"). We start by considering the case of ordinary links (see Figure 4(a)). By denoting $f_l(k)$ as the flow on link l from time instant k to time instant $k+1$ (kth time interval), then the equivalent of (8b) for an ordinary link is:

$$f_l(k) = \min\left\{n_{Bl}(k), Q_{Bl}(k), Q_{El}(k), \frac{w_{El}}{u_{El}^{\max}}(N_{El}(k) - n_{El}(k))\right\}. \tag{15}$$

(Note the similarity between (14) and (15).) For the sake of notational simplicity, let: (1) $f_l(k) \xrightarrow{d} q^{\text{OUT}}(k)$, (2) $n_{Bl}(k) \xrightarrow{d} n(k)$, (3) $Q_{Bl}(k) \xrightarrow{d} Q_1(k)$, (4) $Q_{El}(k) \xrightarrow{d} Q_2(k)$, and (5) $(w_{El}/u_{El}^{\max})(N_{El}(k) - n_{El}(k)) \xrightarrow{d} h(k)$. According to this notation, the function (15) is simplified into:

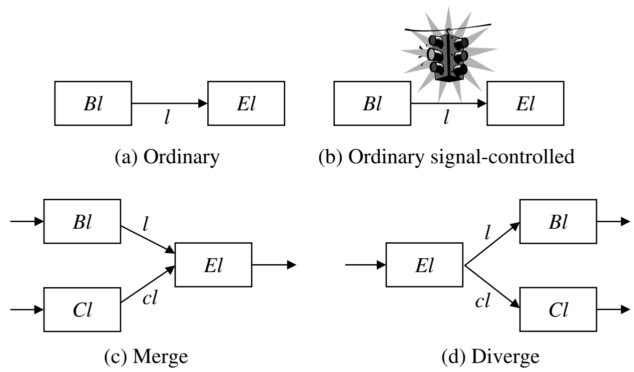

Figure 4 Schematic Representations of Considered Cell Connections for Network Traffic

$$q^{\text{OUT}}(k) = \min\{n(k), Q_1(k), Q_2(k), h(k)\}. \tag{16}$$

The four-place minimum operator in (16) is trivially expressed by a logical sentence that is an extension of (10) and is represented by the equivalent constraint set shown in (17a)–(17d) (γ_1 and γ_2 are binary variables). We refer to the representation in (17a)–(17d) as MIM-CELL-NET-ORDINARY. In Table 4, MIM-CELL-NET-ORDINARY is valuated for each 0-1 combination. It is obvious from this table, by examining the constraints for each 0-1 combination, that MIM-CELL-NET-ORDINARY is an equivalent representation of (16).

MIM-CELL-NET-ORDINARY: An Equivalent Representation of the CPF (16)

$$(\gamma_1 + \gamma_2)\text{M}^- \le q^{\text{OUT}}(k) - n(k) \le 0; \tag{17a}$$

$$(1 + \gamma_1 - \gamma_2)\text{M}^- \le q^{\text{OUT}}(k) - Q_1(k) \le 0; \tag{17b}$$

$$(1 - \gamma_1 + \gamma_2)\text{M}^- \le q^{\text{OUT}}(k) - Q_2(k) \le 0; \tag{17c}$$

$$(2 - \gamma_1 - \gamma_2)\text{M}^- \le q^{\text{OUT}}(k) - h(k) \le 0. \tag{17d}$$

We proceed with the development of a MIM representation for the case of ordinary links controlled

Table 4 Valuating MIM-CELL-NET-ORDINARY for the 0-1 Combinations

0-1 combination (γ_1, γ_2)	Constraints feasible region			
(0, 0)	$q^{\text{OUT}}(k) = n(k)$	$q^{\text{OUT}}(k) \le Q_1(k)$	$q^{\text{OUT}}(k) \le Q_2(k)$	$q^{\text{OUT}}(k) \le h(k)$
(0, 1)	$q^{\text{OUT}}(k) \le n(k)$	$q^{\text{OUT}}(k) = Q_1(k)$	$q^{\text{OUT}}(k) \le Q_2(k)$	$q^{\text{OUT}}(k) \le h(k)$
(1, 0)	$q^{\text{OUT}}(k) \le n(k)$	$q^{\text{OUT}}(k) \le Q_1(k)$	$q^{\text{OUT}}(k) = Q_2(k)$	$q^{\text{OUT}}(k) \le h(k)$
(1, 1)	$q^{\text{OUT}}(k) \le n(k)$	$q^{\text{OUT}}(k) \le Q_1(k)$	$q^{\text{OUT}}(k) \le Q_2(k)$	$q^{\text{OUT}}(k) = h(k)$

by a 2-band and a 3-band signal (see Figure 4(b)). Consider the case of a 2-band signal for formulating, e.g., a ramp metering strategy. Suppose that the signal indication is given as before by the binary variable $\tilde{g}(k)$, so that $\tilde{g}(k)=1$ indicates that the signal is green during the kth time interval $kT \le t \le (k+1)T$. Then, the throughput capacity $Q_{Bl}(k)$ of cell Bl is described by the function (as before $q^{\max}$ denotes the saturation flow rate):

$$Q_{Bl}(k) = \begin{cases} q^{\max} & \text{when } \tilde{g}(k)=1 \\ 0 & \text{when } \tilde{g}(k)=0. \end{cases} \tag{18}$$

Transforming (18) into a set of constraints trivially amounts to

$$Q_{Bl}(k) = \tilde{g}(k)q^{\max}. \tag{19}$$

Consider next the case of a 3-band signal for formulating, e.g., a surface street signal strategy. As before, the pair of binary variables $\tilde{g}(k)$ and $\tilde{y}(k)$ is defined so that the 0-1 the combination $(\tilde{g}(k), \tilde{y}(k)) = (1,0)$ indicates a green, whereas the combination $(\tilde{g}(k), \tilde{y}(k)) = (1,1)$ indicates a yellow. Both combinations $(\tilde{g}(k), \tilde{y}(k)) = (0,0)$ and $(\tilde{g}(k), \tilde{y}(k)) = (0,1)$ indicate a red, with all signal indications corresponding to the kth time interval $kT \le t \le (k+1)T$. Then, $Q_{Bl}(k)$ is given by the following function (the parameters π and $q^{\max}$ follow the earlier definitions):

$$Q_{Bl}(k) = \begin{cases} 0 & \text{when } n_{Bl}(k) \in [0, N_{Bl}(k)] \text{ and} \\ & (\tilde{g}(k),\tilde{y}(k))=(0,0) \text{ or } (\tilde{g}(k),\tilde{y}(k))=(0,1) \\ \pi q^{\max} & \text{when } n_{Bl}(k) \in [q^{\max}, N_{Bl}(k)] \text{ and} \\ & (\tilde{g}(k),\tilde{y}(k))=(1,1) \\ q^{\max} & \text{when } n_{Bl}(k) \in [0, q^{\max}) \text{ and} \\ & (\tilde{g}(k),\tilde{y}(k))=(1,0) \text{ or } (\tilde{g}(k),\tilde{y}(k))=(1,1) \\ & \text{or} \\ & \text{when } n_{Bl}(k) \in [q^{\max}, N_{Bl}(k)] \text{ and} \\ & (\tilde{g}(k),\tilde{y}(k))=(1,0). \end{cases} \tag{20}$$

Because of space limitations, (20) is transformed into a MIM according only to the FMB. For this, we first introduce the binary variable $\tilde{\delta}_k$, so that $\tilde{\delta}_k=1$ indicates that $n_{Bl}(k) < q^{\max}$, whereas $\tilde{\delta}_k=0$ indicates that $n_{Bl}(k) \ge q^{\max}$. Again, the constraints that define the physical restrictions of the variables $n_{Bl}(k)$ and $Q_{Bl}(k)$, i.e., $0 \le n_{Bl}(k) \le N_{Bl}(k)$ and $0 \le Q_{Bl}(k) \le q^{\max}$, must be considered. Then, application of the FMB methodology leads to MIM-FMB-CELL-3BAND shown in (21a)–(21f).

Table 5 **Valuating MIM-FMB-CELL-3BAND for the 0-1 Combinations**

0-1 combination $(\tilde{g}(k), \tilde{y}(k), \tilde{\delta}_k)$	Constraints feasible region	
(0, 0, 0)	$n_{Bl}(k) \ge q^{\max}$	$Q_{Bl}(k)=0$
(0, 1, 0)	$n_{Bl}(k) \ge q^{\max}$	$Q_{Bl}(k)=0$
(0, 0, 1)	$n_{Bl}(k) < q^{\max}$	$Q_{Bl}(k)=0$
(0, 1, 1)	$n_{Bl}(k) < q^{\max}$	$Q_{Bl}(k)=0$
(1, 1, 0)	$n_{Bl}(k) \ge q^{\max}$	$Q_{Bl}(k)=\pi q^{\max}$
(1, 0, 0)	$n_{Bl}(k) \ge q^{\max}$	$Q_{Bl}(k)=q^{\max}$
(1, 1, 1)	$n_{Bl}(k) < q^{\max}$	$Q_{Bl}(k)=q^{\max}$
(1, 0, 1)	$n_{Bl}(k) < q^{\max}$	$Q_{Bl}(k)=q^{\max}$

MIM-FMB-CELL-3BAND: An Equivalent Representation for the CPF (20)

$$n_{Bl}(k) - q^{\max} \le (1-\tilde{\delta}_k)\mathrm{M}^+ - \varepsilon; \tag{21a}$$

$$n_{Bl}(k) - q^{\max} \ge \tilde{\delta}_k\mathrm{M}^-; \tag{21b}$$

$$Q_{Bl}(k) \le \tilde{g}(k)\mathrm{M}^+; \tag{21c}$$

$$Q_{Bl}(k) - \pi q^{\max} \ge (1-\tilde{g}(k)+\tilde{\delta}_k)\mathrm{M}^- + (2-\tilde{g}(k)-\tilde{y}(k))(1-\pi)q^{\max}; \tag{21d}$$

$$Q_{Bl}(k) - q^{\max} \ge (2-\tilde{g}(k)-\tilde{\delta}_k)\mathrm{M}^-; \tag{21e}$$

$$Q_{Bl}(k) - q^{\max} \le (1-\tilde{g}(k)+\tilde{\delta}_k)\mathrm{M}^+ + (1-\tilde{g}(k)-\tilde{y}(k))(1-\pi)q^{\max}. \tag{21f}$$

MIM-FMB-CELL-3BAND is valuated for the eight sufficient 0-1 combinations in Table 5 (the constraint $\tilde{g}(k) + \tilde{y}(k) \ge 1$ reduces the 0-1 combinations necessary to six), which makes evident that it is an equivalent representation of (20). (Note that in the corresponding BMB formulation, three binary variables are introduced, allowing for six sufficient 0-1 combinations—five are necessary, with the corresponding MIM consisting of seven constraints.)

Consider now the case of a merging cell configuration (Figure 4(c)). Cells Bl and Cl can send a maximum of $S_{Bl}(k)$ and $S_{Cl}(k)$, with $S_{Bl}(k) = \min\{Q_{Bl}, n_{Bl}\}$ and $S_{Cl}(k) = \min\{Q_{Cl}, n_{Cl}\}$, respectively, whereas cell El can receive a maximum flow of $R_{El}(k)$ that is defined as $R_{El}(k) = \min\{Q_{El}, (w_{El}/u_{El}^{\max})(N_{El}(k) - n_{El}(k))\}$.

The underlying idea in the merging configuration is that if there is sufficient spare capacity in cell El, i.e., $R_{El}(k) \ge S_{Bl}(k) + S_{Cl}(k)$, then the maximum traffic possible $S_{Bl}(k)$ and $S_{Cl}(k)$ advances into cell El; otherwise, the maximum number possible of vehicles $R_{El}(k)$ advances into cell El. As long as the supply of vehicles from both approaches $S_{Bl}(k)$ and $S_{Cl}(k)$ is not exhausted, it is assumed that a fraction ρ_l of the vehicles comes from Bl and the remaining ρ_{cl} from Cl, with $\rho_l + \rho_{cl} = 1$. (Here, the constants ρ_l and ρ_{cl} that capture priority attributes at the merging site are assumed to be known.) If the supply of vehicles on one of the approaches is satisfied before the end of the kth time interval, then the remaining vehicles to advance will come from the complementary (competing) approach. This is formalized as follows:

$$\text{if } R_{El} \geq S_{Bl} + S_{Cl}, \quad \text{then} \begin{cases} f_l(k) = S_{Bl} \\ \quad \text{and} \\ f_{cl}(k) = S_{Cl} \end{cases}$$
$$\text{if } R_{El} < S_{Bl} + S_{Cl}, \qquad (22)$$
$$\text{then} \begin{cases} f_l(k) = \text{mid}\{S_{Bl}, R_{El} - S_{Cl}, \rho_l R_{El}\} \\ \quad \text{and} \\ f_{cl}(k) = \text{mid}\{S_{Cl}, R_{El} - S_{Bl}, \rho_{cl} R_{El}\} \end{cases}$$
$$\text{with } \rho_l + \rho_{cl} = 1,$$

where mid$\{\cdot\}$ denotes the medium point operator. For transforming (22) into a MIM, we consider the link flow function $f_l(k)$ (the resulting MIM will similarly apply for the flow function $f_{cl}(k)$). We will first provide a MIM for the medium point operator that will be subsequently modified so that it incorporates the conditions regarding $R_{El}(k)$ in (22). We make the following notational simplifications (the letter "d" over the arrow means "denoted as"): (1) $f_l(k) \xrightarrow{\text{d}} f$, (2) $S_{Bl}(k) \xrightarrow{\text{d}} \kappa$, (3) $R_{El} - S_{Cl} \xrightarrow{\text{d}} \lambda$, and, (4) $\rho_l R_{El} \xrightarrow{\text{d}} \mu$. According to this notation, the medium point operator in (22) is simplified as:

$$f = \text{mid}\{\kappa, \lambda, \mu\}. \qquad (23)$$

Following either the FMB or BMB methodology, we introduce three binary variables, σ_1, σ_2, and σ_3 and the constraints $\sigma_1 + \sigma_2 + \sigma_3 \geq 1$ and $\sigma_1 + \sigma_2 + \sigma_3 \leq 2$, which gives rise to six 0-1 combinations. The resulting MIM, called MIM-FMB-MID, is shown in (24a)–(24l).

MIM-FMB-MID: An Equivalent Representation for the Medium Point Operator (23)

0-1 Combinations:

$$\sigma_1 + \sigma_2 + \sigma_3 \geq 1; \qquad (24a)$$
$$\sigma_1 + \sigma_2 + \sigma_3 \leq 2, \qquad (24b)$$

Premises:

$$\kappa - \mu \leq (1 - \sigma_1 + \sigma_3)\text{M}^+; \qquad (24c)$$
$$\mu - \kappa \leq (1 - \sigma_3 + \sigma_1)\text{M}^+; \qquad (24d)$$
$$\lambda - \kappa \leq (1 - \sigma_1 + \sigma_2)\text{M}^+; \qquad (24e)$$
$$\kappa - \lambda \leq (2\sigma_1 + \sigma_2 + \sigma_3 - 1)\text{M}^+; \qquad (24f)$$
$$\mu - \lambda \leq (3 - 2\sigma_1 - \sigma_2 - \sigma_3)\text{M}^+; \qquad (24g)$$
$$\lambda - \mu \leq (1 + \sigma_1 - \sigma_2)\text{M}^+, \qquad (24h)$$

Conclusions:

$$(\sigma_1 + 2\sigma_2 + \sigma_3 - 1)\text{M}^- \leq f - \kappa \leq (\sigma_1 + 2\sigma_2 + \sigma_3 - 1)\text{M}^+; \qquad (24i)$$

Table 6 Valuating MIM-FMB-MID for the 0-1 Combinations

0-1 combination $(\sigma_1, \sigma_2, \sigma_3)$	Constraints feasible region	
(1, 0, 0)	$\lambda \leq \kappa \leq \mu$	$f = \kappa$
(0, 0, 1)	$\mu \leq \kappa \leq \lambda$	$f = \kappa$
(0, 1, 0)	$\kappa \leq \lambda \leq \mu$	$f = \lambda$
(1, 0, 1)	$\mu \leq \lambda \leq \kappa$	$f = \lambda$
(1, 1, 0)	$\kappa \leq \mu \leq \lambda$	$f = \mu$
(0, 1, 1)	$\lambda \leq \mu \leq \kappa$	$f = \mu$

$$(1 + \sigma_1 - \sigma_2 + \sigma_3)\text{M}^- \leq f - \lambda \leq (1 + \sigma_1 - \sigma_2 + \sigma_3)\text{M}^+; \qquad (24j)$$
$$(2 - \sigma_1 + \sigma_2 - \sigma_3)\text{M}^- \leq f - \lambda \leq (2 - \sigma_1 + \sigma_2 - \sigma_3)\text{M}^+; \qquad (24k)$$
$$(3 - \sigma_1 - 2\sigma_2 - \sigma_3)\text{M}^- \leq f - \mu \leq (3 - \sigma_1 - 2\sigma_2 - \sigma_3)\text{M}^+. \qquad (24l)$$

MIM-FMB-MID is valuated for the six 0-1 combinations in Table 6, confirming that it is an equivalent representation of (23). To express function (22), MIM-FMB-MID is trivially modified. Specifically, we introduce the constraints shown below:

$$R_{El}(k) - S_{Bl}(k) - S_{Cl}(k) \leq (1 - \nu)\text{M}^+ - \varepsilon \quad \text{and} \qquad (25a)$$
$$R_{El}(k) - S_{Bl}(k) - S_{Cl}(k) \geq \nu\text{M}^-, \qquad (25b)$$

which express that $\nu = 1$ when $R_{El} < S_{Bl} + S_{Cl}$ and $\nu = 0$ when $R_{El} \geq S_{Bl} + S_{Cl}$. In addition, we consider the constraints

$$\nu\text{M}^- \leq f - \kappa \leq \nu\text{M}^+. \qquad (26)$$

Remaining are appropriate modifications to constraints (24i)–(24l) to incorporate (25a) and (25b). For this, it is necessary to add to their left-hand side the term $(1 - \nu)\text{M}^-$, and to their right-hand side the term $(1 - \nu)\text{M}^+$ (i.e., if the constraint is of the form $\geq$ and of the form $\leq$, respectively).

We refer to the constraint set comprising (24a)–(24h), the modified constraints (24i)–(24l), and the constraints (25a), (25b), and (26) as MIM-CELL-NET-MERGE. It is easily verified that it is an equivalent representation of the CPF (22) that corresponds to decision variable $f_l(k)$.

Consider, finally, a diverging network cell configuration, as represented in Figure 4(d). As before, cell Bl can send a maximum of $S_{Bl}(k)$ vehicles during the kth time interval $kT \leq t \leq (k+1)T$ with $S_{Bl}(k) = \min\{Q_{Bl}, n_{Bl}\}$, whereas cell El can receive a maximum of $R_{El}(k)$ vehicles defined as $R_{El}(k) = \min\{Q_{El}, (w_{El}/u_{El}^{\max})(N_{El}(k) - n_{El}(k))\}$ and, similarly, cell Cl can receive a maximum of $R_{Cl}(k)$ vehicles that is defined as $R_{Cl}(k) = \min\{Q_{Cl}, (w_{Cl}/u_{Cl}^{\max}) \cdot$

$(N_{Cl}(k)-n_{Cl}(k))\}$. It is also assumed that the diverging proportions of $\vartheta_{Bl}(k)$ are $\vartheta_{El}(k)$ and $\vartheta_{Cl}(k)$ (with $\vartheta_{El}(k)+\vartheta_{Cl}(k)=1$) and, as in the merging case, exogenously specified, as well as that traffic flows in these proportions continuously during the kth time interval. Then, the number of vehicles emitted by cell Bl, i.e., $f_{Bl}(k)$, determines the turning flow according to the following linear constraints:

$$f_l(k)=\vartheta_{El}(k)f_{Bl}(k), \tag{27a}$$

$$f_{Cl}(k)=\vartheta_{Cl}(k)f_{Bl}(k). \tag{27b}$$

In a similar fashion, the amount of traffic emitted by cell Bl should be as large as possible without exceeding the amount of traffic that can be received by any of the diverging branches. Thus, $f_{Bl}(k)$ is determined by

$$f_{Bl}(k)=\min\{S_{Bl}(k), R_{El}(k)/\vartheta_{El}(k), R_{Cl}(k)/\vartheta_{Cl}(k)\}. \tag{27c}$$

Equation (27c) is similar to (9) that is represented in constraint form by MIM-FMB-CELL.

For completeness it is noted that the aforementioned MIM that describe the flows at the cell connectors must be considered together with linear equations analogous to (8a) for updating the corresponding cell occupancies. Moreover, in the aforementioned cases it is assumed that the merging, ρ_l and ρ_{cl}, and diverging, $\vartheta_{El}(k)$ and $\vartheta_{Cl}(k)$, portions are known. In case these are unknown, the CTM for network traffic includes a three-point maximum operator in the case of diverging cells. However, this can be trivially transformed into an MIM by, e.g., considering the negative of a minimum operator containing the negatives of the corresponding values.

6. Analyzing Existing MIM Representations

In the previous sections, the applicability of the BMB and FMB methodologies was demonstrated. Virtually every possible CPF that can be found when one is developing a mathematical representation for surface street networks, based either on the dispersion-and-store or the cell transmission traffic flow models, can be transformed via these procedures. However, the methodologies can be used in a "reverse way": for analyzing the structural properties and identifying possible redundancies in existing MIM representations. Such an example is provided in the following case.

Lo (2001) provides an alternative MIM for the three-place minimum operator (9) that appears in the CTM. According to this approach, (9) is viewed as nested two-place minimum operators, i.e.,

$$q^{\mathrm{OUT}}(k)=\min\{\min\{n(k), Q(k)\}, h(k)\}. \tag{28}$$

Then, a continuous variable $\phi(k)$ is initially introduced so that

$$\phi(k)=\min\{n(k), Q(k)\}, \tag{29a}$$

$$q^{\mathrm{OUT}}(k)=\min\{\phi(k), h(k)\}. \tag{29b}$$

Subsequently, in a first phase (29a) is transformed into a MIM after introduction a binary variable y_1, and in a second phase (29b) it is transformed into a MIM after introduction of another binary variable y_2, as shown in (30a)–(30f).

Analyzing an Alternative MIM Representation for the Minimum Operator (9)

First transformation phase:

$$\mathrm{M}^-(1-y_1)\le n(k)-Q(k)\le \mathrm{M}^+y_1; \tag{30a}$$

$$0\le n(k)-\phi(k)\le \mathrm{M}^+y_1; \tag{30b}$$

$$0\le Q(k)-\phi(k)\le \mathrm{M}^+(1-y_1), \tag{30c}$$

Second transformation phase:

$$\mathrm{M}^-(1-y_2)\le \phi(k)-h(k)\le \mathrm{M}^+y_2; \tag{30d}$$

$$0\le \phi(k)-q^{\mathrm{OUT}}(k)\le \mathrm{M}^+y_2; \tag{30e}$$

$$0\le h(k)-q^{\mathrm{OUT}}(k)\le \mathrm{M}^+(1-y_2). \tag{30f}$$

To summarize: for transforming (9) into constraints, three variables are introduced, namely, the continuous $\phi(k)$ and the binary y_1 and y_2, which give raise to four 0-1 combinations, and the corresponding MIM consists of twelve linear inequalities (30a)–(30f). The valuation of the MIM consisting of (30a)–(30f) for each of the 0-1 combinations makes evident that Lo's MIM is the constraint representation of the logical sentence (31) shown below:

$$\begin{aligned}&\{(h(k)\le Q(k))\wedge(Q(k)\le n(k))\Leftrightarrow(q^{\mathrm{OUT}}(k)=h(k))\}\\ &\vee\{(Q(k)\le n(k))\wedge(Q(k)\le h(k))\Leftrightarrow(q^{\mathrm{OUT}}(k)=Q(k))\}\\ &\vee\{(h(k)\le n(k))\wedge(n(k)\le Q(k))\Leftrightarrow(q^{\mathrm{OUT}}(k)=h(k))\}\\ &\vee\{(n(k)\le Q(k))\wedge(n(k)\le h(k))\Leftrightarrow(q^{\mathrm{OUT}}(k)=n(k))\}.\end{aligned} \tag{31}$$

Apparently, the transformation approach described in Lo (2001) implicitly considers a redundant logical sentence (the redundancy is expressed by the first and third argument of (31), as compared to sentence (10)) that consequently resulted in a redundant MIM, as compared to MIM-FMB-CELL (where 2 binary variables are introduced, allowing for three necessary 0-1 combinations, and the resulting model consists of 7 inequality constraints).[7]

[7] To realize the magnitude of the redundancies, recall that each MIM is considered for every cell h and discrete time instant k.

7. Future Work

The challenges associated with making the transformation procedure of either BMB or FMB automatic include:

a. the modeling of strict inequality constraints;

b. the programming (automation) of the trial-and-error procedure that takes place for the appropriate modification of constraint sets (corresponding to arguments);

c. the presence of irregularities that could arise in the literal expression corresponding to a CPF (as shown, e.g., for the logical sentence (3)); and

d. the nonuniqueness of the MIM representation for a specific CPF; following either methodology could lead to a variety of MIMs, some of which could contain redundant binary variables and/or allowable 0-1 combinations, and/or modified constraints.

Further issues that should be addressed are whether any particular approach is more convenient in developing MIM representations than another, and, for a specific CPF, whether any model from the array of models that can be developed either by the BMB or the FMB methods is better than the others. Regarding the first issue, it is noted that neither approach is inherently more convenient in model development over the other. From our experience, the FMB methodology is conceptually closer to human reasoning, because it initially deals with premises, but the appropriate transformation of the original constraints is harder. Alternatively, the BMB methodology is conceptually harder, because it initially deals with the conclusions, but the constraint transformation is easier when more conclusions than premises are present in the tautology. It is easily shown that both approaches lead to MIMs that are structurally different but equivalent. For the second issue, the answer is that it depends on the specific problem but, because both approaches lead to equivalent but structurally different MIMs, it is a simple task to identify the formulation that satisfies the property of having the smallest size of the search tree, i.e., of the feasible solutions (Nemhauser and Wolsey 1998).

Also of particular interest is the investigation of whether relaxation schemes other than big-M relaxations are possible for MIM built in the methodological context of BMB and FMB (see, e.g., Jeroslow and Lowe 1984, Jeroslow 1987, Beaumont 1990, Hooker 2000), as well as their potential impact in the solution of the corresponding mixed-integer problem. However, this nontrivial task is beyond the scope of this paper, and its investigation is left for future endeavors.

A last note regarding the relevance of our modeling approach, as compared to constraint programming (CP) follows. In comparing the convenience between mixed integer linear programming (MILP) and CP for modeling formulations for combinatorial optimization problems, we find that the latter provides a rich declarative language for stating problems because the model is developed as a computer program; this enables an effortless representation of logical constraints, etc. Alternatively, the solution algorithms of CP solvers are generally characterized by certain shortcomings in searching for the optimal solution; for this, for combinatorial optimization problems—which are typically decomposed into a feasibility and an optimization problem—CP is often used when a quick (and good) first feasible solution to a problem is desired, rather than proving the optimality of an existing feasible solution; see Lustig and Puget (2001) for an overview and a comparison between MILP and CP modeling and solution algorithms. There is a growing body of evidence that combining the complementary strengths of MILP and CP techniques results in the development of mathematical models (and corresponding algorithms) for efficiently solving hard combinatorial optimization problems. The aim of these methods is to combine the strength of MILP for proving optimality by using the LP relaxations and the power of CP for finding feasible solutions by using specialized constraint propagation algorithms (Bockmayr and Kasper 1998). Ways in which this modeling integration can be achieved include, e.g., the creation of a "shadow" mixed-integer model corresponding to the constraint programming model and its separate solution in two synchronized search tress (Jain and Grossmann 2001) or decomposition of the problem in two distinct subparts—one handled by CP and the other by MILP solvers—in a single search tree (Rodosek, Wallace, and Hajian 1999), or incorporating one within the other, resulting in the so-called mixed logical linear programming (Hooker and Osorio 1999).

8. Conclusions

A mathematical model of a process is usually an imperfect representation of what happens in reality; however, modeling inaccuracy should be such that model predictions describe the real-life process behavior within some specified error bounds. In particular, common in the consistent modeling of such traffic control operations as, e.g., traffic signal timing settings, ramp metering, etc., are CPFs that pose a modeling challenge in their representation as constraints in a mathematical programming problem.

In this paper, we have presented two general methodologies for transforming CPFs into MIM representations: (1) a BMB methodology, in which the two properties of the corresponding intermediate transformation that is a logical sentence are established by following a backward procedure, i.e., from the truth or falsehood of the conclusions to that of the corresponding premises, and (2) an FMB, in which the properties of the corresponding intermediate logical sentence are established by following a forward procedure. Each generates an MIM with such distinct structural properties as the number of linear constraints in the final representation, the number of binary variables introduced, etc., which reflects the different, though equivalent, angles from which the problem of transforming a logical sentence into a MIM can be attacked.

The methodologies can be applied along two fronts. First, for transforming particularly complex CPFs into a MIM, we developed MIMs for virtually every possible CPF that is found when one is modeling traffic control systems based either on the dispersion-and-store or the cell transmission model. Second, for analyzing the structural properties of existing MIMs, we showed how to identify and eliminate redundancies in binary variables, 0-1 combinations, and constraints.

Although there is empirical evidence that CPFs (as defined above) whose literal expression is a tautology could be represented by an MIM in binary variables, there is no theoretical proof that such a transformation is always possible. The methodologies presented are a set of guidelines (meta-algorithm) that provide a practical tool for transforming CPFs into MIMs in binary variables, requiring only very fundamental knowledge of mathematical logic (i.e., how to express logical connectors by truth-values combinations), and some trial-and-error for appropriately modifying constraints. The transformation examples presented here show the applicability of the methodologies and identify several practical aspects that could arise during the transformation procedure.

Acknowledgments
This research was partially supported by an award from the University of California Transportation Center and a Ph.D. dissertation award by the Regents of the University of California. Their support is gratefully acknowledged.

References

Algadhi, S., H. Mahmassani. 1990. Modeling crowd behavior and movement: Application to Makkah pilgrimage. M. Koshi, ed. *Transportation and Traffic Theory, Proc. 11th Internat. Sympos. Transit and Traffic Theory*. Elsevier, New York, 59–78.

Beaumont, N. 1990. An algorithm for disjunctive programs. *Eur. J. Oper. Res.* **48** 362–371.

Bockmayr, A., T. Kasper. 1998. Branch-and-infer: A unifying framework for integer and finite domain constraint programming. *INFORMS J. Comput.* **10**(3) 287–300.

Chang, G-L., J. Wu, H. Lieu. 1994. Real-time incident-responsive corridor control: A successive linear programming approach. *Proc. 4th Annual Meeting of IVHS America*, Vol. 2. Atlanta, 907–918.

Chang, G-L., H. Mahmassani, R. Herman. 1985. A macroscopic traffic simulation model to investigate peak-period commuter decision dynamics. *Transportation Res. Record* **1005** 107–121.

Cremer, M., S. Schoof. 1989. On control strategies for urban traffic corridors. *Proc. XXX IFAC Conf. Control Comput. Comm. Transportation*, Paris, France. Pergamon Press, Oxford, NY, 213–219.

Daganzo, C. F. 1994. The cell-transmission model: A simple dynamic representation of freeway traffic. *Transportation Res.-B* **28**(4) 269–287.

Daganzo, C. F. 1995a. The cell-transmission model. Part II. Network traffic. *Transportation Res.-B* **29**(2) 79–93.

Daganzo, C. F. 1995b. A finite difference approximation of the kinematic wave model of traffic flow. *Transportation Res.-B* **29**(4) 261–276.

Daganzo, C. F. 1995c. Requiem for second-order fluid approximations of traffic flow. *Transportation Res.-B* **29B**(4) 277–286.

D'Ans, G. C., D. C. Gazis. 1976. Optimal control of oversaturated store-and forward transportation networks. *Transportation Sci.* **10** 1–19.

Dantzig, G. B. 1963. *Linear Programming and Extensions*. Princeton University Press, Princeton, NJ.

De Schutter, B., B. De Moor. 1998. Optimal traffic light control for a single intersection. *Eur. J. Control* **4**(3) 260–276.

Gomes, G., R. Horowitz. 2006. Optimal freeway ramp metering using the asymmetric cell transmission model. *Transportation Res. Part C* **14** 244–262.

Hooker, J. N. 2000. Logic-based methods for optimization: Combining optimization and constraint satisfaction. *Wiley-Interscience Series in Discrete Mathematics and Optimization*. Wiley-Interscience, New York.

Hooker, J. N., M. A. Osorio. 1999. Mixed logical/linear programming. *Discrete Appl. Math.* **96–97**(1–3) 395–442.

Hromkovic, J. 2004. *Algorithms for Hard Problems* (*Introduction to Combinatorial Optimization, Randomization, Approximation, and Heuristics*), 2nd ed. Springer-Verlag, Berlin.

Ibaraki, T. 1976. Integer programming formulation of combinatorial optimization problems. *Discrete Math.* **16** 39–52.

Jain, V., I. E. Grossmann. 2001. Algorithms for hybrid MILP/CP models for a class of optimization problems. *INFORMS J. Comput.* **13**(4) 258–276.

Jeroslow, R. G. 1987. Representability in mixed integer programming, I: Characterization results. *Discrete Appl. Math.* **17** 223–243.

Jeroslow, R. G., J. K. Lowe. 1984. Modeling with integer variables. *Math. Programming Stud.* **22** 167–184.

Lighthill, M. J., J. B. Whitham. 1955. On kinematic waves. I flow movement in long rivers. II A theory of traffic flow on long crowded roads. *Proc. Roy. Soc. A* **229** 317–345.

Lin, W.-H., C. Wang. 2004. An enhanced 0-1 mixed-integer LP formulation for traffic signal control. *IEEE Trans. Intelligent Transportation Systems* **5**(4) 238–245.

Lo, H. K. 2001. A cell-based traffic control formulation: Strategies and benefits of dynamic timing plans. *Transportation Sci.* **35**(2) 148–164.

Lustig, I., J.-F. Puget. 2001. Program does not equal program: Constraint programming and its relationship to mathematical programming. *INTERFACES* **31**(6) 29–53.

Mayer, H.-H. 1981. A theoretical and computational comparison of "equivalent" mixed integer formulations. *Naval Res. Logist. Quart.* **28** 115–131.

Nemhauser, G. L., L. A. Wolsey. 1998. Integer and combinatorial optimization. *Wiley-Interscience Series in Discrete Mathematics and Optimization*. Wiley-Interscience, New York.

Papageorgiou, M. 1995. An integrated control approach for traffic corridors. *Transportation Res.-C* **3**(1) 19–30.

Richards, P. I. 1956. Shocks on the highway. *Oper. Res.* **4** 42–51.

Rodosek, R., M. G. Wallace, M. T. Hajian. 1999. A new approach to integrate mixed integer programming with constraint logic programming. *Ann. Oper. Res.* **86** 63–87.

Sheffi, Y., H. Mahmassani, W. B. Powell. 1982. A transportation network evacuation model. *Transportation Res.* **16A** 209–218.

Shoenfield, J. R. 1967. *Mathematical Logic*. Addison-Wesley, Reading, MA.

Singh, M. G., H. Tamura. 1974. Modelling and hierarchical optimization for oversaturated urban road traffic networks. *Internat. J. Control* **20**(6) 913–934.

Stephanedes, Y. J., K.-K. Chang. 1993. Optimal control of freeway corridors. *J. Transportation Engrg.* **119**(4) 504–514.

Ukkusuri, S. V., S. T. Waller. 2007. Linear programming models for the user and system optimal dynamic network design problem: Formulations, comparisons and extensions. *Networks and Spatial Econom.* Published online, DOI 10.1007/S11067-007-9019-6.

Williams, H. P. 1988. *Model Building in Mathematical Programming*. Wiley, New York.

Williams, H. P. 1995. Logic applied to integer programming and integer programming applied to logic. *Eur. J. Oper. Res.* **81** 605–616.

Wu, J., G.-L. Chang. 1999. An integrated optimal control and algorithm for commuting corridors. *Internat. Trans. Oper. Res.* **6** 39–55.

Ziliaskopoulos, A. K. 2000. A linear programming model for the single destination system optimum dynamic traffic assignment problem. *Transportation Sci.* **34**(1) 37–49.