# Solving Optimal Location of Traffic Counting Points at Urban Intersections in CLP(FD)

Ana Paula Tomás

DCC-FC & LIACC, Faculdade de Ciências, Universidade do Porto,
R. do Campo Alegre, 823, 4150-180 Porto, Portugal
Email: apt@ncc.up.pt

**Abstract.** We present an application of Constraint Logic Programming (CLP) for finding the minimum number and location of count-posts at urban roundabouts so as to obtain origin-destination data at minimum cost. By finding nice mathematical properties, we were able to model this problem as a constraint satisfaction problem in finite domains, and use CLP(FD) systems to solve it, with almost no implementation effort and very quickly.

## 1 Motivation and Introduction

Traffic studies require origin-destination (OD) data whose acquisition is difficult and expensive. Important, but often insufficient data is collected by counting vehicles that pass at specific points of the traffic network, either manually or automatically. In the past two decades, there has been a considerable amount of research on the difficult problem of OD matrix estimation from (link) traffic counts. Two recent publications [3, 13] address the optimisation of the number and location of traffic count-posts for traffic networks. In [3], a model and heuristic based algorithms are given for solving the minimum-cost Sensor Location Problem (SLP). This problem is that of finding the minimum number and location of count-posts in order to obtain the complete set of traffic flows on a transport network at minimum cost. Traffic studies for urban intersections are far less complex but still of prime importance in traffic engineering. In order to obtain OD data, it is typically necessary to carry out OD surveys, such as by manually recording registration numbers or making roadside video surveys. In this work, we present results from our research on the problem of finding the minimal number of exits and entries where OD surveys must be made, when some user-defined subset of traffic counts can be obtained at a relatively negligible cost (e.g., by direct observation in site). This can be viewed as a variant of the minimum-cost SLP. The model we propose is that of a constraint satisfaction problem in finite domains (CSP). No prior knowledge of turning movement coefficients is assumed or used in our work, which makes the problem clearly distinct from that in [3], so that we needed another approach.

This paper appears as a complement to previous work [14, 15], which involved an exhaustive case analysis of all hypothetical roundabouts with $n$ legs (i.e., where $n$ roads/streets meet), for increasing values of $n$, in computer. In it all strings $R_1R_2\ldots R_n \in \{\mathrm{E},\mathrm{D},\mathrm{S}\}^\star$ were seen as possible roundabouts, with $R_i \in \{\mathrm{E},\mathrm{D},\mathrm{S}\}$ indicating whether road $i$ is just an entry (E), just an exit (S) or both an entry and exit (D) road. In this way, we did not care whether some of these strings would ever represent real-world roundabouts. Both E and S refer to one-way streets, whereas D means double-way. Computer programs were developed (in C) to enumerate minimal subsets of OD flows that, when counted and used together with total counts at entries, exits and cross-sections, allow to deduce the complete OD matrix $(q_{ij})$ for a given time period. The focus was on the search for patterns for optimal cost solutions, that could eventually be used as practical rules by traffic engineers. To the best of our knowledge, no such a systematic study had been reported before. A major conclusion of [14] is that if a cost $c(q_{ij})$ is assigned to measuring $q_{ij}$, for each $q_{ij}$, the overall cost is minimized if the OD flows that should not be measured are selected in non-decreasing order of cost to form an independent set. In fact, in this case the problem is an instance of that of computing a maximum weight independent subset in a linear matroid which may always be solved by a Greedy Algorithm (see e.g. [4]). Therefore, it is solvable by a polynomial algorithm. Actually, that independence is the linear independence of the columns of the constraints matrix that are associated to the selected OD flows. It can be checked in polynomial time, for instance, by Gaussian elimination. Another important contribution of [14] was the development of a rather simple method to test for linear independence, for this specific problem, which involves only exact operations. By further exploiting its mathematical properties, in [15] it is shown the existence of an exact characterization of the optimal cost for the SLP when the OD flows $q_{i\,i+1}$'s are the ones of reduced measuring cost.

**New Contributions.** The work we now present is motivated by the claim [1] that the cost criteria should be more flexible to encompass specific features of the particular roundabout in study. By contrast to [14], the idea is no longer that of looking for patterns of optimal solutions but rather to solve the minimum-cost SLP for any given real-world roundabout, the relevant data being input by the end-user of the application. The problem structure is further exploited, putting some effort on the formulation of a fairly clean mathematical model of the problem so that, in the end, CLP(FD) systems could be used to solve it. For space reasons, we shall not include experimental results, but just say that problems are solved in fractions of a second. The use of Constraint Programming (CP) not only has led to a drastic reduction in the implementation effort but allows to easily extend the model to cater for other constraints that may be useful in practice. Although this work seems at first sight too focused, its methodology is of wider interest. The proposed solution for encoding the non-trivial constraint (4) (see section 3) represents a compromise between generate-and-test and constrain-and-generate techniques. It is an example of the tradeoff between efficiency and declarativeness. Significant pruning is achieved by extracting global, though incomplete, information. Furthermore, our approach clearly shows the advantage of exploiting affinities to well-known problems when facing a new one.

In the following section, we give a formal description of the problem and some of its mathematical properties. Then, in section 3, the CSP model and aspects of its implementation in CLP(FD) systems are discussed.

## 2 The Problem and Some Background

If we number the $n$ meeting roads in the way traffic circulates, the roundabout is perfectly identified by a string $R_1R_2 \ldots R_n \in \{\mathrm{E}, \mathrm{D}, \mathrm{S}\}^\star$, as defined before. Let $\mathcal{O} = \{\imath_1, \ldots, \imath_e\}$ and $\mathcal{D} = \{\jmath_1, \ldots, \jmath_s\}$ be the ordered sets of origins and destinations, and $e = |\mathcal{O}|$ and $s = |\mathcal{D}|$ be their cardinalities. The traffic flow from the entry $i$ to the exit $j$ is denoted by $q_{ij}$, for $i \in \mathcal{O}$ and $j \in \mathcal{D}$. These flows are related to the total volumes at entries, exits and passing through the cross-sections of the circulatory roadway in frontal alignment with the meeting roads (respectively, $O_i$'s, $D_j$'s and $F_k$'s) by (1)–(3).

$$\sum_{j \in \mathcal{D}} q_{ij} = O_i, \text{ for } i \in \mathcal{O} \tag{1}$$

$$\sum_{i \in \mathcal{O}} q_{ij} = D_j, \text{ for } j \in \mathcal{D} \tag{2}$$

$$\sum_{i \in \mathcal{O} \setminus \{k\}} \quad \sum_{j \in \mathcal{D},\ k \prec j \preceq i} q_{ij} = F_k, \text{ for } 1 \leq k \leq n \tag{3}$$

In (3), $j \in \mathcal{D},\ k \prec j \preceq i$ stands for the exits between road $k$ and road $i$, with $k$ excluded. Vehicles are assumed to exit the roundabout and, in addition, loops are disallowed (i.e., vehicles do not overpass their destination). Other cross-sections could be considered, as those that stand strictly between two consecutive roads, but the indeterminacy of (1)–(3) would not be reduced. For instance, the traffic volume $I_k$ that passes between roads $k$ and $k+1$ equals $F_k + O_k$. We also assume that counting $O_i$'s, $D_j$'s, $F_k$'s and $I_k$'s is of negligible cost when compared to other measuring tasks. So, a maximal independent set of such counts should be used, but it does not matter which one. At most $e + s$ total counts are globally non-redundant [14]. In fact, the matrix of the system (1)–(3), in the variables $q_{ij}$'s, has rank $e + s$ if and only if none of equations in (3) is of the form $F_k = 0$. Otherwise, the rank is $e + s - 1$, and the roundabout is identifiable by a string $R_1R_2 \ldots R_n$ of the language described by the regular expression $\mathrm{S}^\star(\mathrm{D} + \mathrm{SE})\mathrm{E}^\star$. We consider that traffic counts are error-free, an usually accepted condition [3, 13]. Thus, if some flows in (1)–(3) are replaced by traffic counts, the resulting system is consistent. Then, by standard results of systems of linear equations (e.g. [11]), it is equivalent to any of its subsystems that consist of equations (1)–(2) and one of (3), say that of $F_k$, to which our analysis shall be confined.

**Notation.** Let $\mathbf{P}'$ be the matrix of such subsystem, and $\mathbf{p}'_{ij}$ be the column of $q_{ij}$. Let $\mathbf{P}$ be the matrix of (1)–(2) and $\mathbf{p}_{ij}$ be the column of $q_{ij}$. Clearly, $\mathbf{p}'_{ij}$ has just one more element than $\mathbf{p}_{ij}$, namely the coefficient, say $\sigma_{ij}$, of $q_{ij}$ in the equation that defines $F_k$. In the sequel, we always refer to the case $k = 1$.

Provided that $es - rank(\mathbf{P}')$ of the $q_{ij}$'s are known (i.e., counted), the subsystem has a unique solution for the remaining $rank(\mathbf{P}')$ variables if and only if the related columns of $\mathbf{P}'$ are linearly independent. Let us take for example the roundabout SSEDE, where $\mathcal{O} = \{3, 4, 5\}$, $\mathcal{D} = \{1, 2, 4\}$. In this case, (1)–(3) may be written in solved form, for instance, as follows.

$$\begin{cases} q_{31} + q_{32} + q_{34} = O_3 \\ q_{41} + q_{42} + q_{44} = O_4 \\ q_{51} + q_{52} + q_{54} = O_5 \\ q_{31} + q_{41} + q_{51} = D_1 \\ q_{32} + q_{42} + q_{52} = D_2 \\ q_{34} + q_{44} + q_{54} = D_4 \\ q_{32} + q_{42} + q_{44} + q_{52} + q_{54} = F_1 \end{cases} \Leftrightarrow \begin{cases} q_{31} = q_{42} + q_{52} + F_1 - O_4 + D_1 - D_2 \\ q_{32} = -q_{42} - q_{52} + D_2 \\ q_{34} = O_5 - F_1 + O_4 - D_1 + O_3 \\ q_{41} = -q_{42} - q_{51} - q_{52} + D_2 + O_5 - F_1 + O_4 \\ q_{44} = q_{51} + q_{52} + F_1 - D_2 - O_5 \\ q_{54} = -q_{51} - q_{52} + O_5 \end{cases}$$

If the values of $q_{42}, q_{52}$ and $q_{51}$ are known (as well as $O_3$, $O_4$, $O_5$, $D_1$, $D_2$ and $F_1$) then the system has a unique solution for the remaining $q_{ij}$'s. The set $\mathcal{B}' = \{\mathbf{p}'_{31}, \mathbf{p}'_{32}, \mathbf{p}'_{34}, \mathbf{p}'_{41}, \mathbf{p}'_{44}, \mathbf{p}'_{54}\}$ is a basis (i.e., maximal independent set) of the linear subspace spanned by the columns of $\mathbf{P}'$. It is interesting to note that $\mathbf{p}'_{42} = \mathbf{p}'_{41} - \mathbf{p}'_{31} + \mathbf{p}'_{32}$, $\mathbf{p}'_{51} = \mathbf{p}'_{54} - \mathbf{p}'_{44} + \mathbf{p}'_{51}$ and $\mathbf{p}'_{52} = \mathbf{p}'_{54} - \mathbf{p}'_{44} + \mathbf{p}'_{41} - \mathbf{p}'_{31} + \mathbf{p}'_{32}$.

**Our Problem.** To find a basis $\mathcal{B}'$ formed of $rank(\mathbf{P}')$ independent columns of $\mathbf{P}'$, such that to measure the OD flows $q_{ij}$'s for the $\mathbf{p}'_{ij}$'s not in $\mathcal{B}'$, surveys are undertaken at a minimum number of exits and entries. Without further assumptions, it is not difficult to show that the optimal cost would be, in general, $(|\mathcal{O}| - 1) + (|\mathcal{D}| - 1)$, which means that the registration should take place at all the entries but one and at all the exits but one. We want to study whether this cost may be reduced by having some of the $q_{ij}$'s counted by other means, which are supposed to be less expensive, whatever that may mean.

*Praça da República* at Porto is a roundabout of type SEESDSE, for which a realistic scenario is as shown below, on the left, as claimed in [1].

| $\mathcal{O}\backslash^{\mathcal{D}}$ | 1 | 4 | 5 | 6 |
|---|---|---|---|---|
| 2 | | $\star$ | $\star$ | |
| 3 | | $\star$ | $\star$ | |
| 5 | | | | |
| 7 | $\star$ | | | |

| | 1 | 4 | 5 | 6 |
|---|---|---|---|---|
| 2 | $\bullet$ | $\star$ | $\star$ | $\bullet$ |
| 3 | $\bullet$ | $\star$ | ? | |
| 5 | $\bullet$ | $\bullet$ | | |
| 7 | $\star$ | $\bullet$ | $\bullet$ | $\bullet$ |

The OD flows that are cheap to obtain are marked with $\star$'s in the table on the left. In this particular case, by cheap we mean that they can be fully obtained by direct observation in site. An observer, standing at entry 2, can count $q_{24}$ and $q_{25}$. The same applies to the pair $q_{34}$ and $q_{35}$, and to $q_{71}$. By contrast, for instance, the geometry of the roundabout makes it too difficult to obtain $q_{56}$ by direct observation, although roads 5 and 6 are consecutive. The solution on the right is one of the eight optimal solutions found by our prototype implementations of the following model in SICStus Prolog [12, 2] and ECLiPSe [5]. It minimizes the number of locations where recording is done: at two entries (3 and 5) and two exits (5 and 6). The flows $q_{ij}$'s that would not be counted are marked with $\bullet$'s, implying that these $\mathbf{p}'_{ij}$'s belong to $\mathcal{B}'$. The ones that should be obtained by direct observation are marked with $\star$'s. The symbol ? indicates that there are two alternatives to get $q_{35}$, either by recording or by direct observation.

## 3 Modelling the Minimum-Cost SLP as a CSP

Given the string $R_1R_2 \dots R_n$ that identifies the roundabout, let $\mathcal{O} = \{\imath_1, \dots, \imath_e\}$ and $\mathcal{D} = \{\jmath_1, \dots, \jmath_s\}$, as above. Suppose that $\Gamma$ is the set of OD flows that may be obtained by some means that the user, possibly an engineer, finds preferable to recording, say $\Gamma = \{(i,j) \mid q_{\imath_i \jmath_j} \text{ may be collected by direct observation}\}$. Let $\Theta = \{i \mid (i,j) \in \Gamma \text{ for some } j\}$ define the table rows that have $\star$'s. In the sequel, *sensor* refers to recording means (e.g., somebody that is registering number-plates or a video that is recording image).

**The Decision Variables:**

$E_i \in \{0,1\}$ is 1 iff a sensor is located at entry $\imath_i$, with $1 \le i \le e$.
$S_j \in \{0,1\}$ is 1 iff a sensor is located at exit $\jmath_j$, with $1 \le j \le s$.
$V_i \in \{0,1\}$ is 1 iff an observer stands at entry $\imath_i$, with $i \in \Theta$.
$P_{ij} \in \{0,1\}$ is 1 iff $\mathbf{p}'_{\imath_i \jmath_j}$ would be in $\mathcal{B}'$, with $1 \le i \le e$, $1 \le j \le s$.
$X_{ij} \in \{0,1,-1\}$ if $(i,j) \in \Gamma$. Otherwise, $X_{ij} \in \{0,1\}$, for all $i$, $j$.

$$X_{ij} = \begin{cases} 1 \text{ iff } q_{\imath_i \jmath_j} \text{ is collected by a sensor} \\ -1 \text{ iff } (i,j) \in \Gamma \text{ and } q_{\imath_i \jmath_j} \text{ is directly collected at entry } i \\ 0 \text{ iff } q_{\imath_i \jmath_j} \text{ is not collected} \end{cases}$$

The model we propose is biased to allow effective propagation of values and constraints, when applying consistency techniques. This is one of the reasons why we have chosen $\{0,1,-1\}$ as domain of $X_{ij}$ when $(i,j) \in \Gamma$.

**The objective:** To minimize $\sum_{i=1}^{e} e_i E_i + \sum_{j=1}^{s} s_j S_j + \sum_{i \in \Theta} o_i V_i$, where $e_i$, $s_j$, and $o_i$ denote the defined costs for sensors and observers, the latter being assumed relatively insignificant.

**The constraints:** For all $1 \le i \le e$ and $1 \le j \le s$,

- To record the flow $q_{\imath_i \jmath_j}$, sensors must be located at entry $\imath_i$ and exit $\jmath_j$, that is $E_i + S_j \ge 2X_{ij}$. Notice that, if $E_i + S_j \in \{0,1\}$ then $X_{ij} \in \{0,-1\}$.
- To obtain $q_{\imath_i \jmath_j}$ by direct observation, there must be an observer at entry $\imath_i$, that is $V_i \ge -X_{ij}$.
- If $q_{\imath_i \jmath_j}$ is not explicitly collected then $\mathbf{p}'_{\imath_i \jmath_j} \in \mathcal{B}'$. Thus, $X_{ij} = 0 \Leftrightarrow P_{ij} = 1$ or equivalently, $X_{ij} + P_{ij} = 1$, for all $(i,j) \notin \Gamma$ and $X_{ij} P_{ij} = 0 \land X_{ij} + P_{ij} \neq 0$ for all $(i,j) \in \Gamma$.
- The set $\mathcal{B}' = \{\mathbf{p}'_{\imath_i \jmath_j} \mid P_{ij} = 1\}$ must be a basis to $\mathbf{P}'$. Equivalently, $\mathcal{B}'$ has cardinality $rank(\mathbf{P}')$, so $\sum_{i=1}^{e} \sum_{j=1}^{s} P_{ij} = rank(\mathbf{P}')$, and $\mathcal{B}'$ is a free set (i.e.

its elements are linearly independent), which is translated by (4), where the variables $\beta_{ij} \in \mathbb{R}$.

$$\sum_{\mathbf{p}'_{\imath_i \jmath_j} \in \mathcal{B}'} \beta_{ij} \mathbf{p}'_{\imath_i \jmath_j} = \mathbf{0} \Rightarrow \beta_{ij} = 0, \text{ for all } \beta_{ij} \tag{4}$$

– Although redundant, these are useful constraints during the search. An observer is placed at entry $\imath_i$ only if it has to count some $q_{\imath_i \jmath_j}$, with $(i,j) \in \Gamma$.

$$V_i = 1 \Rightarrow \exists_{(i,j)\in\Gamma}\ X_{ij} = -1 \tag{5}$$

A sensor is located at entry $\imath_i$ only if some OD flow $q_{\imath_i \jmath_j}$, with $(i,j) \notin \Gamma$, has to be counted. The same holds for sensors at exits.

$$\sum_{j=1,(i,j)\notin\Gamma}^{s} X_{ij} \geq E_i \qquad \text{and} \qquad \sum_{i=1,(i,j)\notin\Gamma}^{e} X_{ij} \geq S_j$$

Anyone with basic background on CLP(FD) may see that, with the exception of (4) and (5), these constraints can be straightforwardly encoded in CLP(FD). But, (5) may be implemented by *cardinality* operators — in SICStus Prolog, as `count(-1,XiStr,#>=,Vi)`, where `XiStr` is the list of $X_{ij}$ with $(i,j) \in \Gamma$ — and, in ECLiPSe, as `#(Vi,CtrXi,NSti)`, where `CtrXi` is a list of constraints $X_{ij} = -1$ and `NSti` its length, for $(i,j) \in \Gamma$. Encoding (4) is certainly more tricky and challenging. Firstly, (4) renders the model of MIP (mixed integer programming) type rather than a CSP, so that we would have to combine two different constraint solvers. Secondly, $\mathcal{B}'$ is in fact a variable, which means that a constraint as the builtin `element/3` would be of use to link the variables. Furthermore, (4) is too sophisticated to be too often touched during optimization, unless we keep some track of no-goods. Because of that, in [14], we have not used CP. Rather we computed $\mathcal{B}'$ by a completion procedure that implemented depth-first search with chronological backtracking. Gaussian elimination was used to test for freeness in an incremental way, which also allowed to implement intelligent backtracking. Conflicts due to choices made near the root of the search tree were detected during search. When some $\mathbf{p}'_{ij}$ was found dependent on a subset of the columns already in $\mathcal{B}'$, such information was propagated upwards along the search tree. A careful analysis of the program output gave us a deeper insight on the problem structure. An important result of [14] is that in (4), $\beta_{ij} \in \mathbb{R}$ may be replaced by $\beta_{ij} \in \{0, 1, -1\}$, meaning that the problem is actually a CSP in finite domains. Since (1)–(2) are the constraints of a Linear Transportation Problem, $\mathbf{P}$ has well-known nice properties (e.g., [8, 9]). From them, it follows that if a given $\mathbf{p}'_{ij} \notin \mathcal{B}'$ is a combination of the elements in a free subset $\mathcal{B}'$ of $\mathbf{P}'$, the unique linear combination giving $\mathbf{p}'_{ij}$ may be written as (6). Note that $\mathbf{p}'_{ij} = \mathbf{e}_i + \mathbf{e}_{e+j} + \sigma_{ij}\mathbf{e}_{e+s+1}$, for the unit vectors $\mathbf{e}_1, \ldots, \mathbf{e}_{e+s+1}$ of $\mathbb{R}^{e+s+1}$.

$$\mathbf{p}'_{ij} = \mathbf{p}'_{ij_1} - \mathbf{p}'_{i_1 j_1} + \mathbf{p}'_{i_1 j_2} - \cdots - \mathbf{p}'_{i_k j_k} + \mathbf{p}'_{i_k j} \tag{6}$$

In fact, the combination must mimic the one that defines $\mathbf{p}_{ij}$ in terms of the corresponding columns of $\mathbf{P}$. In SSEDE, $\mathbf{p}'_{52} = \mathbf{p}'_{54} - \mathbf{p}'_{44} + \mathbf{p}'_{41} - \mathbf{p}'_{31} + \mathbf{p}'_{32}$, because $\mathbf{p}_{52} = \mathbf{p}_{54} - \mathbf{p}_{44} + \mathbf{p}_{41} - \mathbf{p}_{31} + \mathbf{p}_{32}$ and $\sigma_{52} = \sigma_{54} - \sigma_{44} + \sigma_{41} - \sigma_{31} + \sigma_{32}$.

Given $\mathcal{B}'$, let $\mathcal{B} = \{\mathbf{p}_{ij} \mid \mathbf{p}'_{ij} \in \mathcal{B}'\}$. If $\mathcal{B}$ is not free, then $\mathcal{B}'$ is free if and only if there exists $\mathbf{p}_{ij} \in \mathcal{B}$ such that $\mathcal{B} \setminus \{\mathbf{p}_{ij}\}$ is free and the equality that gives $\mathbf{p}_{ij}$ as a combination of the columns in $\mathcal{B} \setminus \{\mathbf{p}_{ij}\}$ does not hold for the corresponding ones in $\mathcal{B}'$, because $\sigma_{ij} \neq \sigma_{ij_1} - \sigma_{i_1 j_1} + \sigma_{i_1 j_2} - \cdots - \sigma_{i_k j_k} + \sigma_{i_k j}$, as shown in [14].

Now, (6) has an interesting graphic interpretation. If we represent the elements of $\mathbf{P}$ in a table and link the ones that occur consecutively in (6) by edges, a simple cycle is defined, as we exemplify below.

| | 1 | 2 | 4 |
|---|---|---|---|
| 3 | $\mathbf{p}_{31}$ | $\mathbf{p}_{32}$ | $\mathbf{p}_{34}$ |
| 4 | $\mathbf{p}_{41}$ | $\mathbf{p}_{42}$ | $\mathbf{p}_{44}$ |
| 5 | $\mathbf{p}_{51}$ | $\mathbf{p}_{52}$ | $\mathbf{p}_{54}$ |

| $\sigma_{ij}$ | 1 | 2 | 4 |
|---|---|---|---|
| 3 | **0** | **1** | 0 |
| 4 | **0** | | **1** |
| 5 | 0 | **1** | **1** |

| | 1 | 4 | 5 | 6 |
|---|---|---|---|---|
| 2 | $\mathbf{p}_{21}$ | $\mathbf{p}_{24}$ | $\mathbf{p}_{25}$ | $\mathbf{p}_{26}$ |
| 3 | $\mathbf{p}_{31}$ | $\mathbf{p}_{34}$ | $\mathbf{p}_{35}$ | $\mathbf{p}_{36}$ |
| 5 | $\mathbf{p}_{51}$ | $\mathbf{p}_{54}$ | $\mathbf{p}_{55}$ | $\mathbf{p}_{56}$ |
| 7 | $\mathbf{p}_{71}$ | $\mathbf{p}_{74}$ | $\mathbf{p}_{75}$ | $\mathbf{p}_{76}$ |

| $\sigma_{ij}$ | 1 | 4 | 5 | 6 |
|---|---|---|---|---|
| 2 | **0** | **0** | 0 | 0 |
| 3 | **0** | **0** | 0 | 0 |
| 5 | 0 | 1 | **1** | **0** |
| 7 | 0 | 1 | **1** | **1** |

These cycles show that $\mathbf{p}'_{52} = \mathbf{p}'_{54} - \mathbf{p}'_{44} + \mathbf{p}'_{41} - \mathbf{p}'_{31} + \mathbf{p}'_{32}$ in SSEDE, $\mathbf{p}'_{21} = \mathbf{p}'_{24} - \mathbf{p}'_{34} + \mathbf{p}'_{31}$ (a problematic square) and $\mathbf{p}'_{55} \neq \mathbf{p}'_{56} - \mathbf{p}'_{76} + \mathbf{p}'_{75}$ (an unproblematic square) in SEESDSE. By our hypothesis that no vehicle overpasses its destination and the definition of $F_1$, it follows that the coefficient $\sigma_{ij}$ (wrt $F_1$) is given by $\sigma_{ij} = 0$ if and only if $j = 1$ or $j > i$.

A subset $\mathcal{B}$ of the columns of $\mathbf{P}$ is free if and only if the graph $G_{\mathcal{B}}$ built in the following way is acyclic. Its vertices are the elements of $\mathcal{B}$, and each edge links a vertex in a given row (respectively, column) to the closest vertex in the same row (respectively, column), if there are at least two vertices in that line.

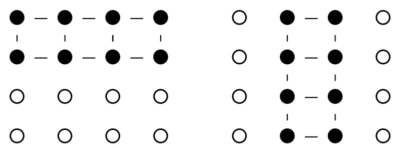
A subset $\mathcal{B}$ is a basis of the subspace spanned by the columns of $\mathbf{P}$ if and only if $G_{\mathcal{B}}$ is a tree and contains at least a vertex in each column and in each row of the table (e.g., [8, 9]). This result supports our definition of `basis/5`, below.

In order to profit from consistency-based approaches to solving CSPs [10], we would like to replace (4) by other constraints that, although being incomplete, may be used to prune the search space in a more active and effective way. Moreover, we did not really want to spend much time tuning the implementation, but rather take advantage of CP as "one of the closest approaches computer science has yet made to the Holy Grail of programming: the user states the problem, the computer solves it" [7]. Three necessary conditions on $\mathcal{B}'$ are deduced from the properties that $G_{\mathcal{B}}$ must satisfy: $\sum_{j=1}^{s} P_{ij} \geq 1$ and $\sum_{i=1}^{e} P_{ij} \geq 1$ and what we call *the Square-Free Property* (SQF, for short), which aims at getting rid of cycles that have a square shape. SQF states that,

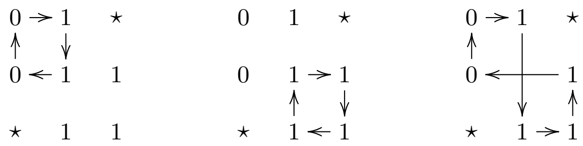
$$\text{if } \sigma_{\imath_i \jmath_j} - \sigma_{\imath_i \jmath_{j'}} + \sigma_{\imath_{i'} \jmath_{j'}} - \sigma_{\imath_{i'} \jmath_j} = 0 \text{ then } P_{ij} + P_{ij'} + P_{i'j'} + P_{i'j} < 4 \qquad (7)$$

and if in addition, none of these four positions (in a square) contains a $\star$ (i.e., if $\{(i,j),(i,j'),(i',j'),(i',j)\} \cap \Gamma = \emptyset$), then $E_i + E_{i'} \geq 1$ and $S_j + S_{j'} \geq 1$. Though the latter are binary constraints and arc-consistency techniques may be employed, we can achieve at least the same pruning while possibly defining fewer and more global constraints. Let us refer back to SEESDSE. Suppose there are no $\star$'s and that the entries with $\bullet$'s are of $\mathbf{p}'_{ij}$'s in $\mathcal{B}'$. Then, in both the following examples, $\mathcal{B}'$ would not be a basis since the graph $G_{\mathcal{B}}$ has problematic cycles.

| | 1 | 4 | 5 | 6 |
|---|---|---|---|---|
| 2 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 |
| 5 | 0 | 1 | 1 | 0 |
| 7 | 0 | 1 | 1 | 1 |



Two global constraints, $S_1 + S_2 + S_3 + S_4 \geq 3$ and $E_1 + E_2 + E_3 + E_4 \geq 3$, are deduced from these chains, and thus the optimal cost is $6 = (e-1)+(s-1)$. SQF may be insufficient to ensure that the computed set is a basis. It fails to prune cycles as the one shown below on the right, for SSEDE with $\Gamma = \{(1,3),(3,1)\}$.



However, by enforcing the constraints given by SQF, relevant propagation and pruning is often achieved. So, we decided to firstly label $P$, guarded by such incomplete constraints, and then to check that $P$ defines a basis. In SICStus, we do `minimize(solution(Vars,P,Rank,Ne,Ns,Sigmas),Cost)` with `solution/6` defined as follows, where `Vars` is the list of all the decision variables except the $X_{ij}$'s (so, we had previously a `?` in the solution), with the $P_{ij}$'s in its tail.

```
solution(Vars,P,Rank,Ne,Ns,Sigmas) :-
   labeling([],Vars),  % with no preferential strategy
   basis(P,Rank,Ne,Ns,Sigmas).
```

In real-life, we cannot usually expect an observer to be able to count more than three OD flows even if their destinations are in sight. An extensive simulation for randomly generated roundabouts, with $4 \leq n \leq 10$ and less than four $\star$'s per row, has shown that no-good solutions (i.e., for which `basis/5` fails) were found just in less than 5% of the cases.

When SQF is applied to two given rows $i$ and $i'$, it splits the columns that do not have $\star$'s in rows $i$ and $i'$, in two sets, say $\mathcal{A}_{(i,i')}$ and $\mathcal{B}_{(i,i')}$, according to whether or not they would be incompatible with the first one if the table had no $\star$'s ($j$ and $j'$ are *incompatible* if and only if $S_j + S_{j'} \geq 1$ must hold). All the columns in the same set are pairwise and globally incompatible.

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 |

| | | |
|---|---|---|
| $E_1 + E_2 \geq 1$ | $S_1 + S_2 + S_3 + S_4 \geq 3$ | $\mathcal{A}_{(1,2)} = \{1,2,3,4\}, \mathcal{B}_{(1,2)} = \emptyset$ |
| $E_1 + E_3 \geq 1$ | $S_1 + S_4 \geq 1,\ S_2 + S_3 \geq 1$ | $\mathcal{A}_{(1,3)} = \{1,4\}, \mathcal{B}_{(1,3)} = \{2,3\}$ |
| $E_1 + E_4 \geq 1$ | $S_2 + S_3 + S_4 \geq 2$ | $\mathcal{A}_{(1,4)} = \{1\}, \mathcal{B}_{(1,4)} = \{2,3,4\}$ |
| $E_2 + E_3 \geq 1$ | $S_1 + S_4 \geq 1,\ S_2 + S_3 \geq 1$ | $\mathcal{A}_{(2,3)} = \{1,4\}, \mathcal{B}_{(2,3)} = \{2,3\}$ |
| $E_2 + E_4 \geq 1$ | $S_2 + S_3 + S_4 \geq 2$ | $\mathcal{A}_{(2,4)} = \{1\}, \mathcal{B}_{(2,4)} = \{2,3,4\}$ |
| $E_3 + E_4 \geq 1$ | $S_1 + S_2 + S_3 \geq 2$ | $\mathcal{A}_{(3,4)} = \{1,2,3\}, \mathcal{B}_{(3,4)} = \{4\}$ |

The constraints $E_i + E_{i'} \geq 1$, for all $i$ and $i'$, would be replaced by a single global constraint $E_1 + E_2 + E_3 + E_4 \geq 3$, which the program obtains by computing the *maximal independent sets in a graph*, as we explain below. And, similarly, for $S_1 + S_2 + S_3 + S_4 \geq 3$. As for Praça da República, SQF yields the following,

| | | | |
|---|---|---|---|
| 0 | $\star$ | $\star$ | 0 |
| 0 | $\star$ | $\star$ | 0 |
| 0 | 1 | 1 | 0 |
| $\star$ | 1 | 1 | 1 |

| | | |
|---|---|---|
| $E_1 + E_2 \geq 1$ | $S_1 + S_4 \geq 1$ | $\mathcal{A}_{(1,2)} = \{1,4\}$, $\mathcal{B}^{(1,2)} = \emptyset$ |
| $E_1 + E_3 \geq 1$ | $S_1 + S_4 \geq 1$ | $\mathcal{A}_{(1,3)} = \{1,4\}$, $\mathcal{B}_{(1,3)} = \emptyset$ |
| | | $\mathcal{A}_{(1,4)} = \{1\}$, $\mathcal{B}_{(1,4)} = \{4\}$ |
| $E_2 + E_3 \geq 1$ | $S_1 + S_4 \geq 1$ | $\mathcal{A}_{(2,3)} = \{1,4\}$, $\mathcal{B}_{(2,3)} = \emptyset$ |
| | | $\mathcal{A}_{(2,4)} = \{1\}$, $\mathcal{B}_{(2,4)} = \{4\}$ |
| $E_3 + E_4 \geq 1$ | $S_2 + S_3 \geq 1$ | $\mathcal{A}_{(3,4)} = \{2,3\}$, $\mathcal{B}_{(3,4)} = \{4\}$ |

so that, $E_1+E_2+E_3 \geq 2$ (a clique). If $R_1 \in \{\mathrm{S}, \mathrm{D}\}$, any table of $\sigma_{ij}$'s (wrt $F_1$) has only 0's in the first column. In each row only 0's occur to the right of a second 0. In each column, there are only 1's below each 1. This makes possible to design an elegant algorithm for finding pairs of incompatible columns (or rows). Then, to obtain more global constraints on the $E_i$'s, we consider the complementary relation (of the pairs $(i, i')$ of compatible rows) and find all maximal independent sets with at least two elements in its undirected graph. This is not efficient in general, since finding *all* such sets is an NP-complete problem (e.g. [6]), but it works quite well in our specific case, as the graphs are either small or sparse. In SICStus, we use builtin predicates, `findall(Set,independent_set(UGraph,2,Set),LSets)`, to obtain such sets for a given `UGraph`. To compute the compatibility relations, the program starts from a complete graph with vertices `1..s`, from which it subsequently removes edges that link incompatible columns, while it constructs the graph of the compatibility relation for the rows and imposes (7).

We now look at `basis/5` and see how it may be defined in a declarative way. Let $\mathcal{B}' = \{\mathbf{p}'_{\imath_i\jmath_j} \mid P_{ij} \text{ is labelled to } 1\}$ and let $\mathcal{N}_{\mathcal{B}'} = \{(i,j) \mid \mathbf{p}'_{\imath_i\jmath_j} \in \mathcal{B}'\}$. The main idea is to obtain the largest subset $\mathcal{S} \subseteq \mathcal{B}'$ such that the graph $G_{\mathcal{S}}$ is a tree. Notice that if $\mathcal{B}'$ is a basis then either $\mathcal{S} = \mathcal{B}' \setminus \{\mathbf{p}'_{\imath_o\jmath_d}\}$, for one $\mathbf{p}'_{\imath_o\jmath_d} \in \mathcal{B}'$, or `Rank` is $e+s-1$ and $\mathcal{S} = \mathcal{B}'$. Moreover, $\mathcal{S}$ can be found by a deterministic completion procedure, in which the possible resulting $\mathbf{p}'_{\imath_o\jmath_d}$ is the first node found to violate the tree-shape. It can be done iteratively — at iteration $i$, we try to link the chain of the nodes $(i,j) \in \mathcal{N}_{\mathcal{B}'}$ that were selected in row $i$, adopting a similar idea as that of the Kruskal's Algorithm for minimum spanning trees (e.g. [6]).

```
forest([],Trees,Trees,_,[]).
forest([I-NodesI|Nds],Trees,Treesf,LPod,[I-BasisI|Basis]) :-
 deleting(NodesI,Trees,RTrs,LTrees,I,LPod,BasisI),
 flatten(LTrees,NewTr),forest(Nds,[NewTr|RTrs],Treesf,LPod,Basis).
```

In calls to `forest/5`, `LPod` is bounded to `[]` or `[Pod]`, depending on `Rank`, `Trees` is `[[1],[2],...,[Ns]]`, with `Ns`=$s$, `Treesf` is `[[1,...,Ns]]` and `NodesI` is a list of terms `n(J,SigmaIJ)` that are the selections in row `I`. As suggested by its name, `deleting/7` implements deletion to pick up the tree where each node in `NodesI` is to be linked, gathering these trees (i.e., their vertices) in `LTrees`. It is deterministic. The node `n(J,SigmaIJ)` is linked to the tree that contains `J`, and `forest/5` fails when two nodes in `NodesI` are to be attached to the same tree, because such tree is removed when the first node is found. If `forest/5` succeeds then, when `Rank` is $e+s$, we still have to check that $\mathbf{p}'_{\imath_o\jmath_d}$ (in `Pod`) could not be written as a combination of the elements in $\mathcal{S}$ (in the last argument of `forest/5`). This is dealt by a predicate that trivially succeeds if `LPod=[]`. If `LPod=[Pod]`, it solves $\sum_{\mathbf{p}'_{\imath_i\jmath_j}\in\mathcal{S}} \beta_{ij}\mathbf{p}_{\imath_i\jmath_j} = \mathbf{p}_{\imath_o\jmath_d}$, in $e+s-1$ variables $\beta_{ij} \in \{0,1,-1\}$ and succeeds if and only if $\sum_{\mathbf{p}'_{\imath_i\jmath_j}\in\mathcal{S}} \beta_{ij}\sigma_{ij} \neq \sigma_{\imath_o\jmath_d}$, for the computed $\beta_{ij}$'s. This CSP in finite domains has a unique solution and it is easily definable since $\mathbf{p}_{ij} = \mathbf{e}_i + \mathbf{e}_{e+j}$, where $\mathbf{e}_i$ and $\mathbf{e}_{e+j}$ are unit vectors of the real space $\mathbb{R}^{e+s}$.

**Further Work.** An application of CLP(FD) to solve a real world problem was described, which illustrates how an adequate usage of problem-specific knowledge, namely to deduce global constraints, helps reduce the computational effort. We are now studying whether Constraint Programming may be successfully applied to sensor location problems for Transportation Networks.

**Thanks.** To anonymous referees for helpful comments. This work was partially supported by funds granted to LIACC through the *Programa de Financiamento Plurianual, Fundação para a Ciência e Tecnologia* and *Programa POSI*.

## References

1. Andrade M.: *Métodos e Técnicas de Recolha de Dados de Tráfego – Algoritmo para a Definição da Matriz Origem-Destino*. Msc. Thesis, Faculty of Engineering, University of Porto, Portugal, 2000.
2. Carlsson M., Ottosson G., Carlson B.: An Open-Ended Finite Domain Constraint Solver. In *Proceedings of PLILP'97*, LNCS 1292, 191-206, Springer-Verlag, 1997.
3. Bianco L., Confessore G., and Reverberi P.: A Network Based Model for Traffic Sensor Location with Implications on O/D Matrix Estimates. *J. Transportation Science*, 35(1), 50–60, 2001.
4. Cook W., Cunningham W., Pulleyblank W., and Schrijver A.: *Combinatorial Optimization*. John Wiley & Sons, 1998, Chapter 8.
5. ECLiPSe User Manual Version 5.1.2, IC-PARC, Inperial College, London, 2001.
6. Cormen H., Leiserson C., Rivest R.: *Introduction to Algorithms*, MIT Press, 1990.
7. Freuder E.: In Pursuit of the Holy Grail, *Constraints*, 2(1), 57–62, 1997.
8. Hadley G.: *Linear Programming*. Addison-Wesley, 1969.
9. Hillier S. H., Lieberman G. J.: *Introduction to Operations Research*. McGraw-Hill, 6th Ed, 1995.
10. Marriott K., and Stuckey P.: *Programming with Constraints – An Introduction*, The MIT Press, 1998.
11. Mansfield L. E.: *Linear Algebra with Geometric Applications*, Marcel Dekker, 1976.
12. SICStus Prolog User Manual Release 3.8.6, SICS, Sweden, 2001.
13. Yang H., and Zhou J.: Optimal Traffic Counting Locations for Origin-Destination Matrix Estimation. *J. Transportation Research*, 32B(2), 109–126, 1998.
14. Tomás A. P., Andrade M., and Pires da Costa A.: Obtaining OD Data at Optimal Cost at Urban Roundabouts, Internal Report DCC-2-2001, DCC-FC & LIACC, University of Porto, 2001. `www.ncc.up.pt/fcup/DCC/Pubs/treports.html`. Presented at CSOR'01 Workshop (part of EPIA'01), Porto, December 2001.
15. Tomás A. P.: A Note on Sensor Location for Traffic Counting at Roundabouts – Solutions for a Particular Cost Function, Internal Report DCC-3-2001, DCC-FC & LIACC, University of Porto, 2001.