Very_cool_project.mzn

Valerio Costa, Luca Domeniconi, Claudia Maiolino, Tian Cheng Xia
{valerio.costa, luca.domeniconi5, claudia.maiolino, tiancheng.xia}@studio.unibo.it

1 Introduction

2 CP model

2.1 Decision variables

The CP model relies on the following variables:

- For each package p, $A_p \in \{1, \dots, m\}$ (`assignments[p]` in MiniZinc) indicates which courier delivers it. More specifically, $A_p = c$ iff the package p is delivered by the courier c.
- For each courier c, $P_{c,d} \in \{1, \dots, n+1\}$ for $d \in \{1, \dots, n+1\}$ (`path[c, d]` in MiniZinc) models the path taken by the courier c. The path is defined such that $P_{j,d_1} = d_1$ iff the location d_1 is not part of the route of c and $P_{c,d_1} = d_2$ iff d_2 is visited immediately after d_1 in the route of c. In other words, each relevant location indicates which is its successor and the overall route is defined as a Hamiltonian cycle that starts from $n+1$ (i.e., the depot).

2.2 Objective function

2.3 Constraints

To impose the capacity limits of each courier, the following constraint can be defined:

$$\forall c \in \{1, \dots, m\} : \sum_{p \in \{1, \dots, n\} : A_p = c} s_p \leq l_c \tag{1}$$

In MiniZinc, the global constraint `bin_packing_capa` also models this constraint. In our experiments, we did not notice significant differences between the two formulations and decided to use the latter following the best practice of preferring global constraints.

To model the route, the following constraints have to be imposed:

$$\forall c \in \{1, \dots, m\} : \begin{cases} P_{c,n+1} = n+1 & \text{if } \nexists p \in \{1, \dots, n\} : A_p = c \\ P_{c,n+1} \neq n+1 & \text{if } \exists p \in \{1, \dots, n\} : A_p = c \end{cases} \tag{2}$$

$$\forall c \in \{1, \dots, m\}, \forall p \in \{1, \dots, n\} : \begin{cases} P_{c,p} = p & \text{if } A_p \neq c \\ P_{c,p} \neq p & \text{if } A_p = c \end{cases} \tag{3}$$

The constraint defined in Equation (2) imposes that, for each courier, the depot has a successor only if that courier delivers at least a package. On the same note, Equation (3) imposes that only the packages delivered by a specific courier have a successor in the route. Moreover, it is necessary to impose that the route defined by P_c is a Hamiltonian cycle that passes through the relevant destinations. This can be done by constraining all the elements of P_c to be different and by defining a subtour elimination constraint (e.g., Miller-Tucker-Zemlin [1]). In MiniZinc, this can be easily modelled by using the global constraint `subcircuit`.

2.3.1 Symmetry breaking constraints

The most notable symmetry in this problem is between couriers with the same capacity. In fact, as the assignments between two couriers with the same capacity are interchangeable, it is reasonable to fix an ordering and avoid redundancies during search. We experimented with two approaches:

- By imposing an ordering on the amount of assigned packages:

$$\forall c_1, c_2 \in \{1, \dots, m\} : (c_1 < c_2 \wedge l_{c_1} = l_{c_2}) \Rightarrow Q_{c_1} \leq Q_{c_2} \tag{4}$$

  where Q_c is the amount of packages delivered by the courier c.

- By imposing an ordering on indexes of the assigned packages:

$$\forall c_1, c_2 \in \{1, \dots, m\} : (c_1 < c_2 \wedge l_{c_1} = l_{c_2}) \Rightarrow A_{c_1} <_{\texttt{lex}} A_{c_2} \tag{5}$$

  where A_c is a vector containing the packages delivered by the courier c (in practice, it is a vector of n elements where the i-th position is 0 if the package i is not in the route and i if it is. An alternative formulation for MiniZinc uses a list comprehension and the `deopt` operator, but we observed that this slows down the model).

Moreover, we experimented with a stronger version of these constraints that is applied between two couriers that satisfy the following condition:

$$\max \{L_{c_1}, L_{c_2}\} \leq \min \{l_{c_1}, l_{c_2}\} \tag{6}$$

where L_c is the actual load carried by the courier c.

2.4 Validation

3 SAT model

3.1 Decision variables

3.2 Objective function

3.3 Constraints

3.4 Validation

4 SMT model

4.1 Decision variables

4.2 Objective function

4.3 Constraints

4.4 Validation

5 MIP model

5.1 Decision variables

5.2 Objective function

5.3 Constraints

5.4 Validation

References

[1] Martin Desrochers and Gilbert Laporte. “Improvements and extensions to the Miller-Tucker-Zemlin subtour elimination constraints”. In: *Operations Research Letters* 10.1 (1991), pp. 27–36. ISSN: 0167-6377. DOI: `10.1016/0167-6377(91)90083-2`.