Computational Project - Project Planning

Luiz Fernando Bueno Rosa - RA: 221197
Lucas Guesser Targino da Silva - RA: 203534

June 3, 2022

1 checklist what to do

- Título
  - denominação do problema
  - metodologia de solução
- Resumo
  - objetivos
  - informações sobre o problema
  - metodologia de solução
  - como será feita a avaliação dos resultados
- Introdução
  - descrição formal do problema (formulação matemática)
  - revisão bibliográfica do problema (e/ou problemas relacionados)
  - metodologias previamente utilizadas
- Metodologia
  - justificativa
  - descrição das técnicas de otimização
  - descrever as técnicas de otimização contextualizando-as ao problema de otimização combinatória proposto
- Avaliação dos Resultados
  - proposta de experimentos
  - descrição das instâncias
  - como será feita a avaliação dos resultados
- Referências Bibliográficas

2 Definitions

2.1 Item

An **Item** ι is a 6-tuple:

$$\iota = \langle \chi, \psi, \omega, x, y, z \rangle \tag{1}$$

in which the components represent the item's[1]:

1. $\chi \in \mathbb{R}_+^*$: dimension in the x direction;
2. $\psi \in \mathbb{R}_+^*$: dimension in the y direction;
3. $\omega \in \mathbb{R}_+^*$: dimension in the z direction;
4. $x \in \mathbb{R}_+$: x position;
5. $y \in \mathbb{R}_+$: y position;
6. $z \in \mathbb{R}_+$: z position;

We represent by $\mathcal{I} = \mathbb{R}_+^{*3} \times \mathbb{R}_+^3$ the set of all items.
The reason an items is seen in that way is because of the stacking

2.2 Vehicle

A Vehicle υ is a 3-tuple:

$$\upsilon = \langle \alpha, \eta, L \rangle \tag{2}$$

in which:

1. α is the number of components of the vehicle's loadings;
2. $\eta : \mathcal{I} \to \mathbb{R}_+^\alpha$: a function that associates every item to a vehicle's loading;
3. $L \in \mathbb{R}_+^\alpha$: represents the vehicle's loading limit;

We represent by $\mathcal{V}$ the set of all vehicles.

3 Problem Statement

3.1 Input

1. $I_o \subseteq \mathcal{I}$: the set of items
2. $\upsilon \in \mathcal{V}$: the vehicle

3.2 Constraints

Loading Limit Constraint

$$\sum_{\iota \in I_o} \eta(\iota) \leq L \tag{3}$$

[1] See a definition for "dimension" in [1]

Stacking Constraint

$$\text{An item can only be removed if all items above it have already been removed} \tag{4}$$

We represent whether such constraint is satisfied or not by $\mathcal{SC}\left(I_f\right) \in \{true, false\}$

3.3 Output

A subset $I_f \subseteq I_o$ of the input items.

3.4 Objective

$$\begin{aligned} \min \quad & |I_o| - |I_f| \\ \text{subjected to} \quad & \sum_{\iota \in I_o} \eta\left(\iota\right) \leq L \\ & \mathcal{SC}\left(I_f\right) \end{aligned} \tag{5}$$

Minimize the number of items removed so that all constraints are satisfied.

4 Similarity with the Knapsack Problem

[2] gives a definition for a 0-1 Knapsack Problem (0-1KP):

$$\begin{aligned} \max \qquad & \sum_{j=1}^{n} p_j x_j \\ \text{subjected to} \qquad & \sum_{j=1}^{n} w_j x_j \leq c \\ & x_j \in \{0, 1\} \quad \forall j \in \{1, \ldots, n\} \end{aligned} \tag{6}$$

in which p_j and w_j are know as the profit and the weight of the item j, respectively.
Our problem can be seen as:

1. a Multidimensional Knapsack Problem (MKP) since the weight function is multidimensional;
2. a 0-1KP in which the profit of all items are equal (the exact value of the profit depends on how one decide to formulate the problem);
3. a 0-1KP with an extra restriction on how the items can be inserted (or removed) from the knapsack (constraint (4));
4. it is still a "0-1" problem in the sense that one has to decide whether to remove a specific item or not;

References

[1] Cambridge Dictionary. dimension, 2022.

[2] Silvano Martello and Paolo Toth. *Knapsack problems: algorithms and computer implementations*. John Wiley & Sons, Inc., 1990.