LINFO2266 - Solutions

November 30, 2021

1 Lagrangian Relaxation

1.1 Set Covering Problem

1. We introduce one Lagrangian multiplier for each constraint to obtain the Lagrangian relaxation:

$$\min \sum_{i=1}^{m} c_i x_i + \sum_{j=1}^{n} \lambda_j \left(1 - \sum_{\substack{i=1 \\ j \in S_i}}^{m} x_i \right) \tag{1}$$

   with $\lambda_j \geq 0, j \in \{1, \ldots, n\}$. Note how we integrated the constraints: for any *feasible* solution of the original problem, we will have

$$1 - \sum_{\substack{i=1 \\ j \in S_i}}^{m} x_i \leq 0 \tag{2}$$

   which means a *negative* value will be added to the objective function and that we will thus obtain a *lower bound* on the optimal value of the problem.

   For clarity, we rewrite eq. (1) as:

$$\min \sum_{i=1}^{m} \left(c_i - \sum_{j \in S_i} \lambda_j \right) x_i + \sum_{j=1}^{n} \lambda_j \tag{3}$$

   For fixed values of the Lagrangian multipliers λ_j, we can obtain a lower bound by solving eq. (3). This problem is very simple to solve if we notice that:

   - each x_i is multiplied by a fixed coefficient $\left(c_i - \sum_{j \in S_i} \lambda_j \right)$
   - the second term $\left(\sum_{j=1}^{n} \lambda_j \right)$ is constant

   In order to minimize the objective function, it is sufficient to select the sets which have a negative coefficient:

$$x_i = \begin{cases} 1 & \text{if } c_i - \sum_{j \in S_i} \lambda_j \leq 0 \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

2. By applying exactly what we described above, we get a lower bound value of 4.2.

3. We adapt the subgradient procedure covered in the lectures for this particular problem, and for multiple Lagrange multipliers in algorithm 1.

Algorithm 1 Subgradient procedure for the Set Covering Problem.

$\mathcal{L}^* \leftarrow -\infty, k \leftarrow 1, \mu_0 \leftarrow 1$
$\lambda_{0,j} \leftarrow 0, 1 \leq j \leq n$
$\mathcal{C}^* \leftarrow$ a trivial solution, or none
while $\mu_k \geq \epsilon$ **do**
  Compute cover $\mathcal{C}_k$ with weights $\left(c_i - \sum_{j \in S_i} \lambda_{k,j}\right)$ for $1 \leq i \leq m$
  $\mathcal{L}_k \leftarrow lagrangianValue\,(\mathcal{C}_k)$ (given by eq. (3))
  if $\mathcal{L}_k > \mathcal{L}^*$ **then**
    $\mathcal{L}^* \leftarrow \mathcal{L}_k$
  end if
  if $\mathcal{C}_k$ is feasible and $value\,(\mathcal{C}_k) < value\,(\mathcal{C}^*)$ **then**
    $\mathcal{C}^* \leftarrow \mathcal{C}_k$
  end if
  if $\mathcal{L}^* = value\,(\mathcal{C}^*)$ **then**
    break
  end if
  $\lambda_{k+1,j} \leftarrow \max\left(0, \lambda_{k,j} + \mu_k \left(1 - \sum_{\substack{i=1 \\ j \in S_i}}^{m} x_i\right)\right)$ for $1 \leq j \leq n$
  $\mu_{k+1} \leftarrow \frac{1}{k}$ (or another valid update rule)
  $k \leftarrow k + 1$
end while
return $\mathcal{L}^*$ and $\mathcal{C}^*$

2 Local search

2.1 Pigment Sequencing Problem

In order to formulate the cost invariants, we use the variables $x_p \in I \cup \{idle\}$ which decide which item is produced at period $p \in [0, p_{max}]$ or if the machine is idle.

If we were to change the value of x_p from i to j and with $i, j \in I \cup \{idle\}$ and $d_i, d_j \geq p$ if $i, j \neq idle$ respectively, the cost of the solution would be increased by a Δ given by:

$$\Delta = \Delta_{stocking} + \Delta_{changeover} \tag{5}$$

where $\Delta_{stocking}$ and $\Delta_{changeover}$ respectively denote the variation of the total stocking and changeover costs.

We first define the individual stocking cost:

$$c_{stocking}(x_p = i) = \begin{cases} 0 & \text{if } i = idle, \\ (d_i - p)h & \text{otherwise.} \end{cases} \tag{6}$$

which allows to formulate $\Delta_{stocking}$ easily:

$$\Delta_{stocking} = c_{stocking}(x_p = j) - c_{stocking}(x_p = i) \tag{7}$$

Similarly, we will express the individual changeover cost. To simplify the coming equations, let us define:

$$q(i, j) = \begin{cases} 0 & \text{if } i = idle \text{ or } j = idle, \\ q^{t_i, t_j} & \text{otherwise.} \end{cases} \tag{8}$$

Let p', p'' respectively denote the previous and next periods where an item is produced (i.e. the machine is not idle). Formally, we write:

$$p' = \max\left(\{\pi \mid \pi \in [0, p-1] \text{ and } x_\pi \neq \mathit{idle}\}\right) \tag{9}$$

$$p'' = \min\left(\{\pi \mid \pi \in [p+1, p_{max}] \text{ and } x_\pi \neq \mathit{idle}\}\right) \tag{10}$$

The items produced at those periods, or the value *idle* if there is no such period, are given by $k = x_{p'}$ and $l = x_{p''}$.

$$c_{changeover}(x_p = i) = \begin{cases} q(k, l) & \text{if } i = \mathit{idle}, \\ q(k, i) + q(i, l) & \text{otherwise.} \end{cases} \tag{11}$$

Finally, we can write:

$$\Delta_{changeover} = c_{changeover}(x_p = j) - c_{changeover}(x_p = i) \tag{12}$$