Optimization of foundry production processes

Kelemen Anna
Supervisor: Jüttner Alpár

Eötvös Loránd Tudományegyetem
June 6, 2025

Foundry processes

- casting: shaping molten metal by pouring it into molds


mold making → assembly
core making → assembly
assembly → casting ⇢ finishing operations

Summary

1. IP model
2. LP relaxation $\longrightarrow$ column generation + dynamic programming
3. Rounding

- P: set of products
- A, C: sets of assembly and casting shifts
- $\forall\ p \in P$ - S_p: feasible schedules
- the cost of a schedule $s_{p_i} \in S_p - c_{p_i}$: sum of the idle time between consecutive tasks and the time between the completion of the order and the deadline

IP model

$$\min \sum_{p \in P} \sum_{s_{p_i} \in S_p} c_{p_i} x_{s_{p_i}} + \sum_{c \in C} k_c y_c$$

$$x_{s_{p_i}} \in \{0, 1\} \qquad \forall p \in P, \forall s_{p_i} \in S_p$$

$$y_c \in \{0, 1\} \qquad \forall c \in C$$

$$\sum_{s_{p_i} \in S_p} x_{s_{p_i}} = 1 \qquad \forall p \in P$$

$$\sum_{p \in P} \sum_{\substack{s_{p_i} \in S_p \\ a \in s_{p_i}}} t_p x_{s_{p_i}} \leq T_a \qquad \forall a \in A$$

$$\sum_{p \in P} \sum_{\substack{s_{p_i} \in S_p \\ c \in s_{p_i}}} w_p x_{s_{p_i}} - W_c y_c \leq 0 \qquad \forall c \in C$$

LP relaxation

$$\min \sum_{p \in P} \sum_{s_{p_i} \in S_p} c_{p_i} x_{s_{p_i}} + \sum_{c \in C} k_c y_c$$

$$0 \leq x_{s_{p_i}} \leq 1 \qquad \forall p \in P, s_{p_i} \in S_p$$

$$0 \leq y_c \leq 1 \qquad \forall c \in C$$

$$\sum_{s_{p_i} \in S_p} x_{s_{p_i}} = 1 \qquad \forall p \in P$$

$$\sum_{p \in P} \sum_{\substack{s_{p_i} \in S_p \\ a \in s_{p_i}}} t_p x_{s_{p_i}} \leq T_a \qquad \forall a \in A$$

$$\sum_{p \in P} \sum_{\substack{s_{p_i} \in S_p \\ c \in s_{p_i}}} w_p x_{s_{p_i}} - W_c y_c \leq 0 \qquad \forall c \in C$$

Dual

Variables:

- $\alpha_p \in \mathbb{R} \quad \forall p \in P$
- $\beta_a \geq 0 \quad \forall a \in A$
- $\gamma_c \geq 0 \quad \forall c \in C$

Dual:

$$\min \sum_{p \in P} \alpha_p + \sum_{a \in A} \beta_a T_a$$

$$c_{p_i} + \alpha_p + \sum_{a \in s_{p_i}} \beta_a t_p + \sum_{c \in s_{p_i}} \gamma_c w_p \geq 0 \quad \forall p \in P, s_{p_i} \in S_p$$

$$k_c - \gamma_c W_c \geq 0 \quad \forall c \in C$$

Column generation - pricing problem

$$\min_{s_{p_i} \in S_p} \left\{ c_{p_i} + \alpha_p + \sum_{a \in s_{p_i}} \beta_a t_p + \sum_{c \in s_{p_i}} \gamma_c w_p \right\}$$

- this is equivalent to finding a schedule with the lowest total cost

Pricing problem - DP algorithm

- optimal mold-core-assembly schedule - $O(|A|)$
  - compute best core + assembly
  - compute best mold + assembly
  - compute best mold + core + assembly
- optimal schedule - $O(|A| \cdot |C|)$
  - check all possible assembly shifts for every casting shift before the deadline

Rounding

- after the initial column generation: fractional optimum
- in each round:
  - select K highest value x variables
  - fix the best to 1, fix all the others for the same product to 0
  - fix the corresponding y to 1
  - update columns, column generation
- update shift capacities
- repeat until all products are scheduled

Implementation and Results

Implementation:

- C++
- LEMON

Runtime:

- Few minutes for smaller cases
- $\sim$ 1 hour for about a month's worth of orders

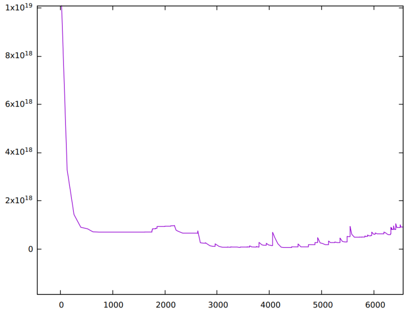

Figure: Change in objective value with the number of columns

Thank you for your attention!