

鉄鋼業における材料引当システムの局所探索アルゴリズム

日本 IBM (株) 東京基礎研究所 *柳沢 弘揮 YANAGISAWA Hiroki
01013100 日本 IBM (株) 東京基礎研究所 岡野 裕之 OKANO Hiroyuki

1 はじめに

鉄鋼業界においては、原則として注文に応じて材料（スラブ・コイル等）を生産するため、製鋼所の各生産工程において注文と材料は関連付いた状態が保持されている。しかし、材料が各工程を通過するうちに、様々な原因により、注文と材料の関係が崩れることがある。こうした場合に、注文に関連付いていない材料を改めて引当てなおすシステムのことを材料引当システムという。注文と材料の関係が崩れることは少なくないため、鉄鋼業界では、材料引当の最適化は重要な問題として認識されている [1]。

我々が構築した材料引当システムは、次の4つの指標に基づいて引当の良さを点数化する。1) 重要度の高い注文をどれだけ満たせたか、2) 引当は相性の良い材料と注文の組で行われているか、3) 優先度の高い材料をどれだけ引当てたか、4) 小さな余剰が発生していないか。小さな余剰は使い道が少なく、廃棄せざるを得ない公算が高いため望ましくない。本稿で我々が実装したシステムでは、[1]で報告されているケースとは異なり、上記の1)や2)の比重が高く、余剰量については高くなかった。

材料引当システムは、様々な制約条件を満たしながら引当を行わなくてはならない。第1の制約条件として、引当可能な注文と材料の組（紐付け）が限定されている（図1）。また、1つの材料は任意の大きさに切断できるので、複数の注文に引当てることができるが、当該紐付け間である種の制約（抱き合わせ可否条件）も満たす必要がある。第2に、各注文には単重（分割後のひとまとまりの重量）の上下限が設定されており、引当てる材料はこの単重の上下限の範囲におさまるように切断して引当てる必要がある（単重制限）。単重制限を満たしていれば、複数種類の材料を用いることは問題ない。第3に、小さな余剰を残すことを避けるために、注文量を超えて引当てることも可能だが、各注文ごとにマージンとして設定されている引当量の上限を超えてはいけぬ。他にも歩留計算に関する考慮条件があるが、本稿では割愛する。

材料引当の最適化は、2部グラフ上の最小コストフロー問題に似た問題だが、複雑なコスト関数や制約のため簡単には解けない。NP困難であることが容易に証明できる。

次節以降で、我々が構築した材料引当システムの詳細について述べる。

2 定式化

入力 以下が入力として与えられる。

- 注文の集合 $O : i \in O$
- 注文の重要度 $w_i : w_i > 0$
- 注文の注文量と引当量の上限 $RQ_i, RQ_{max_i} :$
 $0 < RQ_i \leq RQ_{max_i}$
- 注文の単重上下限 $UQ_{max_i}, UQ_{min_i} :$
 $0 < UQ_{min_i} \leq UQ_{max_i}$
- 材料の集合 $S : j \in S$
- 材料の重要度 $w_j : w_j > 0$

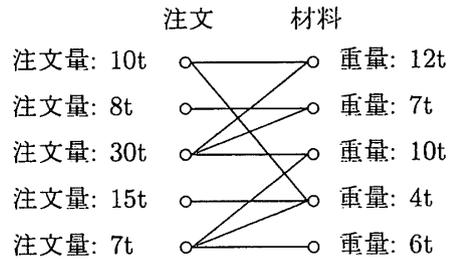


図1: 紐付けの例

- 材料のコスト $c_j : c_j > 0$
- 材料の重量 $SW_j : SW_j > 0$
- 紐付けの集合 $M : (i, j) \in M$
- 紐付けのコスト（相性） $c_{i,j} : c_{i,j} \geq 0$
- 抱き合わせ可否判定関数 $g : M \times M \rightarrow \{true, false\}$

制約条件 以下の制約を満たす必要がある。

注文 $i \in O$ の引当量 AQ_i は、引当量の上限を超えてはならないので、 $AQ_i \leq RQ_{max_i}$ を満たさなくてはならない。また、単重制限を満たすためには、ある整数 n を用いて $n \cdot UQ_{min_i} \leq AQ_i \leq n \cdot UQ_{max_i}$ の形で表せなくてはならない。さらに、抱き合わせ可否条件により、1つの材料を複数の注文に引当てるときは、任意の2つの引当たっている紐付け $m_1, m_2 \in M$ について、 $g(m_1, m_2) = true$ が成り立たなくてはならない。

目的関数 各注文 $i \in O$ の引当量を AQ_i 、各材料 $j \in S$ の消化量を AQ_j 、余剰量を $PS_j (= SW_j - AQ_j)$ 、各紐付け $(i, j) \in M$ の引当量を $AQ_{i,j}$ としたとき、以下の関数で目的関数を定義した。

$$\max z = \sum_{i \in O} w_i \cdot \min\{RQ_i, AQ_i\} - \sum_{(i,j) \in M} c_{i,j} \cdot AQ_{i,j} + \sum_{j \in S} (w_j \cdot AQ_j - c_j \cdot f(PS_j))$$

ここで、関数 $f(x)$ は、以下で定義される関数である。

$$f(x) = 100x^{0.3} \cdot \exp(-0.05x^3)$$

図2で示されるように、小さな余剰（およそ5t未満の余剰）に対して、スコアが悪くなるように定義している。

我々が構築した材料引当システムでは、上記目的関数を最大化する引当を探索する。相対的に、注文の重要度 w_i 、紐付けのコスト（相性） $c_{i,j}$ 、現品の重要度 w_j の順に絶対値が小さくなるので、各注文 $i \in O$ について引当量が注文量 (RQ_i) に近いほど目的関数値が大きくなる。

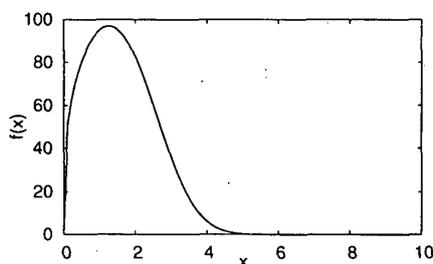


図 2: $f(x)$ の関数形

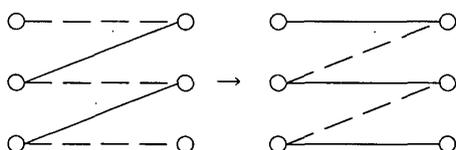


図 3: 近傍操作の例

3 アルゴリズム

我々は、初期解を生成し、局所探索を行うアルゴリズムで材料引当システムを実装した。

初期解の生成アルゴリズムは2種類用意した。一つは、引当可能な紐付けを適当な順序で引当てていく First Fit アルゴリズムである。もう一つは、単重制限や引当可否判断は無視し、各注文の引当量の上限を注文量に設定して線形計画問題として定式化し、その解をラウンディングする手法である。ラウンディングは、線形計画問題の解をもとに、各材料ごとに引当可能な組み合わせを全探索し、最も目的関数値が高くなる紐付けを選んで引当て方式を選択した。

局所探索には、計 11 種類の近傍操作を用意した。まず、1~3 本の引当を解除し、他の紐付けに引当を組み替える近傍操作を7種類用意した(図3に一例)。そして、1本の引当を組み替えることで生まれる空きを利用して別の引当をその空気に引当てる ejection chain と呼ばれる近傍操作も用意した。さらに、1つの注文あるいは現品に着目して、あらゆる引当を探索し、改善する近傍操作も3種類用意した。多数の近傍操作を用意して、目的関数値の改善に努めた。

実装にあたっては、目的関数値は引当量の差分をもとに計算する等、高速化を図った。目的関数が線形ではないため、複数の紐付けの引当量が変わったとき、単純に目的関数値の差分を足し合わせることはできなかったが、必要最小限の値について再計算するよう工夫した。

4 評価

実データをもとに生成したデータ(注文数: 3116, 材料数: 4443, 紐付け数: 94025)を用いて、実装したアルゴリズムの実行時間を評価した。実行環境には、CPU: Pentium 4 3.2GHz, メモリ: 3GB, OS: Windows XP の PC を用いた。線形計画問題のソルバーには、COIN [2] の CLP を使用した。

まず、2つの初期解生成アルゴリズムを比較し、結果を以下の表に載せた。どちらのアルゴリズムを使用しても、トータルの実行時間は10分未満であり、実用上、十分な速度で動作した。2つの初期解生成アルゴリズムの目的関数値を比べると、First Fit に比べ、線形計画問題の解をラウンディングする手

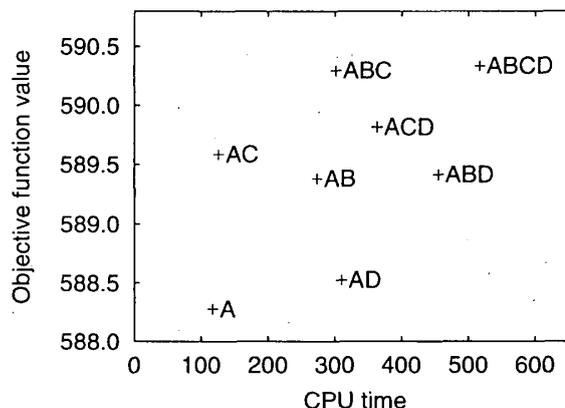


図 4: 実行時間と解の質の分布

法を用いたものの方が、目的関数値が僅かに上回っている。他の実験データについても同様の傾向が見られたため、本番環境では、線形計画問題の解を利用して初期解を求めるアルゴリズムを採用した。

	目的関数値	実行時間 (秒)
First Fit	589.78234	560
CLP	590.33427	572

次に、11種類の近傍操作を以下のようにグループ分けした。

A: 1~2本の引当を解除し、2本以内の新たな紐付けを引当てる近傍(4種類)

B: 2~3本の引当を解除し、3本以上の新たな紐付けを引当てる近傍(3種類)

C: ある注文あるいは材料に着目して、その注文あるいは材料を含むあらゆる引当を探索する近傍(3種類)

D: ejection chain 近傍(1種類)

Aグループ以外の近傍について、それぞれのグループの近傍操作を含める場合と含めない場合について実行し、その実行結果を図4上にプロットした。近傍を追加することによって、解の質(目的関数値)に向上が見られることがわかる。また、ejection chain によって実行時間が約1.5倍~2倍になるが、得られる目的関数値の向上はB, Cグループの近傍操作と比較して、小さいことがわかる。これは、ejection chain の近傍操作が、1つの注文・材料の引当を1組しか引換えないのに対し、B, Cグループの近傍操作では、1つの注文・材料の引当でも複数組の引当を引換えることが多いからである。

5 おわりに

材料引当システムを、局所探索アルゴリズムを用いて構築した。今後は、局所探索のさらなる高速化・探索範囲の拡大や、大域的最適化を試みることで、より最適解に近い解を得られるようにしたい。

参考文献

- [1] 三留立実. 鉄鋼業の生産管理システムへの最適化手法適用. 日本オペレーションズ・リサーチ学会 2004 年秋季研究発表会アブストラクト集, pp. 16-17, 2004.
- [2] Computational Infrastructure for Operations Research (<http://www.coin-or.org/>)