

A_025

ロットグルーピング問題への最適化技術の適用事例 Case Study on the Application of Optimization Method to Lot Grouping

手塚 大† 伊藤 俊明† 山崎 伸晃† 宗形 聡†
TEZUKA Masaru ITO Toshiaki YAMAZAKI Nobuaki MUNAKATA Satoshi

1. はじめに

本報告では、生産計画システムの一部であるロットグルーピングシステムに最適化技術を適用した事例を紹介する。本システムは経験に基づくヒューリスティック手法を用いた旧システムのリプレイスであり、遺伝的アルゴリズム(GA)を用いて最適化を実現する予定であった。しかし規定の計算時間内で旧システムよりも良い結果を出すことができなかった。

そこで調査を行い、探索領域の絞込み、部分問題への分割、適切な問題エンコード法の選択によってシステムの改善を行った。この結果、旧システムを大幅に上回る良い解を実現した。

2. ロットグルーピング問題

現実の生産計画では段取り替え(Setup)を考慮する必要がある。段取り替えとは製造装置の設定変更や補助具の交換作業などである[1]。製造装置は段取り替えを一度行えば無人で連続して大量の製品を製造できるが、段取り替えには長い時間と人手を必要とする。従って製造効率向上のためには段取り替えの発生を少なくする必要がある。

例えば電子装置組み立てに使われるインサータという装置は、カートリッジに部品を装填すると自動的に電子基板に部品を取り付ける。この装置はカートリッジへの部品装填とカートリッジ交換に人手が必要で時間を要するが、一度カートリッジを装填すれば短時間で多数の部品を取り付けられる。したがって、できるだけ同じ部品を使用する基板を連続して流すことでカートリッジ交換の回数および人手を減らし、製造効率を向上できる。

電子部品には同じ働きをする複数の代替部品があり、製品仕様により使用可能な代替部品が決まる。例えば60MHzまで動作保証されている部品と100MHzまでの部品があるとすると、基板の動作クロックが60MHzである場合、どちらの部品を使用しても動作が保証される。

インサータは一度に1枚の基板あたり多数の部品を取り付ける。したがってカートリッジに装填する代替部品の組合せの数は多数存在する。このような代替部品の組合せをリソースと呼ぶことにする。

製造の単位をロットとよび、計画対象のロットの集合を L とする。また、全てのリソースの集合を R とする。ロット $i \in L$ に使用可能なリソース(代替部品の組み合わせ)の集合を $R_i \subseteq R$ とし、これをロット i の代替リソース集合と呼ぶ。また、ロット i について選択されたリソースを $s_i \in R_i$ とする。

二つのロット $i, j \in L$ について $s_i = s_j$ のとき、これらのロットを同じグループにまとめる。すなわち、ロ

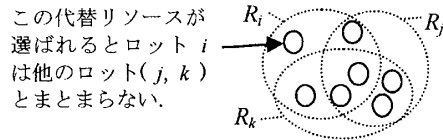


図1: 他のロットとまとまらない代替リソース

ットのグループを G_k とすると、

$$\forall i, j \in G_k, s_i = s_j \quad (1)$$

$$\forall k \neq l, G_k \cap G_l = \phi \quad (2)$$

$$\bigcup_k G_k = L \quad (3)$$

となる。ロットのグループの総数は各ロットのリソースの選び方で変わるが、これを n_G として、

$$\sum_{k=1}^{n_G} \delta \left(\sum_{i \in G_k} f(i) < c \right) \quad (4)$$

を最小化したい。ここで、 $\delta(x)$ は x が真の時に1、偽の時に0である。 $f(i)$ はロット i の関数であり c は何らかの定数である。例えば $f(i)$ はロット i の加工時間であり、総加工時間が定数 c 未満となるようなロットのグループが少なくなるようにグルーピングするのが目的である。

3. 最適化性能改善手法の提案

3.1 現状システムの分析

旧システムは業界の経験に基づく工夫がされた局所探索手法によってロットグルーピングを行っていた。この手法にはアニーリング法やタブーサーチのような局所解を脱出するための仕組み[2]が備わっていない。旧システムで作られた解は、グループにまとまらないロットが多数残り、人手によるグルーピング修正が必要とされていた。そこで、局所解を抜け出すことができるGAを用いて最適化を行う新システムの開発を行った。

しかし、規定の計算時間内で旧システムよりも良い結果を得ることができなかった。このため最適化およびその周辺のプログラムの調査、分析を行った。

3.2 探索領域の絞込み

各ロットの代替リソース集合の中には、図1のように他のロットでは使われることのないリソースが存在する場合がある。このようなリソースが選択されると、このロットは他のロットとまとまらない。そこで、このようなリソースを探索から除外することで探索領域を絞り込み、グルーピングを効率化した。

各ロット $i \in L$ について次式により、他のロットでも使用されているリソースのみを持つ代替リソース集合 R

†(株)日立東日本ソリューションズ

を求め、ここからリソースの選択を行うようにする。

$$\hat{R}_i = \bigcup_{j \neq i} (R_i \cap R_j) \quad (5)$$

ところで $\hat{R}_i = \phi$ となる場合、このロット i はどのようにリソースを選んでも他のロットとまとまらないので計画対象ロット集合 L から除外した。

3.3 部分問題への分解

大規模な問題は、部分問題に分割し部分問題ごとに解くことで効率化ができる。ロットのグルーピング問題では、同一の代替リソースを使用することがない部分問題に分割が可能である。

すなわち、計画対象ロットの集合 L を共通部分の無い複数の部分集合に分割し、部分集合ごとにグルーピングを行う。部分集合を L_1, L_2, \dots とすると、部分問題となる部分集合は次式を満たす。

$$\forall s, t, \left(\bigcup_{i \in L_s} R_i \right) \cap \left(\bigcup_{j \in L_t} R_j \right) = \phi \quad (6)$$

$$\forall s \neq t, L_s \cap L_t = \phi \quad (7)$$

$$\bigcup_s L_s = L \quad (8)$$

グラフの連結性を求めるアルゴリズムなどを用いて、このような問題の分割が可能である。

3.4 エンコーディングの変更

GA による最適化ではエンコーディング法によって最適化性能が大きく変わることが知られている。

改良前の新システムでは各ロットの選択するリソースを直接エンコーディングしていた。この方法は染色体の遺伝子座が一つのロットに対応し、遺伝子値が各ロットの選択するリソースを示す。染色体長は計画対象ロットの総数 $|L|$ であり、ロット i に対応する遺伝子座の遺伝子値は $|R_i|$ 種類の値を取り得る。

この方法では、複数のロットが同じリソースを選択する確率が低く、なかなかグルーピングが進まないことが観察された。例えば、ロット i と j が同じグループとなる確率は $|R_i \cap R_j| / (|R_i| \times |R_j|)$ である。

そこで改良案として、まとまるロットはできるだけまとめてしまうという方針の下で順列エンコーディングを採用することとした。この方法では染色体長はリソースの総数 $|R|$ であり、染色体はリソースの順列となる。図2の手順で、この染色体からロットのグルーピングを行う。図中では順列中で出現する順に、先頭のリソースを r_1 、2番目を r_2 、... としている。

4. 数値実験

探索領域の絞込み、部分問題への分割、問題エンコード法の変更による改良の結果を表1に示す。

各値は総ロット数に対する式(4)の値の割合、すなわち連続処理時間が定数 c 未満となり、連続処理時間条件を満たせなかったロットグループの割合である。値が小さいほど良い。改良前は良い結果を出せず、B工場では旧システムと同じ解であった。これを改良し大幅な改善を実現した。

1. $\hat{L} = L$ とする。
2. for $u = 1$ to $|\hat{R}|$
3. for all $i \in \hat{L}$
4. /*ロット i がリソース r_u 使用可能なら*/
5. /* r_u を選択する。(r_u でまとめる) */
6. if $r_u \in R_i$ then
7. $s_i = r_u$
8. $\hat{L} = \hat{L} \setminus i$
9. end if
10. end for
11. end for

図2: 順列エンコードからのグルーピング

表1: 条件を満たせなかったロットグループの割合

	旧システム	新システム	
		改良前	改良後
A工場	23.5%	22.8%	11.2%
B工場	9.2%	9.2%	3.7%

5. おわりに

本報告ではロットグルーピングの最適化の性能改善の事例を紹介した。ここでは最適化アルゴリズム自体の改良は行わず、また、最新のアルゴリズムの導入なども行っていない。探索領域の絞込み、部分問題への分割、適切なエンコードの使用により性能改善を実現した。

探索領域の絞込みや部分問題への分割は最適化の研究者にとっては当然のことであるが、システム開発者には気づきにくい。このような、アルゴリズムそのものの性能とは無関係な部分が原因で、そのアルゴリズム自体が使い物にならないという誤った判断が、開発現場で行われる恐れもある。最適化ライブラリの開発時には、探索の絞込みや問題分割を自動的に行うような仕組みも合わせて提供することで実用的になるものと考ええる。また、リンケージ同定 [3] のように問題知識によらず分割する手法の導入も有効と考える。

今回は試行錯誤によって適切なエンコード法を決定したが、何らかの体系だったエンコード選択方法を今後検討したい。

参考文献

- [1] John H. Blackstone and James F. Cox. *APICS dictionary 11th ed.* APICS The Association for Operations Management, 2005.
- [2] Colin R. Reeves. モダンヒューリスティクス. 日刊工業新聞社, 1997.
- [3] 棟朝雅晴. エピスタシス尺度に基づくリンケージ同定手法の提案. 情報処理学会論文誌「数理モデル化と応用」, Vol. 43, No. SIG10(TOM7), pp. 6-13, 2002.