2 機械自動生産システムの最適運用計画* (最適工具配分問題について)

木 瀬 洋^{*1}, 軽 野 義 行^{*2}

Optimal Operation Plannings for Automated Two-Machine Manufacturing Systems (On an Optimal Tool Assignment Problem)

Hiroshi KISE and Yoshiyuki KARUNO

This paper considers an optimal tool assignment problem for automated two-machine flowshops such as FMS's (Flexible Manufacturing Systems) and FMC's (Flexible Manufacturing Cells) in order to improve their system efficiencies. This study is motivated by the fact that the system efficiency decreases as the imbalance of machine loads or the diversity of job processing times increases even under an optimal job schedule. Machine loads and job processing times on each machine are determined directly by assigning tools to two machines. Thus we adopt the weighted sum of two variances of machine loads and job processing times as the objective function. This is an extension of the traditional machine load problem. For this problem we propose an approximation algorithm based on simulated annealing technique, and strengthen it by strategically applying the optimal scheduling. Results of numerical experiments performed show that tool assignments and schedules by the proposed algorithm give quite high system efficiencies with small variances.

Key Words: Production Planning, FMS, FMC, Tool Assignment, Scheduling, Approximation, Simulated Annealing, System Efficiency

1. 結 論

本研究は、同種の MC (マシニングセンタ) 2 台と 1 台の AGV (無人搬送車) あるいは RH (ロボットハンド) を中心としてジョブの処理を行うフローショップ型加工システムを対象とし、その最適生産計画の一端として最適工具配分問題を考察する。MC は、その工具マガジンに多数の工具を同時に保有することによって自動化された多種加工を行い、効率のよい多種少量生産を目指す FMS や FMC において中心的な役割を果たす。また、同種の MC を用いることは生産効率およびシステムの信頼性を向上させ、したがって無人運転の実現が期待できる。

最適工具配分問題はシステム稼働率の向上を目的として、必要な工具を複数の MC へ配分する問題である。工具配分によって各 MC のジョブ処理時間が決定されるが、スケジューリングを行わない限りシステム稼働率は決定されない。工具配分から見た 2 台の MC の運用法として、(i) 各 MC 単独で割当てられたジョブの全処理を行う並列型、(ii) 各ジョブの処理を 2 台

の MC に分散するフローショップ型、(iii) (i) と (ii) の混合型、の 3 通りが考えられる。(i) に対する工具配分問題については多くの研究報告がある(たとえば、Berrada ら⁽²⁾)。 (ii) の場合については長沢ら⁽³⁾によって検討されている。本研究もフローショップ型を対象としているが、AGV の搬送時間、MC 間の仕掛り在庫(パレット数)など FMS 特有の制約を考慮した点が長沢らのシステムと異なる。

さらに、最適スケジュールの下でも稼働率はジョブ処理時間のばらつきとともに低下し、逆にこれが小さいと分枝限定法でも短時間で厳密解が得られやすい。そこで本研究では、機械負荷のみならずジョブ処理時間の均一化も目指す工具配分を検討する。すなわち、これらの分散の重み和(以後、アンバランス UB と記す)を種々の実際的な制約条件のもとで最小化する。

この問題は 2 次の 0-1 整数計画問題となり、効率的な厳密解法は望みにくい。よって SA 法(シミュレーテッドアニーリング)に基づく近似解法を検討する。SA 法は反復改善法(または、局所探索法)に確率的要素を取入れたものであり、改善解方向へも探索を行うため、一般に膨大な計算時間を要求する⁽⁴⁾。このため、実用化に対して、(i) 効率探索による反復回数の減少、(ii) 解の生成および目的関数値計算の簡単化につ

* 原稿受付 平成 4 年 8 月 3 日。

^{*1} 正員、京都工芸繊維大学工学部 (〒606 京都市左京区松ヶ崎御所海道町)。

^{*2} 京都大学工学部 (〒606 京都市左京区吉田本町)。

いての工夫が必要である。本研究では(i)のために Matsuo らによって開発された CSSA 法⁽⁵⁾を利用する一方、(ii)のために目的関数値の計算量を2次から1次のオーダーに減らす方法を示す。

また、前述したように工具配分の本来の目的はシステム稼働率向上であるから、本研究では近似解法の探索過程で最良解が得られるごとにスケジューリングを行い、稼働率を計算する手法をとる。UB 最小化と稼働率最大化は高い相関を有するものと考えられるが、これらは必ずしも一致しないのでこの手法は有効である。実際、広範な数値実験により従来の機械負荷均一化よりも常に安定した高い稼働率を与えることを実証する。

2. システムの記述

図1に示すように、対象となるシステムはローディングステーション S_l 、アンローディングステーション S_u 、2台の機械 M_a 、 M_b 、および1台の無人搬送車 AGV からなるフローショップ型 FMS である。ここで、各機械 $M_k(k=a, b)$ が有する APC(自動パレット交換装置)を仕掛品用のバッファ B_k とする。

この加工システムの最適スケジューリング問題に対しては、既に任意のバッファ容量に対応できる近似解法(TSP-J 法)を提案するとともに、この近似解法を利用して、推定期待誤差 1% 以下の解を極めて短い計算時間で与える分枝限定法アルゴリズムを構築した⁽¹⁾。以下の工具配分問題においてシステム稼働率が計算される際にはこのアルゴリズムが適用される。

3. 最適工具配分問題

3.1 問題の定式化 以下のパラメータが与えられているものとする。

$J=\{1, 2, \dots, n_j\}$: 加工されるジョブの集合。

$T=\{1, 2, \dots, n_T\}$: 2台の機械に配分される工具群の集合。ここで、各工具群は使用頻度が高い、脆弱などの理由により、複数の同一工具からなることを許す。

$s(t)$: 工具群 t に必要な工具マガジンのスロット数。

$A(t)$: 技術的制約により、工具群 t の後に使用され

る工具群の集合。

$G(t)$: 加工効率上、工具群 t と同一機械で使用される工具群の集合。

$H(t)$: 加工効率上、工具群 t と異なる機械で使用される工具群の集合。

$q_k(k=a, b)$: 機械 M_k の工具マガジン容量(スロット数)。

$p'(t, j)$: 工具群 t によるジョブ j の処理時間。各々を処理時間要素と呼ぶ。

$p'(t)=\sum_{j=1}^{n_j} p'(t, j)$: 工具群 t の負荷。

$p(j)=\sum_{t=1}^{n_T} p'(t, j)$: ジョブ j の総処理時間。

α : 重み係数 ($0 \leq \alpha \leq 1$)。

なお、ここで $\sum_{t=1}^{n_T} s(t) \leq \sum_{k=a, b} q_k$ を仮定する。他方、決定変数は次のとおりである。

$X=[x(1), x(2), \dots, x(n_T)]$: 工具配分ベクトル。

ここで工具群 t を機械 M_a に配分するならば $x(t)=1$ 、工具群 t を機械 M_b に配分するならば $x(t)=0$ とする。

$p_a(j)=\sum_{t=1}^{n_T} p'(t, j)x(t)$: ジョブ j の機械 M_a での処理時間。

$p_b(j)=\sum_{t=1}^{n_T} p'(t, j)\{1-x(t)\}$: ジョブ j の機械 M_b での処理時間。

$L_k=\sum_{j=1}^{n_j} p_k(j)$: 機械 $M_k(k=a, b)$ の負荷。

$VP(X)=\sum_{k=a, b} \sum_{j=1}^{n_j} \{p_k(j)-P_{av}\}^2/(2n_j)$: ジョブ処理時間の分散。……………(1)

$VL(X)=\sum_{k=a, b} \{L_k-L_{av}\}^2/2$: 機械負荷の分散。……………(2)

ここで、 $TP=L_a+L_b$ は全処理時間、 $L_{av}=TP/2$ は平均負荷および $P_{av}=TP/2n_j$ は平均処理時間を表す。

最適工具配分問題 TAP(α) を以下のように定義する。

TAP(α):

Min. $UB(\alpha, X)=\alpha VP(X)+(1-\alpha) VL(X)$ ……………(3)

S. T. $\sum_{t=1}^{n_T} s(t)x(t) \leq q_a$, ……………(4)

$\sum_{t=1}^{n_T} s(t)\{1-x(t)\} \leq q_b$, ……………(5)

$x(t) \geq x(t'), \forall t' \in A(t), \forall t \in T$, …(6)

$x(t)+x(t')-2x(t)x(t')=0$,
 $\forall t' \in G(t), \forall t \in T$, ……………(7)

$x(t)+x(t')=1, \forall t' \in H(t), \forall t \in T$,
……………(8)

$x(t) \in \{1, 0\}, \forall t \in T$. ……………(9)

目的関数 $UB(\alpha, X)$ は以下の二次形式に置き換えることができる。

$$UB(\alpha, X)=\sum_{s=1}^{n_T} \sum_{t=1}^{n_T} q(\alpha, s, t)x(s)x(t)$$

図1 有限バッファのある2機械自動生産システム

表 1 処理時間要素 $p'(t, j)$ の問題例

$t \backslash j$	1	2	3	4	5	6	7	8
1	2	2	3	5	3	6	8	5
2	3	2	5	0	3	6	4	4
3	7	3	3	5	8	1	6	0
4	3	5	3	4	3	5	0	4

(a) $X=[1, 1, 0, 0]$ with $VP(X)=6.9$ and $VL(X)=0.25$ (b) $X'=[1, 0, 0, 1]$ with $VP(X')=4.9$ and $VL(X')=0.25$

図 2 稼働率に対する二つの工具配分の比較

$$-\sum_{t=1}^T h(\alpha, t)x(t) \quad \dots\dots\dots (10)$$

ここで,

$$q(\alpha, s, t) = \{\alpha \sum_{j=1}^n p'(s, j)p'(t, j) + (1-\alpha)n_j p'(s)p'(t)\}/n_j, \quad \dots\dots\dots (11)$$

$$h(\alpha, t) = \{\alpha \sum_{j=1}^n p(j)p'(t, j) + (1-\alpha)n_j p'(t)TP\}/n_j, \quad \dots\dots\dots (12)$$

である。これは、式(1)、(2)および(3)から直接導出される。

3.2 問題の具体例 表1の問題例に2通りの工具配分 $X=[1, 1, 0, 0]$, $X'=[1, 0, 0, 1]$ を実行する。このとき $VP(X)=6.9$, $VL(X)=0.25$, $VP(X')=4.9$, $VL(X')=0.25$ となる。次に X , X' によって決定される処理時間に同じスケジュールを適用したときのゼロバッファシステムの挙動を図2にガントチャートで示す。この場合、明らかに後者の工具配分を採用するほうがブロッキングやアイドルングを少なくでき、 $VP(X)$ や $VL(X)$ の減少が稼働率に対して有効といえる。

4. 近 似 解 法

4.1 近傍解の生成 本研究で適用するSA法において現在解 $X=[x(1), x(2), \dots, x(n_T)]$ に対する近傍解 $X'=[x'(1), x'(2), \dots, x'(n_T)]$ を以下の(a)~(c)のいずれかを満たす実行可能解〔式(4)~(9)参照〕と定義する。ただし、一般性を失うことなく、 $\exists i, j \in T$ に対して、 $x(i)=1, x(j)=0$ とする。

$$(a) \quad x'(i)=0, x'(j)=1, x'(k)=x(k), k \neq i, j$$

$$(b) \quad x'(i)=0, x'(j)=0, x'(k)=x(k), k \neq i, j$$

$$(c) \quad x'(i)=1, x'(j)=1, x'(k)=x(k), k \neq i, j$$

このとき、 X の近傍解の集合 $N(X)=\{X'\}$ の大きさは

$$|N(X)| \leq \Lambda \equiv \lfloor (n_T^2/4) + n_T \rfloor \quad \dots\dots\dots (13)$$

を満足する(証明略⁽⁶⁾)。ただし、 $\lfloor i \rfloor$ は i を越えない最大の整数を表す。

次に、 X' , X の目的関数値 $UB(\alpha, X')$, $UB(\alpha, X)$ の差は、式(10)~(12)より

$$\begin{aligned} \Delta UB(X', X) &\equiv UB(\alpha, X') - UB(\alpha, X) \\ &= \{q(\alpha, j, j) - h(\alpha, j)\}x'(j) \\ &\quad - \{q(\alpha, i, i) - h(\alpha, i)\}\{1 - x'(i)\} \\ &\quad + 2\sum_{t \neq i, j} [q(\alpha, j, t)x'(j) \\ &\quad - q(\alpha, i, t)\{1 - x'(i)\}x(t) \\ &\quad + 2q(\alpha, i, j)x'(i)x'(j)], \quad \dots\dots\dots (14) \end{aligned}$$

となる(証明略⁽⁶⁾)。 $\Delta UB(X', X)$ は $x(t)$ の一次式であることに注意すれば、 $UB(\alpha, X')$ の計算が簡単化されることがわかる。

4.2 PSA-CS法(Pseudo Simulated Annealing Combined with Scheduling)

TAP(α) に対して以下で示される近似解法のおもな特徴は次のとおりである。

(a) $\Delta UB(X', X) > 0$ となるとき、現在解 X を近傍解 X' に更新する(受容)確率は

$$AP(k) = 1.0 - k/K, k=0, 1, 2, \dots, K, \dots\dots\dots (15)$$

で与え、 $\Delta UB(X', X)$ には依存しないものとする。ここで、 k はSA法における温度(あるいは、コントロールパラメータ)の代わりに用いられるステージ番号である。ただし、 K はあらかじめ与えられた正整数である。これは Matsuo らのCSSA法⁽⁶⁾に準拠している。

(b) 各ステージで探索される近傍解の数は、ステージ番号に関係なく一定値 L とする。このため一般のSA法でいう平衡状態が保証されず、その意味でこのSA法は擬似的(Pseudo)である。

(c) アルゴリズムはそれまでに得られている最良解 X_B をつねに保持し、また各ステージの初期解はその X_B とする。ただし、 $k=0$ における初期解はランダムに選ぶ。上記(a)(b)は計算時間のみを重視しているが、これにより、解の安定性に対する改善を図る。

(d) 最良解 X_B が更新されるごとにスケジュールリングを行い、システム稼働率

$$E_f = [n_f P_{av} / F_{\max}] * 100(\%) \quad \dots\dots\dots (16)$$

を計算する。ただし、 $F_{\max}$ は最大完了時間を表す。このとき、もし、より高い E_f が得られれば、稼働率に

関する最良解 X_E を X_B に更新する。 $UB(\alpha, X)$ 最小化と E_f 最大化は高い相関を有すると考えられるが、必ずしも一致しないので、この手続きは E_f 最大化にとって有効である。

Algorithm PSA-CS :

```

begin
  X := an initial feasible assignment ;
  XB := X ;   XE := XB ;
  for k := 0, 1, ..., K do
    begin
      X := XB ;
      for l := 1, 2, ..., L do
        begin
          X' := a feasible neighbor of X ;
          ΔUB(X', X) := UB(α, X') - UB(α, X) ;
          if ΔUB(X', X) ≤ 0 then
            begin
              X := X' ;
              if UB(α, X) ≤ UB(α, XB) then
                begin
                  XB := X ;
                  if Ef(XB) > Ef(XE) then XE := XB ;
                end ;
            end ;
          end ;
        end ;
      else
        begin
          AP(k) := 1.0 - k/K ;
          r := a random number generated from
                the uniform distribution on [0, 1] ;
          if r ≤ AP(k) then X := X' ;
        end ;
      end ;
    end ;
  end ;
end ;

```

5. 数 値 実 験

5.1 予備実験 PSA-CS 法の評価実験を行う前に、重み係数 α 、ステージ数 $(K+1)$ および各ステージでの反復数 L を決定するため、スケジューリングを伴わないアルゴリズム (PSA と記す) について予備実験を行った。ただし、以下の実験で処理時間要素 $p(t, j)$ (表 1 参照) はあらかじめ与えられた工具使用率

$$TUP = \sum_{j=1}^n \{ [t | p'(t, j) \neq 0, t \in T] / n_T \} / n_j$$

に従って、零または $[7, 97]$ の整数型一様乱数で決定する (この場合、 $p'(t, j) \neq 0$ の平均は $\bar{p}' = 50$ 、変動係数

は $CV = 0.5$ になる)。また、すべての工具群 t に対し、 $s(t) = 1, A(t) = G(t) = H(t) = \{\phi\}, q_a = q_b = n_T/2$ とする [式 (4) ~ (9) 参照]。なお、予備実験の結果はすべて 20 例題を適用したものである。図 3 はステージ数 $(K+1)$ に対する $\alpha = 0.4$ の UB [式 (3)] の変化を示したもので、 $L = 1$ とした [式 (13) 参照]。 $K \geq 25$ で大きな変化はなく、この傾向は他の $\alpha (0.2 \leq \alpha \leq 0.8)$ についても同様である⁽⁶⁾。よって、以後の実験では $K = 25$ とした。図 4 は $SIZEFACTOR = L/1$ に対する UB の変化を示したもので、 $SIZEFACTOR \geq 1.0$ で大きな変化は見られない。この傾向は他の $\alpha (0.2 \leq \alpha \leq 0.8)$ についても同様である⁽⁶⁾。よって以後の実験では $L = 1$ とした。(なお、図中の UB の値はある基準からの修正値である⁽⁶⁾。)

図 5 は式 (3) の重み係数 α が PSA 法による工具配分のもとでジョブ処理時間の変動係数 $CV(P) = \sqrt{VP(X)} / P_{av}$ および機械負荷の変動係数 $CV(L) = \sqrt{VL(X)} / L_{av}$ に及ぼす影響を示している。また、比較のためランダム工具配分のもとでの結果を示している。当然のことながら $\alpha = 0.0, 1.0$ の両極端の場合には

図 3 ステージ数に対する目的関数値の変化

図 4 各ステージの反復数に対する目的関数値の変化

$CV(P)$, $CV(L)$ のいずれか一方が極端に悪い。他方, $0.2 \leq \alpha \leq 0.8$ では $CV(P)$, $CV(L)$ とともに比較的安定している。表 2 は, 2 種類のスケジュールのもとでの稼働率 E_f (式(16)) に及ぼす影響をシステムのバッファが無い場合と大きい場合の両方に対して示している。この場合も, 両極端の α を除けば, α が E_f に及ぼす影響は小さい。これらの結果から, α の最適選択という問題には深刻にならずに済ませられ, 以後の実験では $\alpha=0.4$ とした。

5・2 PSA-CS 法の評価 図 6 は稼働率向上のために提案した PSA-CS 法による工具配分とスケジューリングの同時最適化の効果を調べるため, APC の平均バッファ容量 $C[(C_a + C_b)/2]$ に対する稼働率 E_f の平均値とその標準偏差の変化を示したものである。比較のため, 次の手法も検討した。

MLB/OS : 分枝限定法により機械負荷の均一化

を最適化した後, 最適スケジューリングを行う。

RTA/OS : ランダム工具配分の後, 最適スケジューリングを行う。

OTA/FCFS: PSA 法 ($\alpha=0.4$) による工具配分の後, FCFS スケジュールでジョブの処理を行う。

MLB/FCFS: 機械負荷の均一化を最適化した後, FCFS スケジュールでジョブの処理を行う。

RTA/FCFS: ランダム工具配分の後, FCFS スケジュールでジョブの処理を行う。

表 2 重み係数 α が稼働率に及ぼす影響
(上段: 最適スケジューリング, 下段: FCFS, $n_T=60$, $n_J=30$, $TUP=0.2$, $RT_{AGV}=1.0$)

WEIGHT α	No Buffer ($C=0$)	Large Buffer ($C=9$)
0.0	85.0 / 77.7	96.7 / 88.7
0.2	87.2 / 80.8	95.7 / 89.7
0.4	87.3 / 81.1	95.8 / 90.2
0.6	87.5 / 81.2	95.5 / 91.1
0.8	87.5 / 81.2	95.5 / 90.7
1.0	87.6 / 81.2	94.5 / 90.6
Random	84.0 / 77.2	93.7 / 88.5

図 5 重み係数 α の影響 ($n_T=60$, $n_J=30$, $TUP=0.2$)

(a) 各手法による平均稼働率

(b) 各手法による稼働率のばらつき

図 6 最適工具配分と最適スケジューリングの効果

なお、図中のすべての点は 30 例題の結果である。明らかに、PSA-CS 法は常に最も高い平均稼働率を示し、かつその標準偏差はどのような APC のバッファ容量に対しても小さい。その他、PSA 法 ($\alpha=0.4$) による工具配分の後、最適スケジューリングを行う OTA/OS 法、 $\alpha=0.0$ の PSA-CS 法とも比較を行った。本手法は OTA/OS 法による稼働率を 1~2% 程度改善しており、同時最適化の効果が確認された。また $\alpha=0.0$ の PSA-CS 法は、MLB/OS 法の性能をいくぶん改善するが、特にジョブ処理時間のばらつきの影響を吸収できないわずかなバッファ容量のもとでは平均稼働率、その標準偏差とも本手法に劣る。

PSA-CS 法の CPU 時間は、最良解 X_B の更新時に行われるスケジューリングの回数に大きく依存するためまちまちであるが、 $n_T=60$ の場合、2~30 分程度であった。スケジューリングを伴わない PSA 法の平均 CPU 時間は、 $n_T=60$ のとき 5.6 秒、 $n_T=100$ のとき 24.6 秒であった。なお、すべてのプログラムは FORTRAN 77 で書かれ、Sun SPARC station IPC 上で走らされた。

6. 結 論

同種のマシニングセンタ、AGV などから構成されるフローショップ型 2 機械自動生産システムにおける最適生産計画の一環として、最適工具配分問題を考察した。まず、ジョブ処理時間のばらつきと機械間の負荷のアンバランスに着目した工具配分問題を考え、これにスケジューリングを考慮した SA の手法を適用した。また、数値実験によりこのようなアプローチがシステム稼働率の向上に極めて有効であることを示した。このような FMS の最適運用計画を、ジョブショップ型や大規模 FMS へ拡張することを今後の検討課題としたい。

最後に、1 章で示した並列型、フローショップ型および混合型について比較検討する。並列型に比して、

フローショップ型は、(i) 工具マガジンの容量制約の下で MC 間の負荷バランスが取りやすい、(ii) 同一工具の重複が避けられる、などの利点がある一方、(iii) MC 間でのジョブの搬送が必要、(iv) スケジューリングが難しい、などの短所を有する。並列型では、各 MC で処理されるジョブのグループ化が行われるが⁽⁷⁾、一つの工具が両グループで要求され、その工具の重複を認めなければならない場合がある。また、多種少量化が進むにつれて使用工具数が膨大となり、各グループに必要な工具すべてを対応する MC の工具マガジンに同時保有することができなくなる。このような場合、工具交換を加工の途中で行うという工具交換問題に直面するか⁽⁸⁾、あるいは混合型に移行するかのどちらかである。後者の場合は、1 台の MC で加工完了するジョブの他方の MC での処理時間を零としたフローショップ型とみなせる。実際、3 章で示した工具グループ $G(t)$ は、グループ化されたジョブを念頭においたものである。したがって、 $G(t)$ を GT 手法などによって決定した後、グループ化されなかったジョブの工具に対しては本研究の手法が効果的であると期待できる。

文 献

- (1) 中村・木瀬, 日本 OR 学会 1990 年度秋期研究発表会アブストラクト集, (1990), 160.
- (2) Berrada, M and Stecke, K.E., *Management Sci.*, **32**-10 (1986), 1316.
- (3) 長沢・ほか 3 名, 機論, **55**-512, C(1989), 1129.
- (4) Aart, E. and Korst, J., *Simulated Annealing and Boltzmann Machines*, (1989), John Wiley & Sons.
- (5) Matsuo, H., ほか 2 名, *Annals of Oper. Res.*, **21** (1989), 85-108.
- (6) Karuno, Y., Master's Thesis of Department of Mechanical and System Engineering, *Kyoto Institute of Technology*, (1991), 27.
- (7) Kumar, K. R., Kusiak, A. and Vannelli, A., *European J. Oper. Res.*, **24** (1986), 387-397.
- (8) Tang, C. and Denardo, E., *Oper. Res.*, **36** (1988), 767-777.