GA システム結合型フレキシブルトランスファー生産ライン
シミュレータによるバッファ数決定*山 本 秀 彦^{*1}Buffer Capacity Decision by Simulator of Flexible Transfer Line
Including Genetic Algorithm System

Hidehiko YAMAMOTO

This paper describes a simulator of a flow-type production line that manufactures a variety of parts. The simulator includes the genetic algorithm system and decides the buffer capacity between each machine tool. In this paper, the algorithm for the production simulator including genetic algorithm system and the rules for moving parts are proposed. The paper also defines the model of a flexible-transfer-type line. The simulator is applied for a flexible automation line such as a production line for automotive parts. As the result of 20-40 generation simulations, a more approximate solution of the capacity of buffers was acquired.

Key Words: Production System, Production Planning, Flexible Manufacturing System, Production Simulation, Genetic Algorithm

1. 結 言

離散系シミュレーションは、1つのジョブから次のジョブに移る時、その行先が複数存在する Flexible Manufacturing System (FMS) などにおいて、稼働状況を把握する上で有効な支援となっている。一方、次のジョブの行先が1つである、フロー型の生産ラインでは、部品の流れが一定であるため、稼働状況の把握はシミュレーションを行わなくても比較的容易に予測できそうに思われた。

ところが、近年の製品品種の増加にあわせ、フロー型の生産ラインも多品種の部品を1つの生産ラインで加工、または組付する必要が生じてきた。フロー型生産ラインも、扱う部品が複数になると、稼働状況の把握は必ずしも容易とは言えなくなる。特に、自動車産業や家電産業の自動化生産ラインに見られるように、多品種の部品を同時に投入するフレキシブルトランスファー型生産ラインでは、各部品毎に各工作機械での加工サイクルタイムが異なるため、工作機械の前で投入されるのを待つ待機部品が発生しやすくなる。更に、品質チェックや故障などによる機械停止の要因が加わ

れば、投入待ち部品の数は増加する。フレキシブルトランスファー型生産ラインは、ある工程から次の工程へは行先が決まっているため、送り出す次の工程のバッファに、投入待ちの部品が全部詰まってしまうと、この工程の工作機械は、加工を完了したにも拘らず、ワークを機械外に排出できず、機械停止をしなければならなくなる。このことから、フレキシブルトランスファー型生産ラインでは、各工作機械の前に部品をためておくバッファの容量が、生産ラインの稼働率の良し悪しに直接おきかな影響を与えることになる。バッファ数を多くとれば、機械停止は防げるが、ストックコンベアが長大化し、レイアウト上の問題やコスト高につながる。

実際、生産ラインを設計する生産技術者は、この投入待ちの部品がいくつになるか分からないため、設計時には、過去の経験で全くの任意にストックコンベアの最大バッファ数を決定している。

FMSラインでの機械前のバッファ数の決定では、幾通りかのバッファ数の組み合わせを決めてシミュレーションを行い、良い結果のバッファ数を実際に採用している⁽¹⁾⁽²⁾。ところが、フレキシブルトランスファー型生産ラインでは、多くの工作機械がコンベアな

* 原稿受付 平成6年6月17日。

*1 正員, 和歌山大学 (〒640 和歌山市栄谷 930)。

どで結ばれ、そのラインサイクルタイムも秒単位と短いため、一台のコンベアが必要とするバッファ数が30～50個以上になることもまれではない。この様に、フレキシブルトランスファー型生産ラインのバッファ容量の決定は、その部品のフローが単純なため、容易に思われるが、1台のコンベアの持つバッファ容量が数十個以上となり、またこのコンベア数も多数となることより、すべてのバッファ数に対するシミュレーションの組み合わせが膨大となる。この意味で、フレキシブルトランスファー型生産ラインのシミュレーションによるバッファ容量決定も容易でない。

そこで、本論文では多品種部品を取り扱うフレキシブルトランスファー型生産ラインをモデル化し、このモデルを対象とした遺伝子アルゴリズム(GA)システムを結合する生産ラインシミュレータの開発について報告する。そして、このシミュレータを用い、自動的に各工作機械間のバッファ数の決定を行い、本シミュレータの有効性を検証する。

2. GAシステムと生産シミュレータの結合

多品種部品を1つの生産ラインで加工、組立てを行うフレキシブルトランスファー型生産ラインの運用が効率の良い生産ラインの1つとして望まれる。この種の生産ラインを設計する場合、各機械前の部品バッファ数をいくつに設定するかを決定しなければならない。

この様な、一種の生産設計のスケジューリング問題の解の探索には、入力するパラメータを熟練者の知識により決定を行うルールベース・スケジューリング法がある^{(3)～(5)}。さらに、このルールベースを自動的に得る手法として、生産シミュレータの実行結果をフ

ィードバック情報とし、ヒューリスティック・ルールを充実していくインテリジェントシミュレータが研究されている⁽⁶⁾。ところが、フレキシブルトランスファー型生産ラインのバッファ数を決める問題においては、生産ライン設計者自身、生産ライン状況から判断して、各機械間のバッファ容量がいくつが適当であるかを決定できる具体的知識を持っているわけではないのが現状であるため、この種のルールを作ることが難しい。

また、生産スケジューリング問題を、定式化による最適化問題とし、GAシステムを用いて解く自動化システム研究もある^{(7)～(8)}。しかし、多工程を持つ生産ラインを対象とする本バッファ数決定問題は、定式化することが容易でない。

本論文では、この定式化が難しいフレキシブルトランスファー型生産ラインのバッファ容量の決定を、自動的に行うための手法である、GAシステムによるフィードバック情報を利用する生産ラインシミュレータについて述べる。この様なGAシステムと生産ラインシミュレータとの結合利用の研究はいままでなされていない。そこで、生産ラインシミュレータにより得られた結果をフィードバック情報として、この情報をGAシステムで処理し、その結果を再び生産ラインシミュレータの入力情報として利用する、GAシステム結合フレキシブルトランスファー型生産ラインシミュレータを提案する。

図1にGAシステムとフレキシブルトランスファー型生産ラインシミュレータの結合概念図を示す。GAシステムはシミュレータシステムを駆動する入力パラメータの一部を決定する役割をはたす。シミュレータシステムの構成成分の一つとしてGAシステムを位置付けている。この結合されたGAシステムのフィード

図1 GA結合フレキシブルトランスファー型生産ラインシミュレータ

バック情報により、最終的に良い近似解となるバッファ数の組み合わせを自動的に発見することが、本シミュレータの目標である。

このGAシステム結合生産シミュレータの運用は、次に示す二種類に分けた入力パラメータを基本とするシミュレータ運用アルゴリズムに基づき行う。

STEP1: 設計する生産ラインの特徴となる複数の基本パラメータ $P = \{P_1, P_2, \dots\}$ に関する値を決定する。

STEP2: バッファ数に関する S 種のパラメータ $B = \{B^1, B^2, \dots, B^S\}$ の値をGAシステムにより生成する。

STEP3: 基本パラメータとバッファ数に関するパラメータの1つの要素とを組み合わせ、これを一つの要素とした S 個の集合 $T = \{(P, B^1), (P, B^2), \dots, (P, B^S)\}$ を生成する。この T の要素を総合パラメータとする。

STEP4: 総合パラメータを一回のシミュレーションの入力条件とし、それぞれ S 回のシミュレーションを実行する。

STEP5: S 回の実行結果に対し、終了条件の判定を行う。満足していればシミュレーションは終了する。満足していなければ、この実行結果をGAシステムに送りSTEP2に戻る。

この様に、二種のパラメータをあわせて利用する本シミュレータは、終了条件に満足するシミュレーション結果が得られるまで、何度もSTEP2からSTEP4までの遺伝子操作とシミュレーションを繰り返し行うことで良い解を探索する。

3. フレキシブルトランスファー型生産ラインのモデル化

本シミュレータの対象とするフレキシブルトランスファー型生産ラインを図2のようにモデル化する。このモデルは多品種の部品を1つの生産ラインに投入し、加工または組立てを行うことのできるフロー型生産ラインである。部品投入部、加工部、完成品保管部の3つの部分より構成する。

部品投入部は K 品種の部品に対し番号付けを行い品種を区別する。例えば、 $(W_1, W_2, \dots, W_K)$ = $\mathbb{W}$ という K 種の部品 W を表す。そして品種ごとの生産目標比率に比例し各種の方法⁽⁹⁾で品種ごとの投入順番を決定し、1個ずつ部品を加工部へ投入する。

加工部は N 台の工作機械 $\mathbb{M} = \{M_1, M_2, \dots, M_N\}$ より構成する多段階工程の生産ラインとする。各工作機械間はコンベア類の自動搬送システムでつながる。また、各工作機械の前には部品投入待ちのバッファ装置 N 台、 $BF_1, BF_2, \dots, BF_N$ が付帯している。この様に、加工部は1台のバッファ装置と1台の工作

図2 フレキシブルトランスファー型生産ラインのモデル

機械を1組のステーションとした N ステーションの集合 $\mathbb{T} = \{(BF_j, M_j) \mid j=1, 2, \dots, N\}$ として構成する。

完成品保管部は加工部より加工を完了した完成品の排出先であり、品種ごとに保管する。

次に、この様に構成したフレキシブルトランスファー型生産ラインの稼働制御について述べる。本研究では、自動車部品産業界の生産ラインによくみられる、多品種混合部品を1つの生産ラインで自動的に搬送、加工を行い、また品種による段取り時間をゼロとする稼働状況を作り出す制約条件とそのパラメータを以下に定める。

【制約条件】

- ① 部品の加工時間は工作機械ごとに異なる。また、部品の品種によっても加工時間は異なる。
[パラメータ] $F_{\text{FINISH}}\{M_j(W_k)\}$ = 品種 W_k の工作機械 M_j での加工に必要な時間, $j=1, 2, \dots, N, k=1, 2, \dots, K$
- ② 各工作機械は、部品を任意個数加工するごとに品質チェックのため任意時間機械停止する。
[パラメータ] Q_{STOP} = 品質チェックの機械停止時間, Q_{PIECE} = 品質チェックの発生個数
- ③ 各工作機械は、部品を任意個数加工するごとに工具交換のため任意時間機械停止する。
[パラメータ] T_{STOP} = 工具交換の機械停止時間, T_{PIECE} = 工具交換の発生個数
- ④ ある特定の工作機械は、加工不良を多発する。
[パラメータ] R_{NOGOOD} = 加工不良品の発生確率
- ⑤ 各工作機械は軽微な故障で短時間の機械停止、

いわゆるチョコ停を頻繁に繰り返す。

[パラメータ] R_{STOP} = チョコ停の発生頻度,

T_{RSTOP} = チョコ停の機械停止時間

上記の制約条件の下で, モデル化された生産ライン内の部品の移動・停止を制御しなければならない。この制御を, 以下の部品移動ルールに従い行う。

【部品移動ルール】

[ルール1]

IF : 対象とする部品 W_k が工作機械 M_j 内へ投入

THEN : [ルール2] を実行する

[ルール2]

IF : (M_j の処理個数) = Q_{PIECE}

THEN : W_k は時間 Q_{STOP} だけ M_j 内にとどめ, [ルール3] を実行

ELSE : [ルール3] を実行

[ルール3]

IF : (M_j の処理個数) = T_{PIECE}

THEN : W_k は時間 T_{STOP} だけ M_j 内にとどめ, [ルール4] を実行

ELSE : [ルール4] を実行

[ルール4]

IF : M_j = チョコ停をおこす機械, かつ, M_j の稼働時間が R_{STOP} に一致する

THEN : W_k は時間 T_{RSTOP} だけ M_j 内にとどめ, [ルール5] を実行

ELSE : [ルール5] を実行

[ルール5]

IF : M_j = 加工不良をおこす機械, かつ, M_j の加工個数が R_{NOGOOD} に一致する

THEN : M_j をライン外へ搬出する

ELSE : W_k を $F_{INISH} \{M_j(W_k)\}$ の時間 M_j にとどめ, [ルール6] を実行

[ルール6]

IF : BF_{j+1} に 1 以上の空スペースが存在する

THEN : W_k を BF_{j+1} に移動させる

ELSE : BF_{j+1} に 1 以上の空スペースが存在するまで W_k を M_j 内にとどめる

なお, 部品は各バッファ装置内では, FIFO方式のルールに従う。

上記ルールは, ST_j を前工程, ST_{j+1} を後工程と考えた場合, 次に示す基本ルールに集約できる。

if (ST_{j+1} に空スペースが存在する) かつ

(ST_j 内の部品が搬出可能)

then (ただちに ST_j 内の部品を ST_{j+1} に送り出す)

即ち, ST_{j+1} に空スペースが発生した時, 可能な限り前工程が部品を供給するルールである。言い換えれば, このルールは, 後工程が必要な時前工程から部品が可能な限り供給される内容であり, 自動車生産現場でよく用いられるJIT方式の原則である。後工程引き

取り方式⁽¹⁰⁾にも準拠している。

4. シミュレーションのパラメータ設定

GAシステム結合の生産ラインシミュレータを駆動するには, 2章のSTEP1, STEP2で述べたパラメータを決定しなければいけない。このパラメータは2種類, 即ち基本パラメータ P とバッファ数に関するパラメータ B である。

基本パラメータ P は, 何度も繰り返しシミュレーションが行われる中で, 初期に値を決定すれば, その値に変更を加えないものである。その主なものは, 前章の制約条件に関するもので, 具体的には次の様に設定する。

$P_1 = F_{INISH} \{M_j(W_k)\}$, $P_2 = Q_{STOP}$, $P_3 = Q_{PIECE}$, $P_4 = T_{STOP}$, $P_5 = T_{PIECE}$, $P_6 = R_{NOGOOD}$, $P_7 = R_{STOP}$, $P_8 = T_{RSTOP}$, $\dots$

バッファ数に関するパラメータ B は, シミュレーションを実行することに値を変更するパラメータである。このパラメータ B の任意の一要素 B^i は図3に示すように, N 台の工作機械の, 各工作機械の前にためることのできる部品の最大数量 L_N を使って式(1)の様に構成する。

$$B^i = \{L_1, L_2, \dots, L_N\}, i=1, 2, \dots, S \quad \dots (1)$$

ここで基本パラメータ P の具体的な値は, これから設計しようとする生産ラインの仕様にあわせてユーザー自身が決定できる。一方, バッファ数に関するパラメータ B は, GAシステムにより自動的に決定する。

5. バッファ数パラメータの進化プロセス

5.1 初期個体群の生成 本章は, バッファ数に関するパラメータ B のGAシステムによる自動決定方法について述べる。初めに, GAシステムの初期個体群の数として, バッファ数に関する式(1)のパラメータ B^i の種類数 S を採用する。続いて, 一つの初期個体の各要素を, 乱数により次の様に割り当てる。

バッファ数の許容限度数 L_{MAX} を設定し, 1 から L_{MAX} の間の整数値をランダムに取る値 $R(L_{MAX})$ をステーションの数 N 回生成し, 式(1)の $L_1, \dots$,

図3 バッファ数のパラメータ

L_N に割付ける。この様に割付けられた集合が式 (3) の様に一つの個体 $\mathbb{B}_i$ を形成する。

$$L_j = R(L_{\max}) \quad \dots (2)$$

$$\mathbb{B}_i = \{L_j \mid j=1, 2, \dots, N\} \quad \dots (3)$$

$$i=1, 2, \dots, S$$

式 (2), (3) で j はステーション数を, $\mathbb{B}$ の右下添字は世代数ゼロを示す。同様にして, この様な 1 個の個体の遺伝子を種類の数 S 個生成し, 式 (4) に示す初期個体群 $\mathbb{B}_0$ を生成する。以上の過程は, 2 章のアルゴリズム STEP2 に該当する。

$$\mathbb{B}_0 = \{\mathbb{B}_i \mid i=1, 2, \dots, S\} \quad \dots (4)$$

5・2 シミュレーションと適応度 1 回のシミュレーションは, 2 章のアルゴリズム STEP3 で述べた様に, 基本的パラメータ $\mathbb{P} = \{P_1, P_2, \dots\}$ と, バッファに関する 1 個体のパラメータ $\mathbb{B}_i$, ($i=1, 2, \dots, S$; G は世代数) の組合せからなる総合パラメータ $\mathbb{T}^i = \{\mathbb{P}, \mathbb{B}_i\}$ を用いて行う。

さて, この 1 回のシミュレーションでは, K 個の部品ごとの完成品数が測定できるので, この測定値を使い次式により, 生産ラインの稼働率を計算する。

$$e = \frac{\sum_{K=1}^K F_K}{(\text{シミュレーション時間} / \text{ラインサイクルタイム})} \quad \dots (5)$$

F_K : 品種名 K の完成品数量

式 (5) の稼働率の値を, このシミュレーションで用いた 1 個体の適応度とする。そして, シミュレーションを S 回実行し, 一世代の適応度の集合として式 (7) の構成を得る。

ここで, 適応度が同値な複数の異なる個体が出現した場合, 全バッファ容量の少ない個体の適応度が大きくなる様に, この各個体の適応度を次の様に修正する。即ち, 全バッファ容量を式 (6) で計算し, この全バッファ容量の一番少ない個体の適応度から任意定数 Pe 値を差し引いた値を, 全バッファ容量が二番目に少ない個体の適応度とする。全バッファ容量が三番目に少ない個体の適応度は, この二番目にバッファ容量の少ない適応度から, やはり任意定数 Pe 値を差し引いた値とする。四番目以降の適応度についても, 同様の修正を行う。この修正により, 同じ稼働率でもバッファ容量の少ない個体が解として選ばれやすい探索手法となる。

$$\text{全バッファ容量} = \sum_{j=1}^N L_j \quad \dots (6)$$

$$\mathbb{G} = \{e_i \mid i=1, 2, \dots, S\} \quad \dots (7)$$

式 (7) の右上添字 i は個体の種類を, 右下添字 G は世代数を示す。

5・3 遺伝的操作 S 回連続シミュレーションの後, シミュレーション終了条件の判定を行う。終了条件を満たさない時は, 適応度 $\mathbb{G}$ を使い, 選択・交叉・突然変異の遺伝的操作を行う。

遺伝的操作には, 確率的選択を使用する。シミュレーションは, 1 個体について 1 回のシミュレーションを行うのであるが, この 1 回のシミュレーション時間に数分以上必要となる。このため, 個体数 S はあまり大きな値をとることはシミュレーション時間の関係から許されない。

そこで, 個体数 S 個が十分大きくとれない本問題の場合では, 期待値戦略⁽¹¹⁾を用いた確率的選択を採用する。即ち, 適応度集合 $\mathbb{G}$ を式 (8) により期待値 SP_i に変換する。

$$SP_i = S \times \frac{(e_i)^2}{\sum_{i=1}^S (e_i)^2} \quad \dots (8)$$

式 (8) により, 期待値集合 $\mathbb{S}\mathbb{P}_G$ は式 (9) の様に構成される。

$$\mathbb{S}\mathbb{P}_G = \{SP_i \mid i=1, 2, \dots, S\} \quad \dots (9)$$

この期待値集合を用い, ルーレット方式により, 1 組の遺伝子組合せを選択する。この選択により選択・交叉・突然変異の遺伝子操作を繰り返す⁽¹²⁾。この時, 選択された個体の期待値は, 0.5 を差し引かれた値に変化を受け, この新しい期待値として次の選択に関わる。交叉は一点交叉を用いる。突然変異は, 変異を起こす遺伝子座とその値をランダムに選択する。即ち, 式 (3) の j と L の値をランダムに選ぶ。

6. 適用事例

GA システム結合生産シミュレータを開発し, 自動車部品生産ラインでよくみられるフレキシブル自動化ラインに適用した。本システムのシミュレータ部分は S I M A N 言語, GA システム部は C 言語でプログラム化した。対象とする生産ラインは, 4 種の加工対象部品を, 生産量の比率に合わせて均等に, 1 個ずつ生産ラインに投入し, 自動コンベアにつながれた工作機

図4 適応度の推移 (変異率20%)

機8台が自動的に部品をローディングし、加工し、またアンローディングする完全自動化生産ラインである。4種の加工対象部品の生産量の比率はあらかじめ設定し、本例では(W_1 の生産量):(W_2 の生産量):(W_3 の生産量):(W_4 の生産量)=6:4:1:2とした。本事例の基本パラメータの主な各値を次に示す。 $P_1=(14秒, 15秒, 20秒, 10秒)$, $P_2=30秒$, $P_3=300個$, $P_4=20秒/回$, $P_5=250個$, $P_6=(120秒に1回)$, $P_7=(120秒に1回)$, $P_8=20秒$ 。また、STEP4の1回のシミュレーションの終了条件はシミュレーション時間=160時間、STEP5の終了条件はすべての個体が同じ適応度値となるか、世代数=100となる時とした。さらに、 P_e 値は0.01を採用した。

稼働日数20日分のシミュレーションの結果突然変異の確率を20%とした時の各世代ごとの最高適応度と平均適応度を表した図が図4である。平均適応度は、一世代の個体数20個の適応度の平均であり、 $(\sum_{i=1}^{20} e^i) / 20$ により求める。20世代末において、最高適応度が約0.78(稼働率78%に相当)の極限值に達しており、この時のバッファ数の組み合わせは B_1 から順に、17個、22個、23個、16個、20個、26個、19個、2個となった。平均適応度は30世代あたりまでは急速に上昇していった。突然変異確率を35%に上げ、ランダム探索に少し近づいた場合での適応度収束状態も同時に調べた。変異率の高い場合は、やはり多様性のある個体が多く現れ、その結果平均適応度の変動が大きい、やはり20世代前に最高適応度が極限值に達する結果となった。

この様に、設計時では熟練設計者でも全く検討もつかない本事例の各機械間の必要バッファ数が、GAシステムを結合したフレキシブルトランスファー型生産ラインシミュレータを用いることで、数十分で推定できる。

7. 結 言

多品種部品を一つのフロー型生産ラインで加工したり組み付けする生産方式を開発する際に、各工程間のバッファ数をいくつに設定するかを決めることは容易でなかった。本研究では、GAシステムを結合したフレキシブルトランスファー型生産ラインシミュレータを用いて、このバッファ容量の問題に取り組んだ。

各工程間のバッファ数を遺伝子表現とし、これを生産シミュレータの入力パラメータの一部として取り込むことで、自動的にバッファ数の決定が可能となった。実際にシミュレーション実験を行い、有限回の遺伝子操作の結果、熟練設計者が事前に決定できないバッファ数の組み合わせが探索できた。

文 献

- (1) T. Schriber, Simulation Using GPSS, John Wiley, 1974.
- (2) H. Fujimoto, Simulator for Design and Operation of Flexible Manufacturing Systems, Proc. of ASME USA-JAPAN Symposium on Flexible Automation, Vol.1, pp.577, 1988.
- (3) 関根 他, 専門家モデルに基づく生産スケジューリング・システム, 人工知能学会誌, Vol.5, No.2, pp.194, 1990.
- (4) 大村・高橋・小西, シミュレーションによる定量評価を内包した物流制御エキスパートシステム, システム制御情報学会論文誌, Vol.6, No.3, pp.119, 1993.
- (5) 鳩野, 山縣, 田村, ルールベースを用いたフレキシブル生産システムのオンライン・スケジューリング, システム制御情報学会論文誌, Vol.4, No.9, pp.375, 1991.
- (6) Park, S. C., Raman, N. and Shaw, M. J., Heuristic Learning for Pattern Directed Scheduling in a Flexible Manufacturing System, Proc. 3rd ORSA / TIMS Conference Flexible Manufacturing Systems: Operations Research Models and Applications, pp.369, 1989.
- (7) 西川・玉置, ジョブショップ・スケジューリング問題に対する遺伝アルゴリズムの構成法, 計測自動制御学会論文誌, Vol.27, No.5, pp.592, 1991.
- (8) 飯間・三宮, 遺伝アルゴリズムを用いた変形フローショップ問題の解法, システム制御情報学会論文誌, Vol.6, No.10, pp.437, 1993.
- (9) J.R.Jackson, A Computing Procedure for a Line Balancing Problem, Management Science, Vol.2, No.3, pp.261, 1956.
- (10) 大野, トヨタの生産方式, ダイアモンド社, 1978.
- (11) 北野 他, 遺伝的アルゴリズム, 産業図書, 1993.
- (12) D.E.Goldberg, Genetic Algorithm in Search, Optimization and Machine Learning, Addison-Wesley, 1989.