

再構築可能なスケジューリング問題への 遺伝的アルゴリズムの適用

田 代 順 一 (大学院工学研究科情報工学専攻)*
小 野 俊 彦 (情報工学科)

Job-shop Scheduling Problem Enabling Reconstruction by Genetic Algorithms

Jyun-ichi TASHIRO (Communication and Computer Engineering, Graduate School of Engineering)
Toshihiko ONO (Department of Computer Science and Engineering)

Abstract

The purpose of this study is to develop the method using genetic algorithms (GAs) to be able to search the optimal schedule by which the finish time of each product becomes earliest to satisfy customer's requirements. This system produces a schedule which reflects the transportation time of products and the exchange time of tools. When a machine breaks down or the delivery time is changed in the midst of production, the method can set up a new schedule by referring to the current conditions. By incorporating schedule compiling algorithms (SCA) into the system, the search space for the solution decreases and the required calculation time is reduced. The results of simulation studies including re-building of schedules are explained.

Key words: *scheduling problem, genetic algorithms, GAs*

1 はじめに

本論文では、複数の機械を用いていくつかの製品を製作する際に各機械上での作業の処理順序を決定するスケジューリング問題に対して、遺伝的アルゴリズム (Genetic Algorithms: GA) を適用する新しい手法について述べている。

GA は、生物進化 (選択淘汰, 突然変異, 交叉) の原理に着想を得たアルゴリズムであり、米国ミシガン大

学の John H. Holland によって1960年代に提案された確率的探索の一手法である。このアルゴリズムは演算子が比較的簡潔であるため、モデル化が困難な問題にも適用できるという利点がある。そのため工学分野から経済分野までさまざまな問題で適用され、成果を上げている。

スケジューリング問題は機械の特性や仕事に課せられる条件によって種々の型に分類されるが、本研究ではジョブショップ型のスケジューリング問題を対象にした。このスケジューリング問題では、どの製品も完成までに複数回の作業を必要とし、必ず第1作業から順に処理しなければならないという先行制約条件を持っている。この問題は今日まで多くの研究がなされてきた

* 現在(株)エクシース
平成11年5月10日受付

が、NP 困難な組合せ最適化問題であるため、例えば分岐限定法のような数理計画手法を用いるのは計算量の点で必ずしも最良の方法ではなく、十分に満足できる準最適解を効率的に求めることができるこの GA の適用は有効であるといえる。

本研究ではより実用的なシステムを構築するため、スケジューリング問題に対して以下の点を考慮した。

- 製品の移動時間を考慮する。
- 治具の交換時間を考慮する。
- 製造途上で機械の故障や納期の変更といった不慮の事態に対応させる。

まず時間に関しては、作業の処理に必要な時間の他に、ある機械から別の機械への製品の移動時間や、加工内容によっては必要となる治具の交換時間を考慮した。

また機械の故障や納期の変更 (特に短縮) など不慮の事態が発生した場合、あらかじめ作成していたスケジュールに沿って処理を続けては納期に遅れてしまうことが考えられる。このような場合に対処するため、本方式は故障状況や変更になった納期に基づいてスケジュールを再構築することを可能とした。

以下では提案方式について説明するとともに、2 種類のシミュレーションの結果についても報告する。

2 システム構成

本システムは図 1 に示すようにスケジュール作成アルゴリズム (Schedule Compiling Algorithms: SCA) および遺伝的アルゴリズムの 2 つのアルゴリズムと各種データファイルで構成している。本システムの主な動作は以下のとおりである。

まず GA によって生成された初期遺伝子は、交叉、

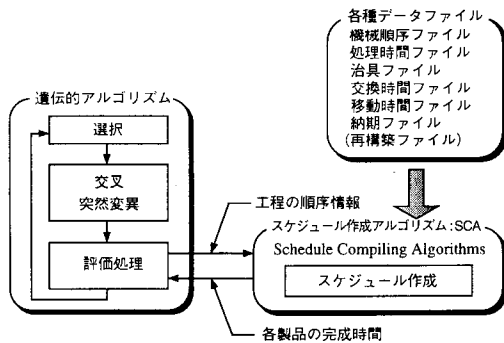


図 1：システムの流れ

突然変異などの遺伝的操作を受けた後に SCA へと渡される。遺伝子を受け取った SCA は、そこに記された情報と制約条件を規定した各種データファイルを参照しながら一つのスケジュールを作成し、各製品の完成時間を GA の評価処理部へと返す。GA ではこれを基に遺伝子を評価する。

評価の高い優秀な遺伝子ほど次の世代に生き残りやすく、世代をくり返すことで本システムは最適なスケジュールを探索していくことになる。

制約条件

スケジューリング問題の制約条件は多数存在するが、本研究で取り扱った制約条件にはつぎのようになっている。

- 各製品は第 1 作業から順に処理しなければならない。(先行制約条件)
- 各作業は所定の機械で処理しなければならない。
- 各作業の処理に必要な時間は予め決まっている。
- 製品を移動する際はスケジュールに所定の時間を確保しなければならない。ちなみに移動時間は移動元と移動先の機械によって異なる。
- 機械に治具・金型等の特殊な部品 (以下治具と呼ぶ) を取り付ける場合はスケジュールに所定の時間を確保しなければならない。交換時間は機械に取り付けてあった治具とつぎに使用する治具の種類によって異なる。
- 各製品の完成時間が製品それぞれの納期に間に合わなくてはならない。

以上の制約条件を規定するデータファイルとそこで規定する制約条件との対応は表 1 のとおりである。

表 1：データファイルと制約条件

| ファイル名 | ファイルに記述された情報 |
|----------|------------------------|
| 機械順序ファイル | 各作業で処理を行う機械の番号 |
| 処理時間ファイル | 各作業で処理に必要な時間 |
| 治具ファイル | 各作業で使用する治具の番号 |
| 交換時間ファイル | ある治具から別の治具へ交換するのに必要な時間 |
| 移動時間ファイル | 各機械間の移動時間 |
| 納期ファイル | 各製品の納期 |
| 再構築ファイル | 再構築に関する情報 |

色のファイルは再構築を行うときのみ必要となる

3 GAの適用

ある問題にGAを適用する際、どのような遺伝子表現を採用するが計画上の重要な要素となる。それは遺伝子の表現方法によってGAの探索能力が大きく左右されるためである。従って、ここではジョブショップスケジューリング問題へのGAの適用方法を遺伝子表現を中心に述べていく。

遺伝子の構成

本方式は解の探索空間を縮小させるために、遺伝子には作業をガントチャートへ描画する製品の順序のみを記述した。この遺伝子情報に従ってSCAは各作業を可能な限り開始時間が早くなるようにガントチャートへ描き出していく。

今回採用した遺伝子の構造を図2に示す。遺伝子に

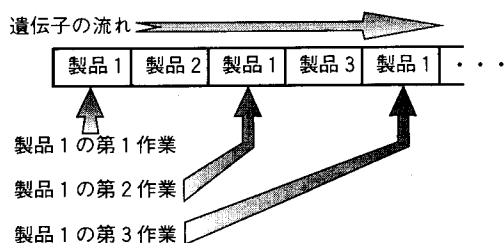


図2：遺伝子構造

は特に作業番号は記述していない。これは今回のスケジューリング問題では“どの製品においても必ず第1作業から順番に処理していかなければならない”という先行制約条件があるためである。まず遺伝子の最初に現れた製品番号はその製品の第1作業を表し、n番目に現れた製品番号はその製品の第n作業を表すようにした。SCAは遺伝子に記述された順序でガントチャートに書き出していくので、この遺伝子構造を採用することにより必ず第1作業から順に描き出していくことができる。従って遺伝子表現はパス表現となり、この問題をパス表現の順序問題として扱うことができる。

交叉操作

順序問題用の交叉には循環交叉 (CX法: Cyclic crossover) や、順序交叉 (Order crossover: OX)、部分写像交叉 (Partially-mapped crossover: PMX) 等[5]があるが、本研究ではOliverにより提案された循環交叉を用いることにした。

突然変異操作

順序問題用の突然変異には、ランダムに選出した2点の遺伝子座を交換する“SWAP法”を適用した。

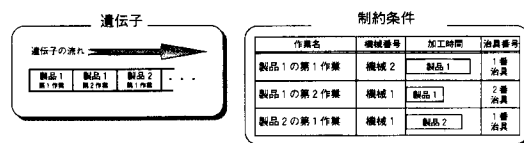
4 スケジュール作成アルゴリズム (SCA)

SCAは遺伝子が持っている作業の順序情報を参照しながら、各製品の完成時間になるべく早くなるように各作業をガントチャートに描き出していくアルゴリズムである。SCAはまず遺伝子を受け取ると、遺伝子に記された情報と制約条件を構成する各種データファイルを参照しつつ、遺伝子に記述された順序で全ての作業をガントチャートに描き出していく。こうして完成したスケジュールが実行不可能なスケジュールにならないよう、SCAに以下の機能を持たせている。

1. どの製品についても第1作業から順に描き出していく。
 2. 描き出す作業に関する条件 (処理をする機械、処理に必要な時間、製品の移動時間、治具の交換時期) を各種データファイルから取り出し、これを満たすようにガントチャートに描き出す。
 3. 各作業の開始時間がその製品の前作業の終了時間後で、かつ、なるべく早くなるように配置する。
- これらの配慮によりSCAで作成したスケジュールは、納期に関する条件以外の全ての制約条件 (先行制約のほか移動時間や交換時間等) を必ず満たすことになる。

次にSCAがガントチャートを作成する過程を簡単な例を用いて示す。例えば図3に示す遺伝子と制約条件がSCAに与えられた場合、各作業を遺伝子に記された順序で図4のようにガントチャートに描き出していく。作業を描き出す際はその作業に関する情報を各種データファイルより取得し、処理を行う機械番号や処理に必要な時間を考慮する。

図4 a) では一番目の遺伝子座に記された情報と各種データファイルに記述された情報に沿って、製品1の第1作業をガントチャートに描き出している。次に



a) 遺伝子の一例

b) 制約条件の一例

図3：遺伝子とデータファイル

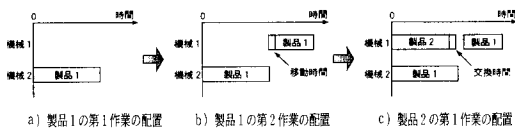


図 4 : SCA の動作

図 4 b) では先ほどと同じように製品 1 の第 2 作業を描き出しているが、その際に SCA は製品 1 を機械 2 から機械 1 へ運ぶために必要な移動時間を考慮している。図 4 c) では製品 2 の第 1 作業をガントチャート上の隙間に描き出している。

この作成過程を基に、SCA の動作の詳細を以下に説明する。

隙間への配置

図 4 C) において SCA は「製品 2 の第 1 作業」を「製品 1 の第 2 作業」の手前に描き出している。これは「製品 1 の第 2 作業」を描き出した時点でガントチャート上に隙間が生じており、SCA にはこの隙間を活用して作業を描き出すことができるためである。

このような配置を実現するため SCA は全ての隙間の場所と長さ、及び隙間の前後の作業で使用した治具の種類を記録している。ただし隙間に配置する際は、図 5 のよう治具の交換時間を加えた上で隙間に描き出せるかどうかを判断する。また作業を隙間に配置した場合は、それと同時に配置された隙間の情報の更新を行い、別の作業が同じ場所へ描き出されることを防止している。

隙間の分断

図 6 のように一つの隙間を二つに分断するような作業の配置を行なった時、SCA は両方の隙間に対して情報の記録を行う。これにより分断された隙間がさらに分断されるような配置が生じてても、SCA はすべての隙間を記録できる。

移動時間と交換時間の重複

図 7 に製品の移動時間と治具の交換時間がオーバーラップしたガントチャートを示す。例えばこの図において、製品 1 を機械 1 から機械 2 へ運搬すると同時に機械 2 において治具を取り付けることは可能であるため、本方式では移動時間と交換時間が異なっても良いとしている。この時、製品 1 の第 2 作業は製品の移動と治具の交換の両方が終了した時間を開始時間として描き出す。

SCA のフローチャート

SCA の動作をステップに分けて説明していく。

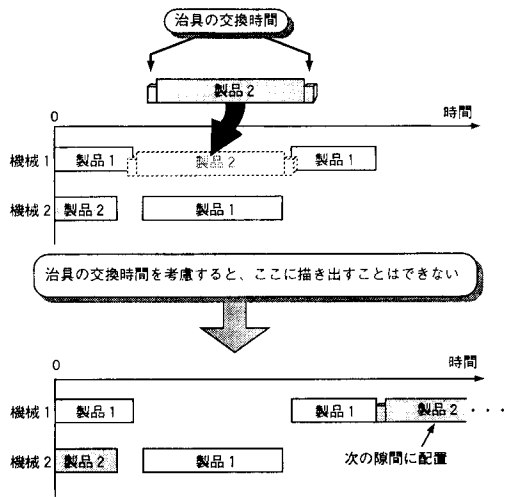


図 5 : 隙間への配置チェック

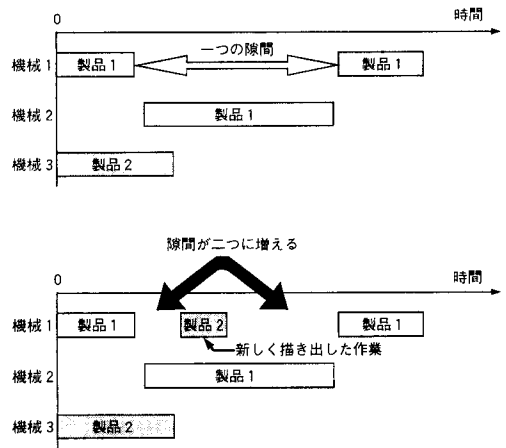


図 6 : 隙間の分断

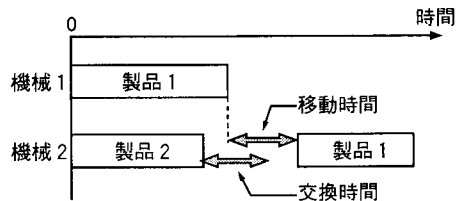


図 7 : 移動時間と交換時間の重複

1. 参照する遺伝子座を最初の遺伝子座とする。
2. 遺伝子座を参照して作業名を取り出す。

3. ステップ2で取り出した作業に関する様々な情報を各種データファイルから取得する。
4. ステップ2, ステップ3で得られた情報を基に交換時間, 移動時間を考慮した上で可能な限り開始時間が早くなるようにガントチャートへ描き出す。
5. 作業を描き出した結果ガントチャート上に新しく隙間が発生した場合は, その隙間に関する情報を記録する。隙間に描写することで隙間の情報に変化した場合はその情報を修正する。
6. すべての遺伝子情報を読み出していれば終了。そうでなければ参照する遺伝子座を次に進めてステップ2へ戻る

SCAのフローチャートは図8のようになる。

5 目的関数

SCAがスケジュールを作成すると, それに伴って各製品の完成時間が決定するので, GAの評価処理部はこの完成時間をSCAより受け取り, 以下に示す目的関数の式により遺伝子の評価を行う。

$$Obj = \sum_i W_i X_i \quad (1)$$

ここで*i*は製品番号, X_i は製品*i*が納期から遅れた時間, W_i は製品*i*の優先度をそれぞれ示す。前述のようにSCAで作成したスケジュールは, 納期に関する条件以外のすべての制約条件を必ず満たすため, 目的関数では納期に間に合うかどうかのみを判断すればよい。特に重要な製品には高い優先度を与え, 納期を大きくスケジュールに反映させる。なお, 目的関数 *Obj* が小さいほど評価が良くなる。

6 スケジュールの再構築

前述の方式により作成した最適スケジュールに基づいて作業を実行中, 以下に示すような不具合が生じた場合, 製品の完成が納期より大きく遅れてしまう可能性がある。

- ある作業が予定の時間に終了しなかった。
- ある機械が故障したため, 作業が一定の時間停止した。
- ある製品の納期が変更になり, このまま続行したのでは納期に遅れてしまう。

以上の各場面では, 前もって用意した最適スケジュールに沿って処理を続けることが最善とは限らない。新たなスケジュールを再び作成し, そのスケジュールに沿って処理を再開した方が納期の遅れをより少なくできることがある。本システムではこれらの事態が発生した場合, 変更以前のスケジュールデータや変更項目のデータを参照しながら, 新たにスケジュールを再構築する機能を持たせている。

再構築時の遺伝子表現

再構築時は前述の遺伝子構造とは異なる新しい遺伝子構造を使用する必要がある。なぜなら再構築時にはすでに終了している作業が存在し, これまでの遺伝子構造では過去のスケジュールをも改変してしまうためである。再構築時に使用する遺伝子構造の一例を図9に示す。

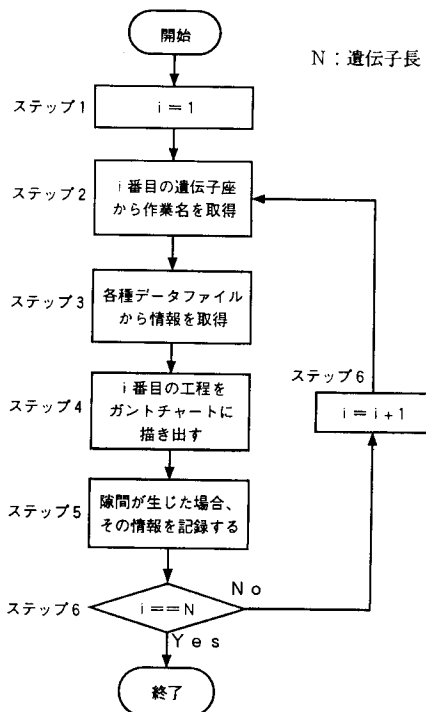


図8: SCAのフローチャート

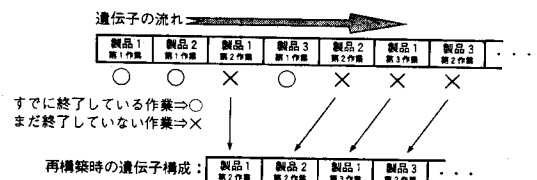


図9: 再構築時の遺伝子構造

まず、再構築を行う時点でのすべての作業を“すでに終了している作業”と“まだ終了していない作業”に分類する。再構築時に処理中の作業については、順調に処理が行われているものとして“すでに終了している作業”として扱う。従ってまだ終了していない作業のみを集めた遺伝子を再構築時の遺伝子として使用する。

前に述べたように遺伝子には製品番号のみを記入しており、作業番号は省略している。再構築時の遺伝子も同じく製品番号のみを記入しているが、すべての製品が第1作業から始まるわけではないので、各製品ごとのどの作業まで終了しているかという情報を取得するようにしている。例えば図9の例では製品1, 2, 3ともに第1作業は終了しているので、再構築時の遺伝子において最初に現れた“製品1”は“製品1の第2作業”を表すことになる。

この遺伝子構造を使用することにより、すでに終了しているスケジュールを除外してスケジュールを作成でき、再構築時もパス表現用の遺伝子操作(交叉・突然変異)を使用できる。

7 シミュレーション結果と検討

提案方式の特性を検証するために各種シミュレーションを行った。シミュレーションの主な目的は、以下のようにまとめられる。

- 収束性の検証
- 再構築の性能の検証

以上の検証を順に説明していく。なお、シミュレーション結果の表示には別に作成したガントチャート表示プログラムを使用した。表示するガントチャートは縦軸に機械番号、横軸に時間を取ったものであり、棒グラフの中に記された数字は製品番号を表している。

シミュレーションデータ

シミュレーションは製品数10, 機械数10, 各製品の

表2: 納期設定

| | | | | | |
|-------|------|-----|-----|------|-----|
| | 製品1 | 製品2 | 製品3 | 製品4 | 製品5 |
| 製品の納期 | 102分 | 94分 | 99分 | 117分 | 94分 |

| | | | | | |
|-------|-----|------|-----|-----|------|
| | 製品6 | 製品7 | 製品8 | 製品9 | 製品10 |
| 製品の納期 | 91分 | 100分 | 99分 | 85分 | 105分 |

表3: シミュレーションに使用するデータ

| 製品名 | 第1作業 | 第2作業 | 第3作業 | 第4作業 | 第5作業 | 第6作業 | 第7作業 | 第8作業 | 第9作業 | 第10作業 |
|------|---------------------|---------------------|---------------------|-------------------|---------------------|-------------------|---------------------|--------------------|-------------------|---------------------|
| 製品1 | 機械1 治具1番 1分2分 | 機械2 治具2番 1分0分 | 機械3 治具3番 1分3分 | 機械4 治具1番 1分 | 機械5 治具2番 9分 | 機械6 治具3番 5分 | 機械7 治具1番 7分 | 機械8 治具2番 3分 | 機械9 治具3番 7分 | 機械10 治具1番 1分 |
| 製品2 | 機械3 治具2番 5分 | 機械5 治具3番 1分 | 機械6 治具1番 9分 | 機械7 治具2番 1分 | 機械8 治具3番 6分 | 機械9 治具1番 7分 | 機械10 治具2番 1分 | 機械1 治具3番 4分 | 機械2 治具1番 4分 | 機械3 治具2番 4分 |
| 製品3 | 機械7 治具3番 6分 | 機械2 治具1番 4分 | 機械4 治具2番 8分 | 機械5 治具3番 3分 | 機械6 治具1番 7分 | 機械8 治具2番 9分 | 機械9 治具3番 8分 | 機械10 治具1番 5分 | 機械1 治具2番 4分 | 機械2 治具3番 7分 |
| 製品4 | 機械2 治具1番 1分0分 | 機械7 治具2番 5分 | 機械5 治具3番 1分2分 | 機械6 治具1番 6分 | 機械8 治具2番 1分3分 | 機械9 治具3番 7分 | 機械10 治具1番 4分 | 機械1 治具2番 8分 | 機械2 治具3番 2分 | 機械3 治具1番 1分2分 |
| 製品5 | 機械5 治具2番 8分 | 機械10 治具3番 6分 | 機械1 治具1番 4分 | 機械4 治具2番 9分 | 機械2 治具3番 3分 | 機械3 治具1番 4分 | 機械8 治具2番 7分 | 機械9 治具3番 8分 | 機械7 治具1番 6分 | 機械6 治具2番 1分0分 |
| 製品6 | 機械3 治具3番 6分 | 機械5 治具1番 2分 | 機械2 治具2番 4分 | 機械8 治具3番 8分 | 機械4 治具1番 1分0分 | 機械9 治具2番 5分 | 機械10 治具3番 8分 | 機械1 治具1番 5分 | 機械2 治具2番 4分 | 機械3 治具3番 5分 |
| 製品7 | 機械4 治具1番 6分 | 機械1 治具2番 5分 | 機械7 治具3番 1分0分 | 機械5 治具1番 9分 | 機械6 治具2番 2分 | 機械8 治具3番 4分 | 機械9 治具1番 4分 | 機械10 治具2番 4分 | 機械3 治具3番 3分 | 機械2 治具1番 2分 |
| 製品8 | 機械10 治具2番 9分 | 機械9 治具3番 4分 | 機械6 治具1番 1分0分 | 機械2 治具2番 7分 | 機械3 治具3番 5分 | 機械4 治具1番 5分 | 機械7 治具2番 6分 | 機械8 治具3番 3分 | 機械1 治具1番 3分 | 機械5 治具2番 4分 |
| 製品9 | 機械6 治具3番 1分2分 | 機械3 治具1番 2分 | 機械5 治具2番 5分 | 機械8 治具3番 8分 | 機械4 治具1番 4分 | 機械9 治具2番 9分 | 機械10 治具3番 3分 | 機械1 治具1番 4分 | 機械2 治具2番 3分 | 機械5 治具3番 5分 |
| 製品10 | 機械9 治具4番 1分 | 機械4 治具1番 1分2分 | 機械10 治具2番 9分 | 機械3 治具3番 5分 | 機械5 治具1番 5分 | 機械7 治具2番 3分 | 機械8 治具3番 1分2分 | 機械6 治具4番 8分 | 機械2 治具1番 4分 | 機械1 治具2番 2分 |

● 処理する機械
● 使用する治具番号
● 処理に必要な時間
表示データは、使用する治具番号である。

作業数10のスケジュールを対象に行った。それに使用したデータを表2および表3に示す。表2は各製品の納期を示し、表3は製品別に各作業で使用する機械および治具の種類と加工時間を示す。なお製品の移動と治具の交換に必要な時間は一律に1分としている。

収束性の検証

前項で示したデータを使用して収束性の検証を行った。このシミュレーションにおけるGAのパラメータはつぎのように設定した。

- 遺伝子長 100
- 交叉率 1.0
- 交叉法 CX法
- 突然変異率 0.01
- 突然変異法 SWAP法
- 集団の個体数 100
- 世代数 500

以上の条件でシミュレーションを行なった結果、図10b)のガントチャートを得ることができた。なお、横軸の単位は分である。図10a)に示す初期世代から出発し、239世代で図10b)に示す結果を得た。表4に示すように、いずれの製品も完成時間は納期の要求を満足させている。

初期世代のガントチャートでは隙間が多く、無駄な時間が存在している。しかし、図10b)のガントチャートは隙間が少なく、すべての製品が納期に遅れることのないスケジュールになっている。このことより世代を重ねるごとに良好な解が得られ、GAが有効に働いていることが証明できた。なお、239世代までの計算

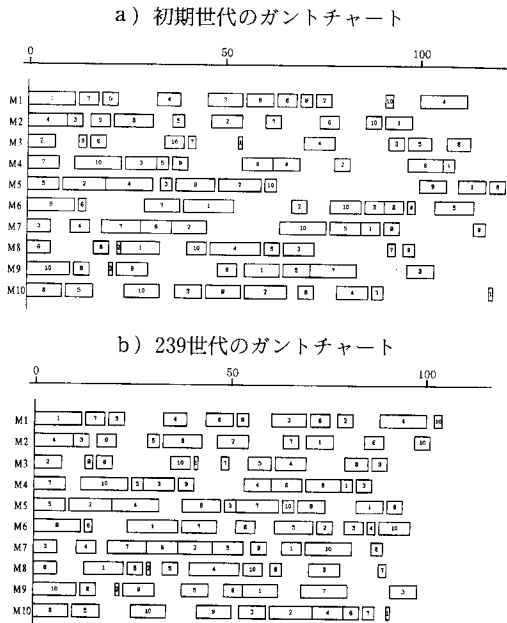


図10：シミュレーション結果1

表4：納期と完成時間の比較

| | 納期 | 初期世代の 完成時間 | 239世代の 完成時間 |
|------|------|---------------|----------------|
| 製品1 | 102分 | 120分 | 92分 |
| 製品2 | 94分 | 83分 | 87分 |
| 製品3 | 99分 | 105分 | 99分 |
| 製品4 | 117分 | 113分 | 101分 |
| 製品5 | 94分 | 115分 | 72分 |
| 製品6 | 91分 | 80分 | 90分 |
| 製品7 | 100分 | 95分 | 91分 |
| 製品8 | 99分 | 123分 | 95分 |
| 製品9 | 85分 | 108分 | 75分 |
| 製品10 | 105分 | 94分 | 105分 |

完成時間が納期に間に合わない場合は で表示する

時間はワークステーション Sun Sparc Classic 上でほぼ115秒となっており、実用的な計算時間で良好なスケジュールを取得することができた。

世代数ごとの最良値の変化を図11に示す。図では横軸に世代数、縦軸にその世代の最良値を示している。

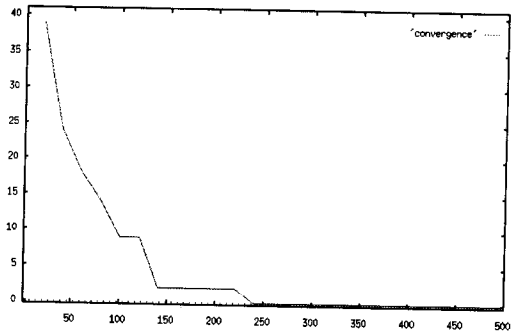


図11：最良値の推移

再構築の性能の検証

次にスケジュールに遅延が生じた際に実行する再構築の有効性を検証する。再構築に関する状況設定は次のようになっている。

- 前述のシミュレーションで得られた図10b) のガントチャートに沿って生産工程を実行中、製品6の第3作業に15分の遅れが生じたため、処理時間4分の予定が19分を必要としたとする。
- 遅延が生じたままの作業の順序を変更せずに処理を続けた場合のガントチャートは図12a) となっている。この結果、完成時間は表5に示すように納期に間に合わないものが出てくる。
- 製品6の第3作業の予定終了時間を1分過ぎた時点で再構築を開始する。
- このシミュレーションにおけるGAのパラメータは前述のものと同じである。

以上の条件でシミュレーションを行なった結果、図12b) のガントチャートを得た。

図12a) のガントチャートでは製品6の第3作業に遅れが生じているため、製品6に関して第4作業以下すべての作業の開始時間が遅れている。またこれらの作業終了後に開始予定の他の作業も同じように遅れるため、7つの製品が納期に間に合わない結果となった。これと比較して再構築を行うことで得られる図12b) のガントチャートでは納期に遅れることのないスケジュールとなっている。

納期が変更になった場合の対処方法

作業を実行中に納期が変更になった場合は、変更が明らかになった時点で新しい納期を入力し、遅延時と同様の方法で再構築を行うことで対処できる。この方式についても先ほど同様のシミュレーションを行い、納期遅れの縮小を確認した。

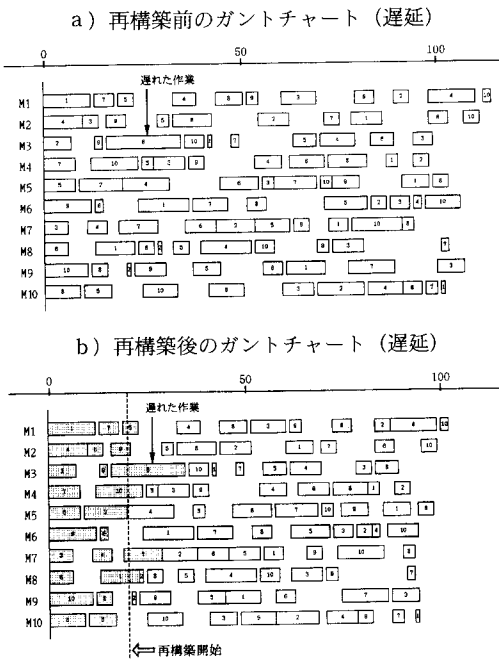


図12：再構築前後のガントチャート比較

表5：再構築による完成時間の改善

| 遅延時 | 納期 | 再構築前の完成時間 | 再構築後の完成時間 |
|------|------|-----------|-----------|
| 製品1 | 102分 | 103分 | 95分 |
| 製品2 | 94分 | 99分 | 93分 |
| 製品3 | 99分 | 108分 | 95分 |
| 製品4 | 117分 | 111分 | 100分 |
| 製品5 | 94分 | 83分 | 72分 |
| 製品6 | 91分 | 104分 | 89分 |
| 製品7 | 100分 | 104分 | 94分 |
| 製品8 | 99分 | 104分 | 99分 |
| 製品9 | 85分 | 81分 | 82分 |
| 製品10 | 105分 | 115分 | 103分 |

完成時間が納期に間に合わない場合は で表示する

8 結 言

ジョブショップ型のスケジューリング問題を対象と

した遺伝的アルゴリズムを適用したスケジュール作成方式を提案した。本方式では製品の移動時間と治具の交換時間を考慮したスケジュール作成や、不慮の事故や納期の変更が発生した場合のスケジュール再構築を実用的な時間で行うことができた。

今後の課題としては治具の交換回数の最小化などの制約条件の追加が挙げられる。

参 考 文 献

- [1] R. Conway, W. Maxwell, L. Miller: Theory of Scheduling, Addison-Wesley, 1967.
- [2] D. E. Goldberg: Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Pub. Co., 412p, 1989.
- [3] Sugato Bagchi, Serddar Uckun, Yutaka Miyabe, Kazuhiko Kawamura: Exploring Problem-Specific Recombination Operators for Shop scheduling, Proc. of 4th Int. Conf. on Genetic Algorithm, pp.10~17, 1991.
- [4] Ryouhei Nakano, Takeshi Yamada: Conventional Genetic Algorithm for Job Shop Problems, Proc. of 4th Int. Conf. on Genetic Algorithms, pp.474~480, 1991
- [5] Zbigniew Michalewicz: Genetic Algorithms + Data structures = Evaluation Program, Springer-Verlag, p.336, 1992.
- [6] S. Kobayashi, I. Ono, M. Yamamura: A Genetic Algorithm Applicable to Large-Scale Job-Shop Problems, Proc. of 6th Int. Conf. on Genetic Algorithms, pp.506~511, 1995.
- [7] G. Shi, H. Iima, N. Sannomiya: A New Encoding Scheme for Solving Job Shop Scheduling Problems by Genetic Algorithms, Proc. of 35th IEEE Conf. on Decision and Control pp.4395~4400, 1996.
- [8] G. Shi, H. Iima, N. Sannomiya: Comparison of Two Genetic Algorithms in Solving Tough Job Shop Scheduling Problems, Trans. IEE of Japan, Vol.117-c, No.7, pp856~864, 1997.
- [9] Guoyong Shi: A genetic algorithm applied to a classic job-shop scheduling problem, International Journal of Systems Science, Vol.28, No. 1, pp.25~32, 1997.

- [10] F. Zhang, Y. F. Ahang, A. Y. C. Nee: Using Genetic Algorithms in Process Planning for Job Shop Machining, IEEE Transactions on Evolutionary Computation, Vol.1, No.4, November, pp.278~289, 1997.
- [11] 小林重信, 小野功, 山村雅幸: スケジューリング問題に対する遺伝的アルゴリズムの一構成法, 計測自動制御学会論文集 Vol.27, No.5, 1995.
- [12] 山田武士, 中野良平: 遺伝的局所探索によるジョブショップスケジューリング問題の解法, 情報処理学会論文誌 Vol.38, No.6, 1997.
- [13] 小野俊彦, 池田武史, 土性雅史, 田代順一: 遺伝的アルゴリズムによる最適問題の解法, 福岡工業大学言語情報工学研究所彙報 第8巻, pp.19~29, 1997.
- [14] 小野俊彦, 田代順一, 横井智幸: 遺伝的アルゴリズムと最適問題, 福岡工業大学言語情報工学研究所彙報 第9巻, pp.11~22, 1998.
- [15] 三宮信夫, 喜多一, 玉置久, 岩本貴司: 遺伝的アルゴリズムと最適化, 朝倉書店, 1998.
- [16] 田代順一, 小野俊彦: 再構築可能なスケジューリング問題への遺伝的アルゴリズムの適用, 第17回SICE九州支部学術講演会予稿集, pp.285~286, 1998.