

ジョブショップスケジューリング問題における デッドロック回避方法の提案

日大生産工 (学部) ○奥村 幸典 日大生産工 山内 ゆかり

1 まえがき

ジョブショップスケジューリング問題

(job-shop scheduling Problem : JSP)はN 個の仕事をM台の機械で処理することを考えたときに、各仕事を処理する機械の順序(技術的順序)、および、各機械上での各仕事の処理時間のもと、すべての仕事を処理し終えるまでの作業時間を最小にするような順序を決定する問題である¹⁾。本報告では、JSPに遺伝的アルゴリズム(Genetic Algorithm : GA)を用いた場合に起こるデッドロックの回避方法として初期集団生成時に技術的順序を考慮して生成する方法と集団の評価時に処理し終えた仕事の数を考慮して評価する方法を提案する。

2 JSPに対するGAの問題点

JSPに対しGAを用いて近似解を求めた場合に技術的順序を満たさないスケジュールを組んでしまう可能性があった。4仕事4機械のJSPにGAを用いて最適解を求めた場合、集団数30で実行すると251世代目に最適解を得られた。しかし、10仕事10機械のJSPにGAを適用した場合、初期集団で生成された遺伝子が技術的順序を満たさない実行不可能なスケジュールとなり、デッドロックが起きていた。この原因は、今回の実験でのGAの染色体表現が各機械でのスケジュールを直接表現していることに起因すると考えられる。これは、JSPのスケジュールの総数は、m機械n仕事の問題で $(n!)^m$ となるため、4仕事4機械の問題であると技術的順序を満たしてスケジュールを集団の中でいくつも組むことができるため最適解を求めることができたが、10仕事10機械の問題ではスケジュールの総数が膨大なので、集団の中に技術的順序を満たすスケジュールを組むことができなかつたからである。また、技術的順序を満たさない遺伝子の明確な評価基準がなく、デッドロックをおこす殆どの遺伝子間で評価値に差がないため最適化が行われず、集団の成長が遅くなり実行時間が長くなるという問題がある。

3 提案手法及び実験方法

この問題を解決するには以下の3つの方法が考えられる。

1 初期集団生成時に組むことができるスケジュールの確率を上げる

2 処理できる仕事を増やすために途中までの処理することができる部分を評価する

3 法染色体表現自体を常に技術的順序を満たすことができる表現にする

本研究では、1と2の方法を提案する。

3. 1 順序重み付け法

GAの初期集団生成は通常、各遺伝子の値はランダムに決定するが、本研究では技術的順序に基づき各機械での各仕事に重みを付け、処理の順序が早い方ほど最初の方にスケジュールが組まれるように生成する。表1の問題であるとM₁をみるとJ₁だと1番目、J₂だと2番目、J₃だと3番目なのでJ₁がスケジュールの始めのほうに組みやすくなり、J₃が後のほうに組みやすくなる。

表1 3仕事3機械JSPの問題

	1番目の処理	2番目の処理	3番目の処理
J ₁	(M ₁ ,3)	(M ₂ ,3)	(M ₃ ,2)
J ₂	(M ₃ ,2)	(M ₁ ,5)	(M ₂ ,3)
J ₃	(M ₂ ,2)	(M ₃ ,6)	(M ₁ ,1)

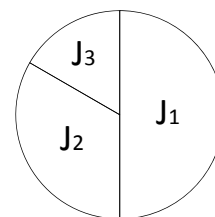


図1. 各仕事を選ばれる確率

3. 2 部分評価法

各個体を評価する時に、通常、技術的順序を満たしていない場合には、技術的順序を満たしている時に得ることができる最大処理時間より低い評価値を与える。本研究では、技術的順序を満たすスケジュールでなかった場合にそれまで処理をすることができた仕事の数を評価する。

評価 = 考慮する時間 × (処理する仕事数 - 処理可能な仕事数)

これにより途中までの処理できる作業を考慮した評価を得られる。考慮する時間は今回50とした。これにより処理できる仕事の数が0の場合は+5000、50の場合は+2500となる。

それぞれの方法での効果を見るために初期集団の技術的順序を考慮せず生成し、デッドロックの起きた遺伝子の評価を同一としたGA(SGA)で求めた場合と比較した。

ここで、今回の実験で用いる評価方法について説明する。評価値は処理時間を表しているの値が小さい方がいい評価となる。技術的順序を満たすスケジュールを組んだら作業時間が評価値となる。技術的順序を満たさないスケジュールを組んだ場合には2つの評価方法を使った。

方法1 最大処理時間(\mathcal{M})の2倍から途中までの処理時間(\mathcal{T})を引いた値が評価値(\mathcal{V})となる。式は以下ようになる。

$$\mathcal{V}_1 = 2\mathcal{M} - \mathcal{T} \quad (1)$$

方法2 部分評価法を用いた評価方法で最大処理時間に総仕事数(\mathcal{J})と処理可能な仕事数(\mathcal{H})のから処理していない仕事数を求めその分の時間を増やす。式は以下ようになる。

$$\mathcal{V}_2 = \mathcal{M} + 50(\mathcal{J} - \mathcal{H}) \quad (2)$$

SGA、順序重み付け法(提案1)、部分評価法(提案2)、での初期集団生成と評価方法の関係を表2に示す。

表2 初期集団生成と評価方法の関係

		初期集団生成	
		ランダム	順序重み付け法
評価	V1	SGA	提案1
	V2	提案2	

これらの方法により得られた処理時間、処理した仕事数、探索時間の変化を比較する。実験結果の値は10試行での平均をとった。尚、実験の問題はCS410/510SS Project Job Shop Scheduling²⁾のベンチマーク問題ft10を使用した。

4 実験結果および検討

4.1 順序重み付け法

図2、図3の結果をみると評価値は他の二つの方法より良い評価値を得ることができ、探索時間を少し速くすることができた。処理できる仕事数はSGAより多かった。しかし、解を得るまでには至らなかった。

4.2 部分評価法

図2、図3の結果をみると評価値はSGAより

良い評価値を得る事ができた。また、評価が5100くらいに達するまでの探索時間が一番速かった。処理できる仕事数は他の二つの方法より多く、とても速い速度で増えていった。しかし、解を得るまでには至らなかった。

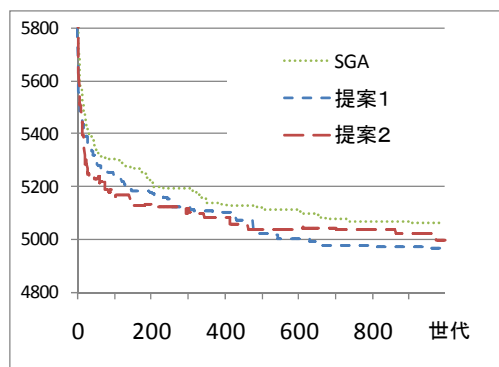


図2. 処理時間の比較

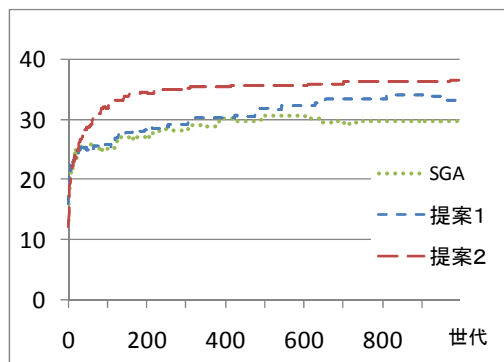


図3. 処理できる仕事数の比較

5 まとめ

本研究では、JSPにGAを用いた場合に起こるデッドロックの回避方法を提案した。SGAで行った場合より多少良い評価値を得ることや探索時間を速くすることができた。しかし、技術的順序を満たすことができるスケジュールを組むことができるようになるほどの効果はなかった。もっと小規模なJSPに対して提案手法を用いるか、集団数および世代数を増やして実験することで、有効性が検証できる可能性がある。

今後の課題としては、染色体表現自体を常に技術的順序を満たすことができる表現にする方法を考える必要がある。また、その場合での今回の方法が探索時間を短くするのに有効であるかを確認したい。

「参考文献」

- 1) ジョブショップスケジューリング問題, <http://mikilab.doshisha.ac.jp/dia/research/report/2002/0504/018/report20020504018.html>
- 2) CS410/510SS Project Job Shop Scheduling, <http://www.cs.pdx.edu/~bart/cs510ss/project/jobshop/jobshop/>