

## ジョブショップスケジューリング問題における 交叉 dMSXF の解探索性能の検証

花 田 良 子<sup>†</sup> 廣 安 知 之<sup>††</sup> 三 木 光 範<sup>††</sup>

dMSXF は JSP の交叉の 1 つである MSXF を改良した交叉である。dMSXF では局所探索過程において、解遷移は温度パラメータではなく個体間の距離による決定的な方法で受理判定が行われる。dMSXF は TSP において良好な解探索を示しているが、JSP においては検討されていない。本研究では dMSXF の仕組みを導入した JSP の交叉を考案し、解探索性能を検証する。また、突然変異の操作として、温度パラメータによらない MSMF についても考案し、解探索性能の向上を図る。

### Examination of deterministic Multi step Crossover Fusion for Job shop Scheduling Problem

YOSHIKO HANADA,<sup>†</sup> TOMOYUKI HIROYASU<sup>††</sup>  
and MITSUNORI MIKI<sup>††</sup>

The dMSXF is an improved crossover method of the MSXF which is one of promised methods of JSP. In the process of local search of this crossover, transition of individual is accepted or rejected by the deterministic rule derived from the distance of parents instead of the temperature parameter. It was shown that dMSXF had effective search in TSP. However its effectiveness has not been examined in JSP. In this paper, we contrived a crossover which has mechanisms of the dMSXF for JSP and examine its search performance. In addition, we introduce a deterministic MSMF mechanism as mutations for improvement its effectiveness.

#### 1. はじめに

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、所与の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 (作業) の処理時間のもと、すべての仕事を処理し終えるまでの総所要時間 (makespan) を最小にするような仕事の処理順序を決定する問題である。ただし、各機械の種類はすべて異なり、同時に複数の作業を処理することができない。また、各作業は、与えられた処理時間をかけて、各機械上で中断されることなく処理される。JSP は NP-困難な順序づけ問題の中でも特に難解な問題の一つであり、その産業上の応用分野の広さから、様々な研究がなされてきた。近年では、確率的最適化手法の中でも特に遺伝的アルゴリズム (genetic algorithm: GA) を用いた研究がさかんに

行われている。JSP の GA の解法として、これまでに、CCM<sup>1)</sup>、ISM<sup>2)</sup> などの世代交代モデル、また、主要な操作である交叉においては、問題固有の性質を考慮した SXX<sup>3)</sup>、inter machine JOX<sup>4)</sup>、MSXF<sup>5)</sup>、EDX<sup>6)</sup> など有効な手法が提案されている。

JSP においては GA と近傍探索法と組み合わせることによって性能が向上することがよく知られており、交叉 MSXF、EDX は JSP 固有の近傍構造を用いた SA (Simulated Annealing) を組み込むことで優秀な解探索能力を有している。しかしながら、これらは近傍個体への解遷移の際に、解探索性能に大きく影響を与える温度パラメータ  $T$  を必要とし、 $T$  の設定は探索のそれぞれの段階での評価値に依存するため適切な値を決定するのは難しい。そこで、MSXF と同様に多段階に行う局所探索の仕組みを持ち、解の遷移の際には温度パラメータではなく、個体間の距離による決定的な方法で解を遷移を行う交叉 dMSXF (deterministic Multi Step Crossover Fusion)<sup>7)</sup> が提案されている。dMSXF のもととなった MSXF は JSP のため考案された交叉であり、良好な解探索性能を示している。一方、dMSXF は他の順序付け問題である巡回セール

<sup>†</sup> 同志社大学大学院工学研究科

Faculty of Engineering, Doshisha University

<sup>††</sup> 同志社大学工学部

Faculty of Engineering, Doshisha University

スマン問題 (traveling salesman problem : TSP) を対象としており、大規模な問題で非常に良好な性能を示しているが、JSP を対象として検証されていない。そこで本研究では、dMSXF の仕組みを導入した JSP の交叉を考案し、その解探索性能を検証する。また、MSXF を GA に採用する際には、MSXF と同様に温度パラメータを用いた近傍探索変異 MSMF (Multi Step Mutation Fusion)<sup>5)</sup> が併用されているが、dMSXF の原論文では突然変異的操作である dMSMF の仕組みについては採用されていない。そこで、本研究では温度パラメータによらない dMSMF (deterministic MSMF) についても考案し、dMSXF と併用することで、解探索性能の向上を図る。

## 2. dMSXF

dMSXF<sup>7)</sup> は池田らによって提案された交叉であり、山田らによる多段階探索交叉 MSXF<sup>5)</sup> を改良した新たな交叉手法である。dMSXF のアルゴリズムを以下に示す。

- Step 1 探索初期点  $x_1$  を  $x_1 = p_1$  とする。
- Step 2 ステップ  $k$  における探索点  $x_k$  の近傍にある  $\mu$  個の解群を近傍  $N(x_k)$  とする。  $N(x_k)$  のすべての近傍解  $y_i$  はかならず  $d(y_i, p_2) < d(x_k, p_2)$  を満たさなければならない。
- Step 3  $N(x_k)$  の中でもっとも良い解  $y_i$  を選択する。
- Step 4 終了条件 (例.  $k = k_{max}$  あるいは  $p_2 \in N(x_k)$ ) が満たされていない場合、 $x_{k+1} = y_i$  とし、Step 2 にもどる。
- Step 5  $x_1 \sim x_{k_{max}}$  の中でもっとも良い解を  $p_1$  と置き換える。次に  $p_2$  について、他の解  $p_3$  を取り出し、 $p_2$  から  $p_3$  に向けて探索をすすめる。

交叉で親個体  $p_1$  から親個体  $p_2$  に向けて局所探索を行う過程において、MSXF では解遷移が温度パラメータに依存していたのに対し、dMSXF では、近傍の解  $y_i$  はすべて解  $x_k$  よりも  $p_2$  に近い個体に制限し、 $x_{k+1}$  が  $x_k$  よりも劣っていたとしても必ずそのまま探索を進めることで解遷移を決定的に行う。

## 3. JSP のための dMSXF

dMSXF を問題に適用する際には、個体間の距離の定義および  $(d(y_i, p_2) < d(x_k, p_2))$  となるような近傍とその生成法を設計しなければならない。本節ではそれらについて述べる。

### 3.1 個体間の距離

JSP において個体間の距離として、様々なものが提案されているが、本研究では、機械上での各作業の絶対的な位置に基づいた距離である  $I_2$  距離を用いる。 $s_a, s_b$  をスケジュールとする。 $M$  を機械数、 $N$  を仕

事数とする。仕事  $p$  に属する作業で機械  $q$  で処理されるものを  $o(p, q)$  とし、仕事  $i$  に属する作業の集合を  $J_i (= \{o(i, k) | k = 1, \dots, M\})$  とする。作業  $o$  に対して、 $L(o)$  を  $o$  が実行される順番とする。 $s_a, s_b$  の仕事  $i$  についての  $I_{2i}$  距離  $I_{2i}(s_a, s_b)$ 、および  $s_a, s_b$  の  $I_2$  距離  $I_2(s_a, s_b)$  は次のように定義される。

$$I_{2i}(s_a, s_b) = \sum_{k=1}^M |L(o_a(i, k)) - L(o_b(i, k))| \quad (1)$$

$$I_2(s_a, s_b) = \sum_{k=1}^N I_{2k}(s_a, s_b) \quad (2)$$

本研究で考案する交叉では親個体  $p_1$  から親個体  $p_2$  に向けて  $I_2$  距離が小さくなる方向に解遷移が行われる。

### 3.2 近傍個体の生成

本研究での探索対象スケジュールはアクティブスケジュールとし、近傍個体には、MSXF や EDX で用いられているアクティブ CB 近傍<sup>5)</sup> を用いる。アクティブ CB 近傍はクリティカルブロック上の作業の移動に基づく近傍で、生成されたスケジュールがアクティブ・スケジュールとなるよう GT 法に基づく修正操作が加えられたものである。dMSXF の Step 2 において生成される  $x_k$  近傍個体  $y_i$  は、 $(d(y_i, p_2) < d(x_k, p_2))$  を満たさなければならない。これを満足させるため、まず、 $x_k$  と  $p_2$  に対して、次に示す手法で得られる移植個体  $x'_k$  を生成し、その個体についてのアクティブ CB 近傍を生成する。

- Step 1  $p_2$  において  $x'_k$  へ投入順序を反映させる仕事を 1 つ選ぶ。
- Step 2 Step 1 で決めた仕事について、 $p_2$  から  $x'_k$  へ投入位置を保存するようにコピーする。
- Step 3 Step 2 でコピーしなかった仕事について、各機械で  $x_k$  から  $x'_k$  へ順序を保存するように左から右へあいている投入位置へコピーする。

Step 1 における仕事の選択には、1) ランダム、2)  $I_{2i}$  距離が最大の仕事  $i$ 、3)  $I_{2i}$  距離が大きいほど選ばれやすくなるよう、 $I_{2i}$  距離の大きい順に仕事をソートし、番号  $k$  に反比例する確率で選択するといった方法など様々なものが考えられる。本研究では 3) を採用している。近傍生成のもととなる移植個体の生成することで、移植する仕事  $i$  についての  $I_{2i}$  距離において、 $(d(y_i, p_2) < d(x_k, p_2))$  はほぼ満たされる。ただし、 $I_2$  距離については  $(d(y_i, p_2) < d(x_k, p_2))$  が満たされるとは限らない。なお、この操作で得られるスケジュールはアクティブスケジュールであるとは限らないので、GT 法によるアクティブスケジュールへの修正操作を適用する。

## 4. JSP のための dMSMF

MSXF を GA に採用する際、MSXF は親 2 個体の

距離が近すぎ場合には有効に働かないことから近傍探索変異 MSMF が併用されている。MSXF で親個体  $p_1$  から  $p_2$  に近づいていくのに対して、MSMF は親個体から遠ざかる方向に探索が進む。なお、MSXF と同様に解遷移は温度パラメータに依存する。

交叉 dMSXF は MSXF を決定的に行う手法であり、突然変異的操作である MSMF の仕組みについては採用されていない。交叉 dMSXF についても MSXF と同様に、親 2 個体が近い場合は有効に働かない。そこで、温度パラメータによらない dMSMF (deterministic MSMF) を dMSXF と併用することで、解探索性能の向上を図る。

親個体  $p_1, p_2$  の距離がある値よりも小さい場合は、次に示す操作を親個体  $p_1$  に適用することで、両個体から離れていく方向に探索を進めていく。これは MSMF と同様に突然変異的な多段階の局所探索である。この探索を進めるにあたって、小野らによって考案された突然変異 job-based shift change<sup>4)</sup> を適用している。dMSXF では  $p_1$  から  $p_2$  に向けて  $I_2$  距離が小さくなる方向に解遷移が行われる。 $I_2$  距離は機械上での各作業の絶対的な位置に基づいた距離であり、job-based shift change は、ある仕事に属する作業全体を左あるいは右にシフトする操作であるので、 $p_1$  に job-based shift change を適用していくことで  $p_1$  から  $I_2$  距離が大きくなる方向に探索が進むこととなる。 $p_1$  と  $p_2$  が非常に近い場合には、この操作を適用することで、2 個体から遠ざかる方向に探索が進み、母集団の偏りが緩和されることでより解探索性能の向上を図ることができると思われる。

- Step 1 探索初期点  $x_1$  を  $x_1 = p_1$  とする。
- Step 2 ステップ  $l$  における探索点  $x_l$  に対して、ランダムに 1 つ選んだ仕事について、それに属する作業全体を左あるいは右にシフトする
- Step 3 探索点  $x_l$  の  $\lambda-1$  個のアクティブ CB 近傍および  $x_l$  からなる解群  $N(x_l)$  を生成する
- Step 4  $N(x_l)$  の中でもっとも良い解  $y_i$  を選択する。
- Step 5  $x_2 \sim x_{l_{max}}$  の中でもっとも良い解を  $p_1$  と置き換える。

## 5. 数値実験

dMSXF および dMSMF の解探索性能を検証する。dMSXF で必要なパラメータはステップ数  $k_{max}$  と近傍個体数  $\mu$  であり、dMSMF で必要なパラメータもステップ数  $l_{max}$  と近傍個体数  $\lambda$  である。dMSXF では、親個体  $p_2$  のある仕事の絶対位置を探索解に採用した移植個体を生成することで、親個体  $p_1$  から  $p_2$  に近づけていく探索を行っているため、 $k_{max}$  が仕事数  $N$  よ

りも非常に大きな値をとったとしても親 2 の近傍を撰動するだけであり、 $k_{max}$  は問題の仕事数に近い値を用いるのが適切であると考えている。同様に  $l_{max}$  についても、 $p_1, p_2$  から離れるという目的で探索を進めるため、仕事数  $N$  に近い値を用いるのが妥当であり、 $k_{max}=l_{max} \sim N$  として問題はないと考えられる。また、近傍個体数についても、近傍個体にはクリティカルブロックをベースとしたアクティブ CB 近傍を用いていることから生成可能な個体数は各問題についてある程度の限界があり、予備実験で十分予測可能であるため、限界の範囲内で適切な値を選ばばよく、設定が困難なパラメータではない。

本実験では、母集団でまったく同じ解が存在した場合は、一方を初期化している。ランダムに生成した初期解と探索の進んだ解に対して、dMSXF を適用したときに探索の進んだ解付近の個体に置き換わる可能性があり、初期化が有効に働かない恐れがある。そこで、個体を初期化する際に、dMSMF の操作を適用することを考える。dMSMF はある個体から  $I_2$  距離が離れていく操作であると同時に、多段階の局所探索でもあり、それを適用し、ある程度良好な解を得ておくことで上記の問題を回避することが可能であると考えられる。初期化の際に用いる dMSMF では Step 5 において  $x_1 \sim x_{l_{max}}$  の解で最も良好な解を  $p_1$  と置き換える。なお、初期母集団生成時においても各個体に dMSMF を適用し、ある程度の良好な解から GA による探索を始める。これは、TSP を解く場合には、初期解に 2-opt 法を適用することで探索性能の向上を図るといったように、初期解にヒューリスティックな方法を適用することで、より効率的に探索を行うことを目的としている。

### 5.1 dMSXF, dMSMF の探索性能

dMSXF の解探索性能および dMSMF を適用することによる効果を検証する。対象問題として、ft10, ft20 および abz5 を用いる。これらは比較的小規模な問題であるので、 $k_{max}=l_{max}=10$ 、近傍個体数  $\mu=\lambda=5$  とする。dMSMF を適用する条件は親 2 個体の  $I_2$  距離が (機械数)  $\times$  (仕事数)  $\times 0.1$  よりも小さい場合、あるいは評価値が同じ場合に適用する。母集団サイズを 100 とし、終了条件は 200 世代評価値が変わらない時点で終了する。なお、個体の評価時には、LR 法<sup>5)</sup> を採用し、評価個体については、 $\mu$  個の任意のアクティブ CB 近傍を生成し、その中でもっとも良い解と置き換えることで評価値の改善確率の向上を測っている。

Table 1 に最適解を得た回数、および最適解を得るのに必要とした評価計算回数を示す。これらは 30 試行の結果である。比較手法として、CCM<sup>1)</sup> に準ずるモ

デルを用いる。CCMについては交叉に inter machine JOX, 突然変異には job based shift change を用いる。母集団サイズを 100 とし, 交叉では 1 ペアに対し 20 子個体生成している。Table 1 から, dMSXF は高い確率で最適解を得ており。また, dMSMF を入れることで解探索性能が向上していることが可能であることがわかる。なお, パラメータ  $k_{max}$  および  $\mu$  については, ft10, abz5 についてはその設定によらず Table 1 に示すように高い確率で最適解を得ていること, また, ft20 については仕事数が多い問題であるため  $k_{max}$  が小さいと最適解を得る確率が減少することが予備実験より分かっている。

表 1 テスト問題による性能比較

	dMSXF+dMSMF	dMSXF	CCM
ft10	30 (1.0x10 <sup>5</sup> )	30 (1.1x10 <sup>5</sup> )	30 (1.4x10 <sup>5</sup> )
ft20	30 (5.7x10 <sup>5</sup> )	24 (7.7x10 <sup>5</sup> )	12 (1.6x10 <sup>5</sup> )
abz5	30 (1.3x10 <sup>5</sup> )	26 (1.0x10 <sup>5</sup> )	1 (1.8x10 <sup>5</sup> )

## 5.2 10 tough problem における解探索性能

10tough 問題での解探索性能を検証する。これらは, 仕事数が 15 ~ 20 であり,  $k_{max}=l_{max}=20$  としている。近傍個体数  $\mu=20$  とする。母集団サイズを 400 とし, 終了条件は 200 世代評価値が変わらない時点か, あるいは総評価計算回数が  $5.0 \times 10^7$  となった時点で終了する。この打ち切りは比較手法である EDX で用いられている条件である。なお, 母集団サイズについては EDX は 50, MSXF は 20 であり, 考案する手法ではサイズが大きいが, dMSXF が並列処理に非常に親和性が高いことから, 問題ないと考えている。

Table 2 に解探索性能を示す。これらは最適解を得た回数, あるいは最良解を比較したものである。比較手法として, JSP において近傍構造を考慮した有力な交叉とされている MSXF, および EDX である。これらの結果より, 考案している手法が多数の問題で良好な結果を得ていることがわかる。また, MSXF, および EDX は SA や SB(shifting bottleneck) 法など他の有力な近似解法と比較して, 解探索性能が良いとされている手法であり, 本研究で考案した JSP における dMSXF および dMSMF が有効な探索手法であることがわかる。

## 6. ま と め

dMSXF は池田らによって提案された交叉であり, MSXF を改良した新たな交叉手法である。交叉で親個体 1 を親個体 2 に近づける局所探索の過程において, MSXF では解遷移が温度パラメータに依存していたのに対し, dMSXF は個体間の距離による決定的

表 2 10tough 問題での性能比較

	dMSXF+dMSMF	MSXF	EDX
abz7	<b>658</b> -	678 -	670 -
abz8	<b>668</b> -	686 -	683 -
abz9	<b>683</b> -	697 -	686 -
la21	*1046 (1/30)	*1046 (9/30)	*1046 (1/10)
la24	*935 (6/30)	*935 (4/30)	*935 (4/10)
la25	*977 (16/30)	*977 (9/30)	*977 (4/10)
la27	*1235 (11/30)	1235 (1/30)	1236 -
la29	<b>1160</b> -	1166 -	1167 -
la38	*1196 (29/30)	1196 (21/30)	*1196 (1/10)
la40	1224 -	1224 -	1224 -

な方法で解遷移を行う。dMSXF の原論文では TSP を対象問題としており, 良好な解探索性能を示しているが, JSP を対象として検証されていない。そこで, 本研究では, dMSXF を JSP に拡張し, dMSXF の仕組みを導入した交叉を考案した。また, 突然変異的な多段階局所探索として温度パラメータによらない dMSMF についても考案し, dMSXF と併用することで解探索性能の向上を図った。代表的な小規模のテスト問題および 10tough 問題に本手法を適用した結果, 非常に良好な解探索性能を示した。

## 参 考 文 献

- 1) Isao Ono et. al.: A Genetic Algorithm Taking Account of Characteristics Preservation for Job Shop Scheduling Problems, Proc. of the International Conference on Intelligent Autonomous Systems 5, pp. 711-718, 1998
- 2) 池田心, 小林重信: 生得分離モデルを用いた GA と JSP への適用, 人工知能学会誌, Vol. 13, No. 5, pp. 530-538, 2002.
- 3) 小野, 佐藤, 小林: サブシーケンス交叉と GT 法に基づくジョブショップスケジューリング問題の進化的解法, 電気学会論文誌 C, Vol.117-C, No.7. pp. 888-895, 1997
- 4) 小野功, 小林重信: Inter-machine JOX に基づく JSP の進化的解法, 人工知能学会誌, Vol. 13, No. 5, pp. 780-790, 1998
- 5) 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
- 6) Jun Sakuma and Shigenobu Kobayashi: Extrapolation-Directed Crossover for Job-shop Scheduling Problems: Complementary Combination with JOX, GECCO 2000, pp. 973-980, 2000
- 7) Kokolo Ikeda, Shigenobu Kobayashi: Deterministic Multi-step Crossover Fusion: A Handy Crossover for GAs, pp162-171. PPSN7, 2002