遺伝的アルゴリズムによる
スケジューリング問題の最適化

戸 島 貴 子
奥 田 和 重

1．緒　　論

スケジューリング問題は，所定の生産計画期間内で生産の対象となる品物を，どの生産設備で，いつ，誰が，いかなる作業を行うかという具体的で詳細な時間日程計画を立てようとするときに生起する問題である。一般的にスケジューリングの対象となる問題は，(1)オペレーション・スケジューリング，(2)プロジェクト・スケジューリング，(3)組立ラインバランシングの3つである。オペレーション・スケジューリングは，機械加工のように個々の生産設備（工作機械）が比較的独立した状態で配置されており，各生産設備で作業が実施されるときに加工の対象となる品物(加工物)の作業順序を決定するものである。プロジェクト・スケジューリングは，工場建設や大規模な土木工事などのように，時間的に長期間にわたる大規模なスケジューリングで，複雑に関連する多くの作業を円滑で効果的な作業実施ができるように作業順序を決定する。最後の組立ラインバランシングは，組立工程のように直列的に結ばれている作業場に，先行関係のある要素作業群を割り当てることによって，無駄時間を排除して均整がとれるように各作業場での作業時間がほぼ均衡となるようにライン編成を行う［1］。これらのいずれの問題もスケジューリングの対象（慣習的に”ジョブ”と呼ばれている）を適当に組み合わせることによって最適なスケ

ジュールを得る組合せ最適化問題（離散型最適化問題）［2］としてとらえることができ，動的計画法，分岐限界法（ないし分枝限定法），整数計画法，0－1計画法などの組合せ最適化法が適用される。本論文ではとくにオペレーション・スケジューリングを取り上げ，その最適化法について考察する。

オペレーション・スケジューリングに対するこのような組合せ最適化法は，現実的な規模の問題に対して計算量と計算時間の点から実際に利用可能なスケジュールを得ることは不可能である。従来のスケジューリング問題では，そのようなスケジューリングを得るために，最適化をある程度犠牲にして満足できるスケジュールを効率的に求める方法や，最適性は保障されないが現実的なスケジュールを得るためのシミュレーションモデル［3］，エキスパートシステム［4］などが提案されてきた。近年，現実的なスケジュールを効率よく求める方法の1つとして遺伝的アルゴリズム（genetic algorithm）［5］が注目されている。これはHollandらによって提案された方法［6］で，ニュラルネットワークやシュミレーテッドアニーリング（焼きなまし法）などと同様に自然界に存在する生物学的あるいは物理学的な原理を利用しており，とくに遺伝的アルゴリズムは生物進化の原理の基づいている。遺伝的アルゴリズムは元来生物の進化を調べるためのシミュレーションモデルとして開発されたが，その特徴である確率的探索や学習などの概念に基づいた組合せ最適化問題に対する逐次近似最適化法として注目されており，数理計画問題においては巡回セールスマン問題，ナップサック問題，グラフ・ネットワーク問題などに応用されている。本論文では，この遺伝的アルゴリズムをオペレーション・スケジューリングに適用し，その有効性と問題点を検討する。

2．遺伝的アルゴリズムの概略

2．1　用語の定義

遺伝的アルゴリズムは，Hollandらによって1960年代に提案されたもので，1980年代後半から一般に注目されるようになり，工学の分野では最適化法と

して注目されている。最近，わが国においても専門書の出版［7，8，9］や，学会誌などで特集〔10，11，12，13〕が組まれたり連載講座〔14，15〕が掲載されたりしているが，研究領域として未発達であるために，用語や手順が統一されていない。そのために本論文では，用語を次のように定義する〔16，17〕。

遺伝的アルゴリズムは複数個の個体（individual）を対象とし，各個体は遺伝情報を伝える1つまたは複数の染色体（chromosome）を持ち，染色体上で遺伝子（gene）が占める位置を遺伝子座（locus）と呼ぶ。同じ遺伝子座を占めることができる遺伝子を対立遺伝子（allele）という。染色体上におけるこの遺伝子の組み合わせによって規定される表現形式を遺伝子型（genotype）といい，遺伝子型によって発現される個体の特性の表現形式を表現型（phenotype）と呼ぶ。遺伝子型に対応する表現型が存在しないとき，そのような遺伝子を致死遺伝子という。遺伝的アルゴリズムでは，0と1を対立遺伝子とし，そのビット列を遺伝子型として表現することが多い。

個体は，染色体によって特徴づけられ，その集まりを集団（population）という。各個体がより多くの子孫を次世代に残そうとして環境に適応する度合いを適応度（fitness），染色体と適応度を対応づける関数を適応度関数（fitness function）という。遺伝的アルゴリズムでは，繁殖（生殖，複製，reproduction）と淘汰（選択，selection）を繰り返して適応度の高い個体を発見し，それを次世代に残そうとする。この繁殖過程は，淘汰，交叉(crossover)，突然変異（mutation）という3つの遺伝的操作からなる。淘汰は，集団における個体の適応度に応じて次世代に残す個体を確率的に決定し，これによって集団は適応度の高い個体群に収束する。このように個体の増減に適応度が影響を及ぼすことを淘汰圧力（selection pressure）という。確率的に淘汰を行う方式には，集団の適応度の総和に対する各個体の適応度の比によって個体の淘汰を行うルーレット方式，適応度の順位によって淘汰を行うランキング方式，および最大の適応度を持つ個体を無条件に残すエリート保存方式などがある。交叉は，2つの個体間で染色体の遺伝子列を一定の確率で組み替えることで，遺伝子列の同一位置で後半部分を交叉する1点交叉，複数の交叉点で交叉

する多点交叉，あらゆる多点交叉が一様に起こる一様交叉がある。突然変異は，ある個体の染色体上にある遺伝子座の遺伝子をある確率によって対立遺伝子に置き換えることで，取り替え，反転があり，削除，挿入を認めると遺伝子長が変化する。

2．2　遺伝的アルゴリズムの構造

遺伝的アルゴリズムは，複製，交叉，突然変異を経た遺伝子の集団に対して，その集団独自の適合関数により各遺伝子の適合値を計算し，その値に基づいて再び複製，交叉，突然変異を行うことを繰り返すアルゴリズムである。このアルゴリズムの基本形の１つに以下の単純遺伝的アルゴリズム（simple geneteic algorithm）がある［16］。

ステップ１：初期化

ステップ２：

　２．１：適応度の評価

　２．２：淘汰

　２．３：交叉

　２．４：突然変異

ステップ１の初期化では，遺伝子座の遺伝子に対し，ランダムに０または１の対立遺伝子を割り当てることにより決められた個体数の初期集団を生成する。ステップ２の遺伝操作の１サイクルが１世代に当たり，終了条件が満たされるまで繰り返される。ステップ２．１において適応度が評価されその値が決定されても，淘汰を行うときにこの値をそのまま確率に反映させる必要はない。淘汰圧力を調整する手段として適応度の違いを拡大または縮小させる適当な関数を導入することがある。これをスケーリング（scaling）と呼び，次のような関数が用いられる。

線形スケーリング　：$f' = af + b$

シグマ切断　　　　：$f' = f - (\overline{f} - c \times \sigma)$

ベキ乗スケーリング：$f' = f^k$

ここでfは元の適応度，f'は新しい適応度，$\overline{f}$はfの平均，σはその標準偏差，a，b，c，kは任意の定数である。

ステップ2．2の淘汰でルーレット方式を採用すると，m個の個体i，$i=1, 2, \cdots, m$の適応度をf_iとすると，淘汰確率p_{s_i}は次のようになる。

$$p_{s_i}=\frac{f_i}{\sum_{i=1}^{m} f_i}$$

淘汰確率p_{s_i}が大きい個体は，（淘汰されるのではなく）次世代に残す個体に選ばれるので，世代を重ねるごとに適応度の高い個体の遺伝子が集団中に広がっていく。ステップ2．3の交叉では，淘汰によって残された個体の中からランダムに親となる個体ペアを生成し，交叉率p_cで交叉を行う。これによって生成された子の個体ペアを残すのであるが，交叉は確率に従って行われるので，交叉を行わない場合もある。このような場合には，親の個体ペアを残す。淘汰と交叉によって適応度が比較的高い個体が初期の世代で支配的になり，集団の多様性が失われることになる。これを初期収束（premature あるいは early convergence）という。多様性が失われると交叉によって新しい遺伝子を持った個体を生成する効力がなくなり，適応度の向上を期待できなくなる。ステップ2．4の突然変異は，集団の多様性を維持するために個体の近傍に新しい個体を生成するもので，ランダムに選ばれた各個体の遺伝子座の遺伝子を突然変異率p_mで対立遺伝子と入れ換える。これによって集団としてすでに失われた対立遺伝子を回復する可能性があり，突然変異は集団の多様性を維持する有効な手段であるといえるが，交叉によって形成された優秀な形質が突然変異によって破壊される可能性もある。

ある個体iの染色体Aが文字列$A_i=a_{i1}a_{i2}\cdots a_{ij}\cdots a_{in}$で表現されている場合を考える。ここで$a_{ij}$は個体$i$の染色体上$j$番目の遺伝子座に位置する遺伝子で，$n$は染色体長である。$a_{ij}$がとりうる値が対立遺伝子で，その値$V$が0と1のみであるとき，すなわち$V=\{0, 1\}$のとき，染色体はビット列で表現でき，それが遺伝子型となる。遺伝子の値を定めることができない場合，そ

の遺伝子の値をドントケア記号（don't care symbol)*で表し，$V^{+}=\{0, 1, *\}$ と置く。このV^{+}で表現される文字列をスキーマ（schema）といい，Hで表わす。スキーマの中で0または1の値が定まっている遺伝子座の数を次数（order）といい，$o(H)$ で表わす。スキーマHの定義長$\delta(H)$ とは，0または1の値が定まっている最初の遺伝子座と最後の遺伝子座の距離である。

スキーマHを含む世代tの個体数を$m(H, t)$，Hを含む個体の平均適応度を$f(H)$，集団を構成するすべての個体の関する平均適応度を$\overline{f}=\Sigma^{m}_{i=1}f_i / n$で表わす。淘汰にルーレット方式を採用している単純遺伝的アルゴリズムにおけるスキーマHを含む$t+1$世代の期待個体数は，次式で求めることができる［18]。

$$m(H, t+1)=m(H, t)\frac{f(H)}{\overline{f}}$$

交叉率p_cで1点交叉することによってスキーマHが破壊される確率p_dは，染色体長をlとすると，

$$p_d=p_c\frac{\delta(H)}{l-1}$$

で求まる。他方，突然変異によってスキーマHが破壊される確率は$p_m o(H)$である。したがって，淘汰の後に交叉と突然変異によってスキーマHが破壊されるので，残っているスキーマHを含む個体数は，

$$m(H, t)\frac{f(H)}{\overline{f}}\left[1-p_c\frac{\delta(H)}{l-1}-p_m o(H)\right]$$

となる。したがって，$t+1$世代におけるスキーマHを含む個体数の期待値$m(H, t+1)$は

$$m(H, t+1)\geqq m(H, t)\frac{f(H)}{\overline{f}}\left[1-p_c\frac{\delta(H)}{l-1}-p_m o(H)\right]$$

となる。これは次世代に残るスキーマHを含む個体数の期待値の下限を与えるもので，スキーマ定理と呼ばれている遺伝的アルゴリズムの基本的な定理である［19]。しかし，この定理は進化する集団の中でスキーマを含む個体の数を

予測しているだけで，交叉や突然変異によって生成されたスキーマに関する分析はなされていない。

スキーマ定理によると，適応度が高く，定義長が短い，かつ次数が低いスキーマは世代を通じて保存されるので，そのようなスキーマを含む個体の遺伝情報は集団内に蓄積される。これを building block（「機能単位」あるいは「積木」）と呼び，これに関して「他の building block と結合して，最適な染色体を持つ個体が生成される」という building block 仮説がある。この仮説の妥当性に関して小林らは理論的に検討している [16]。

これらの定理と仮説を用いて適応度の高い個体を得るためには，表現型から遺伝型へのコーディング，遺伝型から表現型へのデコーディング，および交叉が適切に行われなければならない。これについては山村らが巡回セールスマン問題を対象に検討しており [20]，Goldberg はコーディングに関して次のように指摘している。

(1) できるだけ少ないアルファベットで構成する。

(2) 与えられた問題が低い次数のスキーマに関係するようにコーディングする。

2．3 遺伝的アルゴリズムによる最適化

遺伝的アルゴリズムによる個体の進化の過程を最適化問題における探索過程であると解釈すれば，このアルゴリズムは最適化問題に対する1つの最適化アルゴリズムを与える。遺伝的アルゴリズムを最適化問題へ適用するために，一般的な最適化問題を次のように記述する。

$$\text{Max. } f(x) \tag{1}$$

$$\text{sub. to } x \in X \tag{2}$$

ここで x は最適化問題の決定変数ベクトルで，X は決定変数 x に許容された値の集合で実行可能領域である。関数 $f(x)$ は実行可能領域 X を含む集合上で定義される実数値関数である。最適化問題は，式(2)で与えられた制約条件のもとで，式(1)の目的関数を最適（上記の問題の場合は最大）にする決定変数の

値を決めることである。

制約条件 $x \in X$ を満たす x を実行可能解といい，

$$f(x^*) \geqq f(x), \ \forall x \in X \tag{3}$$

を満たす x^* を最適解という。$\varepsilon > 0$ を実数値とするとき，$x^* \in X$ の ε －近傍を次のように定義する。

$$N(x^*, \ \varepsilon) = \{x \in X : \|x - x^*\| < \varepsilon\} \tag{4}$$

ここで，$\|\cdot\|$ はユークリッドノルムである。この ε －近傍に対して

$$f(x^*) \geqq f(x), \ \forall x \in N(x^*, \ \varepsilon) \cap X \tag{5}$$

が成り立つとき，x^* を局所最適解という。この局所最適解と式(3)で与えられる最適解を区別するとき，式(3)で与えられる解を大域的最適解と呼ぶ。

式(1)，(2)で表現される最適化問題は，目的関数，制約条件，決定変数の種類によって様々な問題に分類することができる。決定変数が連続的な実数値をとるときの問題は連続的最適化問題となり，離散的な実数値をとる場合には離散的最適化問題あるいは組合せ最適化問題となる。連続的最適化問題において，目的関数と制約条件がともに線形関数で表されるとき，問題は線形計画問題となり，それ以外の場合が非線形計画問題である。線形計画問題はシンプレックス法などによって比較的容易に解くことができるが，非線形計画問題では，目的関数が2次関数で与えられるときの2次計画問題や目的関数が凸関数で制約条件が凸集合である凸計画問題以外の問題の最適解を求めることは一般的に容易ではない。このよう場合，最適解を求めるためには実行可能解の近傍における連続性に基づく逐次近似解法によらざるを得ず，また多峰性関数の場合には局所最適解が現われる可能性がある。離散的最適化問題で決定変数が整数値のみをとる場合，問題は整数計画問題となり，特に0と1の値のみをとる場合は0－1計画問題となる。この種の問題は，原理的には全数探索によって最適解を求めることができるが，決定変数の数が増えるとその組み合わせの数が爆発的に増加するので，現実的に最適解を求めることは不可能である。

一般的に最適解を得ることが困難な非線形計画問題や組合せ最適化問題では，現実的に満足できる近似解を少ない計算量で求めることができるアルゴリ

ズムを必要とする。アルゴリズムとは”与えられた問題を解くための有限の手続き”であり，計算量とは計算の各時点で保持しておかなければならないデータの最大個数である領域量と，計算手間である時間量を合わせたものである [21]。アルゴリズムの計算量とは，問題の規模Nを基準にして計算量がNのどのような関数になるかを評価するものである。この関数型には通常オーダー記法が採用され，$O(h(N))$で表現される。ここで問題の規模とは，問題を定義するために必要な入力データの長さ（語数）で，具体的には変数の個数と係数の大きさである。計算量の中で特に時間量に着目したとき，$h(N)$ は規模がNの問題を解くのに必要な基本演算（加減乗除，大小比較など）の実行回数である。あるアルゴリズムを用いて規模Nの問題を解こうとするとき，同じ規模の問題は多数存在すると考えられるので，それらの問題に関する$h(N)$ の上限を$H(N)(=\sup_N h(N))$ とすると，規模Nの問題はたかだか$H(N)$ 回の基本演算を実行すれば解くことができる。規模Nの問題に対して$H(N)=N^k$(kは任意の定数）であるとき，その計算量は多項式オーダーであるといい$O(N^k)$ で表す。$O(k^N)$，$O(N!)$ などは多項式オーダーより大きなオーダーで，Nの増加にともなって計算量は急激に増加する。計算量が多項式オーダーになるアルゴリズムを多項式時間アルゴリズムと呼び [21]，実用的なアルゴリズムであると評価する。多項式時間アルゴリズムが少なくとも1つ存在する問題をクラスPの問題と呼ぶ。クラスPの問題は実用的に解きうる問題であり，線形計画問題や2次計画問題はこのクラスPの問題に属する。

上記のような最適化問題に遺伝的アルゴリズムを適用するために，最適化問題の目的関数を適応度に，実行可能解を個体に，決定変数をn個の遺伝子座からなる染色体にそれぞれ対応させる。制約条件付き最適化問題の場合では，Lagrange 関数あるいはペナルティ関数を適応度に対応させる。決定変数はn個の文字列$a_1a_2\cdots a_n$からなり，各a_iが遺伝子でそのとりえる値が対立遺伝子である。これが決定変数の遺伝子型で，それに対応して定まる決定変数が表現型である。このように最適化問題の要素を遺伝的アルゴリズムのパラメータに対応づけることにより，遺伝的アルゴリズムを用いて最適化問題を解くこと

ができ，次のような利点がある。

1．解析解を得ることが困難な問題，目的関数が微分不可能な問題，非凸関数の問題などの最適化問題を解くことができる。
2．遺伝的アルゴリズムは確率的探索法の性質を持つので，大域的な探索空間を効率よく探索することができ，"よい近似解" を得ることができる。
3．遺伝的アルゴリズムは，複数の個体の集合を利用した多点探索を行う逐次近似アルゴリズムであるとみなすことができる。探索点が複数存在し，それらの点が互いに協調・競合することで局所最適解を回避することができるかもしれない。
4．関数の勾配を必要としないので，大域的探索性能に優れている。

遺伝的アルゴリズムによる最適化は局所最適解を回避できる可能性があるが，有限回の繰り返し（世代数）で最適解が得られる保証はなく，得られた解が最適である保証もない。さらにDe Jongが5番目にあげているテスト関数のように狭い領域に局所最適解が複数存在するような問題(GA-hardな問題)に対して遺伝的アルゴリズムによる最適化は困難である。

3．スケジューリング問題

3．1 オペレーション・スケジューリング

オペレーション・スケジューリングにおいてジョブが単一の生産設備の作業によって完成品になる場合，問題はジョブをいかなる順序で生産するかという順序付け問題に帰着する。この問題に対しては適当な優先規則（ディスパッチング規則）を用いることによって比較的容易に良好なスケジュールを得ることが出来る。生産が流れ作業型の多段階生産工程を経て行われるとき，ジョブの順序付け問題はフローショップ・スケジューリング問題となり，特別な場合を除き最適なスケジュールを得ることは困難である。2段階工程や特別な条件が成り立つ3段階工程の場合には総処理時間を最小にするスケジュールを与えるジョンソン・ルールが知られている［22］。工程が3段階以上のフローショッ

プ・スケジューリングやスケジューリング基準が総処理時間最小化以外では，このような簡便な方法はない。

このようなフローショップ・スケジューリングやスケジューリング基準が総処理時間最小化基準以外のスケジューリングで最適なジョブ順序を得る方法の１つに分岐限界法がある。この方法は順序付けの対象となるジョブの部分順序を生成して，その部分順序に関するスケジューリング基準を計算する限界手続きと，部分順序の中で最良の基準値をもつものを1つ選んで新しい部分順序を生成する分岐手続きからなる。分岐限界法では，問題の規模が大規模になると計算量が爆発的に増加するので，大規模なフローショップ・スケジューリングに対しては有効な方法ではない。このような問題に対して，より少ない計算量で近似解を得るヒューリスティック・アルゴリズムが開発されている（たとえば Petrov の方法［23]）。

多段階生産工程がフローショップ型でない場合，そのような生産工程はジョブショップ型と呼ばれ，そのスケジューリング問題はジョブショップ・スケジューリグ問題になる。ジョブ数が２，工程が複数，スケジューリング基準が総処理時間最小化の場合の問題では，最適なジョブ順序を得るための簡便な方法として図式解法が提案されている。ジョブ数が３以上の場合では，整数計画法や分岐限界法などを用いた方法が提案されている。さらに２段階工程の場合には，ジョンソン・ルールに基づくジャクソンの方法がある［24]。これらの方法は比較的小規模な問題であっても決定変数や部分順序の数が膨大となり，実際の問題への適用を困難にしている。

3．2　スケジューリング問題

スケジューリング問題は次の４つの要因によって規定される［25]。

1）ジョブ数

2）工程数

3）生産工程の形態

4）スケジューリング基準

1）のジョブ数は，すべてのジョブが同時到着である静的な問題では，順序付けの対象となるジョブの個数であり，記号nで表わす。ジョブが順次生産工程に到着する動的なスケジューリング問題の場合では，ジョブの到着時間間隔の確率分布を付記する。2）は生産工程を構成する工程の数で，記号mで表わす。3）の生産工程の形態は，生産工程内を移動するジョブのフロー・パターンであり，記号F，P，Gを用いる。ここでFはフローショップ型で，すべての工程におけるジョブの順序は同一である。Pはジョブの順序がすべての工程で同一である必要がない場合のフローショップ型の生産工程である。Gはジョブショップ型の場合を表わす。工程数が1（$m=1$）の場合は省略する。4）はスケジュールの評価基準で，以下のような基準がある［26］。

(a)　完了時刻に関する基準

ジョブiのすべての作業が終了して完成品になる時刻をそのジョブの完了時刻と呼び記号C_i，$i=1, 2, \cdots, n$で表わす。これは生産工程へのジョブの到着時間r_iとジョブのk番目の作業の加工時間p_{ik}，およびジョブiのk番目の作業の待ち時間W_{ik}，$k=1, 2, \cdots, K_i$，$i=1, 2, \cdots, m$により次のように定義できる。

$$C_i=r_i+\sum_{k=1}^{K_i}p_{ik}+\sum_{k=1}^{K_i}W_{ik} \tag{6}$$

ここでK_iはジョブiの全作業数である。

ジョブiの滞留時間とは，そのジョブが生産工程内に留まっている時間のことで，記号F_iで表わし，次式で定義できる。

$$F_i=\sum_{k=1}^{K_i}p_{ik}+\sum_{k=1}^{K_i}W_{ik}=C_i-r_i \tag{7}$$

完了時刻に関する評価基準には以下の基準がある。

(1)　総所要時間（make-span）：$C_{max}=\max\{C_1, C_2, \cdots, C_n\}$

(2)　平均完了時刻：$\overline{C}=\frac{1}{n}\Sigma_{i=1}^{n}C_i$

(3)　最大滞留時間：$F_{max}=\max\ \{F_1, F_1, \cdots, F_n\}$

(4)　平均滞留時間：$\overline{F}=\frac{1}{n}\Sigma_{i=1}^{n}F_i$

(5)　総滞留時間：$\sum F_i=\sum_{i=1}^{n}F_i$

(b)　納期に関する基準

ジョブ i に納期 d_i が定められているとき，完了時刻と納期との差を納期ずれ L_i といい，次式で表わす。

$$L_i=C_i-d_i \tag{8}$$

納期遅れ T_i は

$$T_i=\max\{L_i,\ 0\}=\max\{C_i-d_i,\ 0\} \tag{9}$$

で定義され，他方，納期余裕 E_i は

$$E_i=\max\{-L_i,\ 0\}=\max\{d_i-C_i,\ 0\} \tag{10}$$

で定義される。

納期に関する評価基準には以下の基準がある。

(6)　最大納期ずれ：$L_{max}=\max\{L_1,\ L_2,\ \cdots,\ L_n\}$

(7)　平均納期ずれ：$\overline{L}=\frac{1}{n}\sum_{i=1}^{n}L_i$

(8)　最大納期遅れ：$T_{max}=\max\{T_1,\ T_2,\ \cdots,\ T_n\}$

(9)　平均納期遅れ：$\overline{T}=\frac{1}{n}\sum_{i=1}^{n}T_i$

(10)　総納期遅れ：$\sum T_i=\sum_{i=1}^{n}T_i$

(11)　納期遅れジョブ数：$n_T=$ {$C_i-d_i<0$ となるジョブの数}

(c)　その他の基準

完了時刻や納期に関する評価基準は，個々のジョブに関するものであるが，生産工程全体の評価基準を考えることができ，その１つに生産工程内に存在するジョブの個数である仕事量（あるいは仕掛量）がある。時刻 τ における仕掛量は，加工待ちをしているジョブ数 $N_w(\tau)$ と加工中のジョブ数 $N_p(\tau)$，加工を完了したジョブ数 $N_c(\tau)$，まだ完了していないジョブ数 $N_u(\tau)$ とすると，

$$N_u(\tau)=N_w(\tau)+N_p(\tau) \tag{11}$$

$$N_u(0)=n \tag{12}$$

$$N_u(C_{max})=0 \tag{13}$$

したがって仕掛量に関する評価基準は次のようである。

⑿ 平均仕事数：$\overline{N_u}=\frac{1}{C_{max}}\int_0^{C_{max}}N_u(\tau)d\tau$

生産工程全体を対象とする評価基準に設備稼働率がある。設備稼働率Uは工程がジョブの加工に利用される時間と，その工程の利用可能時間の比で

$$U=\frac{\sum_{i=1}^{n}\sum_{k=1}^{K_i}p_{ik}}{F_{max}}$$

で定義される。
稼働率に関する評価基準は次のようである。

⒀ 平均稼働率：$\overline{U}=\frac{\sum_{i=1}^{n}\sum_{k=1}^{K_i}p_{ik}}{mF_{max}}$

個々のスケジューリング問題を表わすためにConway ら［27］に従って，前記の4つの要素を用いて次のように記述する。

ジョブ数／工程数／生産工程の形態／スケジューリング基準

たとえば，ジョンソン・ルールが適用されるnジョブ2機械フローショップ・スケジューリング問題は$n/2/F/C_{max}$と表わすことができる。

前節で最適化問題を多項式オーダー$O(N^k)$で最適解を求めることができるアルゴリズムをクラスPの問題と呼び，実用的なアルゴリズムであると評価することを述べた。他方，列挙法が適用でき，1つの列挙解の計算を多項式オーダーの時間でできる問題をクラスNPの問題という。組合せ最適化問題のほとんどがクラスNPの問題に属しており，クラスPの問題はクラスNPの問題に含まれる。クラスNPに属する任意の最適化問題Aが他の最適化問題Bに変換可能であり，その問題BがAに比べて難しい問題であるとき，問題BをNP-hard（NP困難）という。NP-hardな問題Bがさらにクラス NP の問題であれば，問題BをNP完全という。NP完全である最適化問題はきわめて難易度が高い問題で，多項式オーダーの計算時間で解くことが困難な問題である。スケジューリング問題では，$n/2/F/C_{max}$や$2/m/G/C_{max}$などの問題がクラスPの問題で，$n/1//\overline{C}$や$n/1//L$，$n/m/G/C_{max}$などの問題がNP完全であることが知られている[1)]。

3．3　遺伝的アルゴリズムによるスケジューリング

スケジューリング問題に対する遺伝的アルゴリズムの有効性を考察するために$n/m/G/C_{max}$のジョブショップ・スケジューリング問題を考える。このジョブショップ・スケジューリング問題を整数計画問題として定式化し最適なジョブ順序を得る試みが Bowman [28]，Wagner [29]，Manne [30] らによってなされている。ここでは変数や制約条件式の数が比較的少ない Manne の定式化を技術的順序が存在しない場合に拡張した問題を考える。

$n/m/G/C_{max}$のジョブショップ・スケジューリング問題を０－１計画問題として定式化するために以下の前提条件を設定する。

(1)　ジョブ数をn，工程数をmとし，ジョブと工程をそれぞれ記号J_i，$i=1,\cdots,n$，M_k，$k=1,\cdots,m$で表わす。

(2)　各ジョブはいずれの工程でも１度だけ加工されるものとし，工程M_kにおけるジョブJ_iの加工時間をp_{ik}，$i=1,\cdots,n$，$k=1,\cdots,m$で表わす。

(3)　各ジョブの技術的順序は存在しない。

(4)　各工程は複数のジョブを同時に加工することができない。

工程M_kにおけるJ_iとJ_jの加工開始時刻をそれぞれTS_{ik}，TS_{jk}とすると，前提条件(4)は

$$TS_{ik}-TS_{jk}\geqq p_{ik} \tag{15}$$

または

$$TS_{jk}-TS_{ik}\geqq p_{jk} \tag{16}$$

で表わすことができる。この離接制約条件（ないし二者択一制約条件）を０－１計画問題として定式化するために，次の０－１変数を導入する。

$$y_{ijk}=\begin{cases}1：工程M_kにおいてJ_iがJ_jに先行する場合\\0：そうでない場合\end{cases} \tag{17}$$

したがって，式(15)，(16)は次の２つの独立した制約条件に書き換えることができる。

1）R. Kan は［42］においてスケジューリング問題をクラスＰとクラス NP に分類している。

$$(M+p_{jk})y_{ijk}+(TS_{ik}-TS_{jk})\geq p_{jk}$$

$$(M+p_{ik})(1-y_{ijk})+(TS_{jk}-TS_{ik})\geq p_{ik}$$

ここでMは十分大きな正数である。

前提条件(3)によりジョブの技術的順序を考慮しないので，ジョブの加工順は工程によって異なる。工程M_{k_1}とM_{k_2}におけるジョブJ_iの加工開始時刻は次式のようになる。

$$TS_{ik_2}-TS_{ik_1}\geq p_{ik_2} \tag{18}$$

または

$$TS_{ik_1}-TS_{ik_2}\geq p_{ik_1} \tag{9}$$

これも離接制約条件であるので，式(15)，(16)の場合と同様に次の0－1変数を導入する。

$$z_{ik_1k_2}=\begin{cases}1 : J_i\text{が工程}M_{k_2}\text{に先行して}M_{k_1}\text{において加工される}\\ 0 : \text{そうでない場合}\end{cases} \tag{20}$$

これを用いて式(18)と式(19)を書き換える次のようになる。

$$(M_i+p_{ik_2})\ z_{ik_1k_2}+\ (TS_{ik_2}-TS_{ik_2})\ \geq p_{ik_2}$$

$$(M_i+p_{ik_1})(1-z_{ik_1k_2})+(TS_{ik_2}-TS_{ik_1})\ \geq p_{ik_1}$$

さらにC_{max}を最小にするためには，

$$TS_{ik}+p_{ik}\leqq C,\ i=1,\ \cdots n,\ k=1,\ \cdots,\ m$$

なる制約条件を加え，このCを最小にすればよい。

したがって，ジョブに技術的順序制約が存在しない場合の$n/m/G/C_{max}$を0－1計画問題として定式化すれば以下のようになる。

$$\text{Min. } C \tag{21}$$

$$\text{sub. to } TS_{ik}+p_{ik}\leqq C,$$

$$i=1,\ \cdots,\ n,k=1,\ \cdots,\ m \tag{22}$$

$$(M+p_{jk})\ y_{ijk}+(TS_{ik}-TS_{jk})\ \geq p_{jk}$$

$$i,\ j=1,\ \cdots n,\ k=1,\ \cdots,m \tag{23}$$

$$(M+p_{ik})(1-y_{ijk})+(TS_{jk}-TS_{ik})\ \geq p_{ik}$$

$$i,\ j=1,\ \cdots n,\ k=1,\ \cdots,m \tag{24}$$

$$(M_i+p_{ik_2})\ z_{ik_1k_2}+(TS_{ik_1}-TS_{ik_2})\ \geq p_{ik_2}$$
$$i=1,\cdots n,\ k_1,\ k_2=1,\cdots,m \tag{25}$$

$$(M_i+p_{ik_1})(1-z_{ik_1k_2})+(TS_{ik_2}-TS_{ik_1})\ \geq p_{ik_1}$$
$$i=1,\cdots n,\ k_1,\ k_2=1,\cdots,m \tag{26}$$

$$y_{ijk}\in\{0,1\},\ i=1,\cdots,n,\ k=1,\cdots,m \tag{27}$$

$$z_{ik_1k_2}\in\{0,1\},\ i=1,\cdots,n,\ k_1,\ k_2=1,\cdots,m \tag{28}$$

$$TS_{ik}\geq 0,\ i=1,\cdots,n,\ k=1,\cdots,m \tag{29}$$

この定式化より変数の数は y_{ijk} と $z_{ik_1k_2}$がそれぞれmn個，TS_{ik}が$m\frac{n(n-1)}{2}$個であり，制約条件式の数は，式(22)がmn個，式(23)と(24)がそれぞれ$m\frac{n(n-1)}{2}$個であり，式(25)と(26)が（$m-1$）n個である。$n=10$，$m=5$の場合では変数は285個，制約条件式は580個になる。

加工時間が表１で与えられている$5/2/G/C_{max}$問題を上記の０－１整数計画問題で定式化し，会話型数理計画法パッケージ LINDO を利用して解くと図１のガントチャートで表わされる$C_{max}=14$の最適スケジュールを得た。この最適スケジュールを得るために2732回の分岐と32037回のピボット演算を行った。各ジョブに技術的順序$M_1\rightarrow M_2$が存在する場合の$5/2/G/C_{max}$問題を Manne らの定式化にしたがって解くと分岐回数379回，ピボット演算回数7020回で$C_{max}=16$の最適解を得る。この場合では，技術的順序がジョブの組合せを制約しているので分岐回数とピボット演算回数を少なくしている。

このようにジョブショップ・スケジューリング問題ではジョブ数，工程数，制約条件の有無によってジョブの組合せ数が膨大となり現実的な問題に整数計画法を適用することを困難にしている。このようなジョブショップ・スケジューリング問題に遺伝的アルゴリズムを適用し，実用的で比較的良好なスケ

表１：$5/2/G/C_{max}$問題の加工時間（h）

ジョブ No.	1	2	3	4	5
工　程　1	3	4	2	1	4
工　程　2	1	5	2	3	3

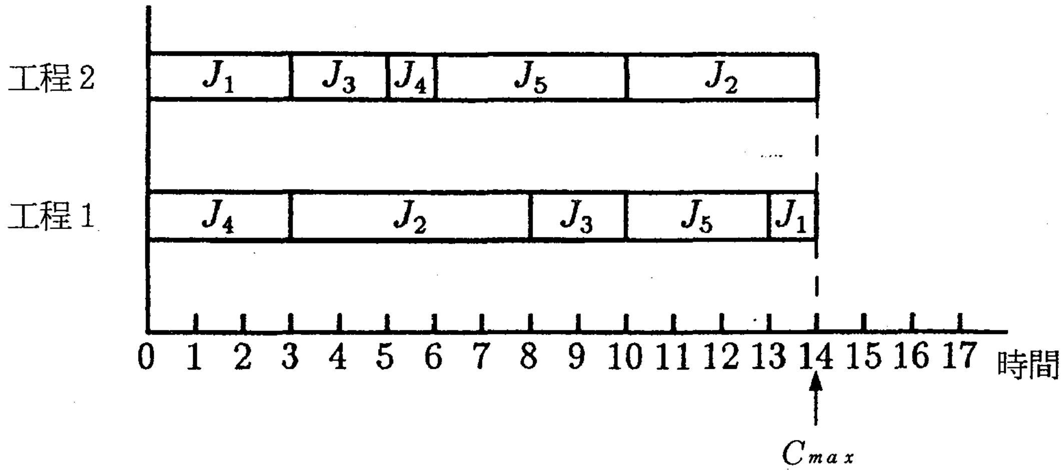

図1：0－1計画法による最適スケジュール

ジュールを得る方法が提案されている。Gupta ら［31］は $n/1//C_{max}$ を対象に，その総所要時間の変動を最小にする問題に遺伝的アルゴリズムを適用している。西川ら［32]，山田ら［33]，小林ら［34］は $n/m/G/C_{max}$ を対象としており，飯間ら［35]，三宮［36］は2工程と1台の搬送車からなるフローショップ問題の製品投入順序を求めている。

ジョブショップ・スケジューリング問題に遺伝的アルゴリズムを適用する際の重要な点はコーディング方法と交叉である。コーディング方法には，ガントチャートを用いて直接指定する方法［33]，選択グラフに基づく方法［32]，各工程の仕事列を指定する方法［16］などが提案されている。スケジューリング問題ではないが，巡回セールスマン問題において致死遺伝子を回避するためのコーディング法として，Grefenstette は順序表現を提案している［37]。これは，巡回する都市をアルファベット順に並べた標準リストを準備し，巡回する都市が標準リストの中で何番目であるかの番号を遺伝子とし，その都市のアルファベットを取り除いたリストを新しい標準リストとして，次に巡回する都市に対しも同様に行う方法である。この方法では，致死遺伝子が生成されることはないが，子の巡回路が親の巡回路と似ても似つかないものになる欠点を持つ。そのために順序表現を用いる遺伝的アルゴリズムでは，突然変異のみからなるランダムサーチ程度の性能しか持たないことが指摘されている［20]。

交叉には次の2つの方法を考えることができる。第1は，1組の対立遺伝子が同一の相同染色体上にあり，遺伝子が連鎖している場合で，一方の親の染色体1本と，他方の親の染色体を交叉させる方法である。この方法では，親の形質を受け継ぎやすいが，大きな形質の変化は突然変異に頼らざるを得ない。第2には，遺伝子が連鎖していない場合で，それぞれ同じ形質を持つ染色体同士が同一の交叉点を持ち，それぞれ交叉させる方法である。この方法では，親の優良な形質が破壊されやすくなるため building block を保存させるような工夫が必要であるが，大きな形質の変化を期待することができる。

先の$5/2/G/C_{max}$問題に遺伝的アルゴリズムを適用するために，上記のことを考慮して以下の前提条件をおく。

(1) 表現型をH，遺伝子型をX，工程M_1をm_1，工程M_2をm_2とする。

(2) 適合関数を加工時間，適合値をf，平均適合値をFとする。

(3) 個体数$n=2$，遺伝子型の長さ5とする。

(4) コーディング方法には先述の Grefenstette の方法を用い，各工程に1本の染色体を持つ2倍体を考える。2倍体の場合，値の大きい染色体の適合値が発現型の適合値となるので，その染色体を優性，他方を劣性と考える。

(5) 初期スケジュールを次のように設定する。

$H(X_1(m_1))=J_1-J_3-J_4-J_5-J_2$

$H(X_1(m_2))=J_4-J_2-J_5-J_3-J_1$

$H(X_2(m_1))=J_3-J_2-J_4-J_5-J_1$

$H(X_2(m_2))=J_2-J_5-J_1-J_3-J_4$

(6) 交叉は1点交叉とする。

(7) 突然変異率は無視する。

(8) 同一ジョブがM_1とM_2で同時に開始できるとき，SPT 規則に基づいて順序付ける。

これらの前提条件に基づいて遺伝的アルゴリズムの操作を行なったところ，表2と表3のような結果を得た。表2は，遺伝的アルゴリズムを適用することによって得たスケジュールで，第3世代以降は省略している。表3より，平均

適合値は第15世代で収束しており，最適スケジュールは，第2世代と第7世代で得ている。この最適スケジュールのガントチャートを図2に示す。このようにジョブショップ・スケジューリング問題に遺伝的アルゴリズムを適用する

表2：遺伝的アルゴリズムによる5／2／G／C_{max}問題のスケジューリング

世代	遺伝子型	工程	親1	親2	交叉点	コード	スケジュール	適合値
0	X_1	M_1				12221	J_1-J_3-J_4-J_5-J_2	14
		M_2				24111	J_3-J_2-J_4-J_5-J_1	14
	X_2	M_1				42321	J_4-J_2-J_5-J_3-J_1	14
		M_2				32221	J_3-J_2-J_4-J_5-J_1	17
								平均適合値＝15.5
1	X_1	M_1	$X_1(m_1)$	$X_2(m_1)$	2	12321	J_1-J_3-J_5-J_4-J_2	15
		M_2	$X_1(m_2)$	$X_2(m_2)$	1	22221	J_2-J_3-J_4-J_5-J_1	14
	X_2	M_1	$X_1(m_1)$	$X_2(m_1)$	2	42221	J_4-J_2-J_3-J_5-J_1	14
		M_2	$X_1(m_2)$	$X_2(m_2)$	1	34111	J_3-J_5-J_1-J_2-J_4	14
								平均適合値＝14.5
2	X_1	M_1	$X_1(m_1)$	$X_2(m_1)$	1	12221	J_1-J_3-J_4-J_5-J_2	14
		M_2	$X_1(m_2)$	$X_2(m_2)$	3	22211	J_2-J_3-J_4-J_1-J_5	14
	X_2	M_1	$X_1(m_1)$	$X_2(m_1)$	1	42321	J_4-J_2-J_5-J_3-J_1	14
		M_2	$X_1(m_2)$	$X_2(m_2)$	3	34121	J_3-J_5-J_1-J_4-J_2	14
								平均適合値＝14.0
3	X_1	M_1	$X_1(m_1)$	$X_2(m_1)$	3	12221	J_1-J_3-J_4-J_5-J_2	14
		M_2	$X_1(m_2)$	$X_2(m_2)$	2	22121	J_2-J_3-J_1-J_5-J_4	16
	X_2	M_1	$X_1(m_1)$	$X_2(m_1)$	3	42321	J_4-J_2-J_5-J_3-J_1	14
		M_2	$X_1(m_2)$	$X_2(m_2)$	2	34211	J_3-J_5-J_2-J_1-J_4	14
								平均適合値＝15.0

表3：各世代の適合値と平均適合値

世代	優性染色体の適合値		平 均 適合値	世代	優性染色体の適合値		平 均 適合値
	X_1	X_2			X_1	X_2	
0	14	17	15.5	11	16	17	16.5
1	15	14	14.5	12	18	17	17.5
2	14	14	14.0	13	18	17	17.5
3	16	14	15.0	14	16	17	16.5
4	18	17	17.5	15	15	14	14.5
5	18	17	17.5	16	15	14	14.5
6	15	14	14.5	17	15	14	14.5
7	14	14	14.0	18	15	14	14.5
9	15	14	14.5	19	15	14	14.5
10	17	18	17.5	20	14	15	14.5

と，さきの０－１計画問題による最適化に比べて少ない手続で良好なスケジュールを得ることができる。

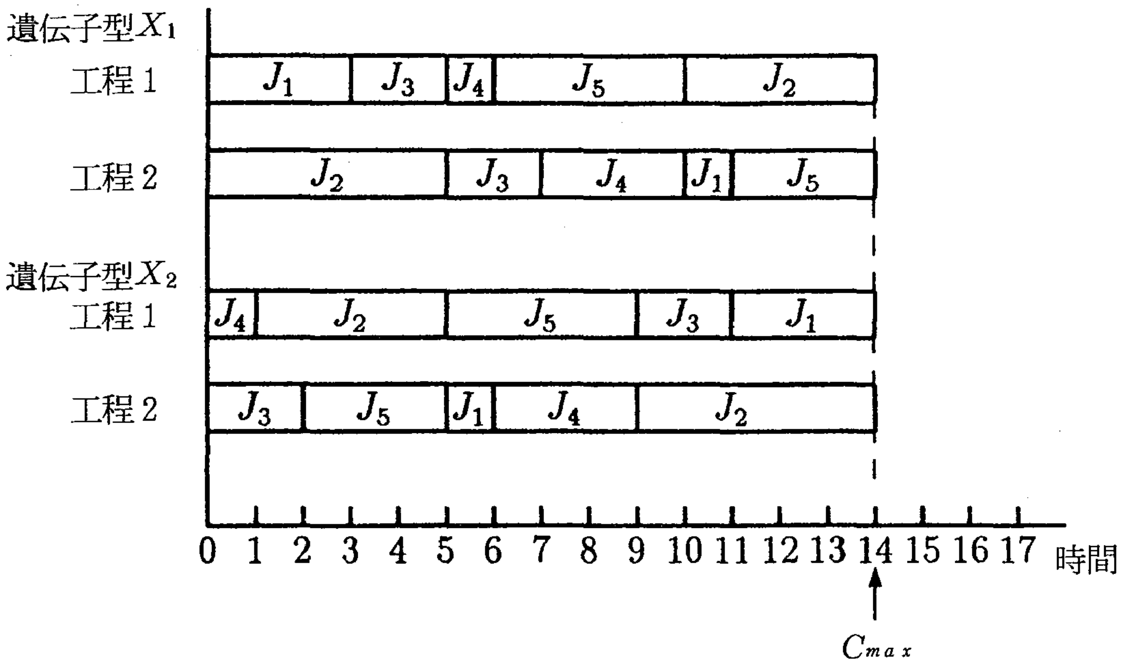

図２：遺伝的アルゴリズムによる最適スケジュール

次にNP完全であるスケジューリング問題の例として，ジョブの処理順序に依存する段取り時間が存在する場合の$n/1//C_{max}$を取り上げ，遺伝的アルゴリズムを用いてシミュレーションを行う。この問題は，$n+1$の都市を巡回する巡回セールスマン問題として取り扱うことができるが，スケジューリングの場合では，最初に出発したの都市（ジョブ）に戻る必要がないのでn都市間の巡回経路を求める問題として考えればよい。

この問題のシミュレーションを実施するために，以下の前提条件を設定する。

(1)　最大個体数を50，最大遺伝子数を30，最大世代数を25とする。

(2)　コーディング法は先述のGrefensttete の方法を用い，ジョブスケジュールを発現型とする。

(3)　適合関数は加工処理時間に段取り時間を加えたものを用いる。

(4)　遺伝子の複製はランダム，交叉は１点交叉，交叉率は0.99，突然変異は

起こらないものとする。

(5)　各ジョブの加工時間と段取り時間はそれぞれ表4と表5のようである。

ジョブ数を10，スケジュール数を10としてシミュレーションを実施した結果を表6（第5世代以降は省略している）に，また世代にともなう適合値の変

表4：n／1／／C_{max} 問題の加工時間（h）

ジョブNo.	1	2	3	4	5	6	7	8	9	10
加工時間	20	25	40	32	50	21	19	27	34	41

表5：n／1／／C_{max} 問題の段取り時間（h）

from＼to	1	2	3	4	5	6	7	8	9	10
1	－	31	2	44	12	18	52	48	37	2
2	5	－	19	29	24	28	62	3	0	29
3	8	46	－	0	37	10	42	0	36	46
4	41	27	0	－	19	5	41	20	23	45
5	49	2	19	31	－	0	91	4	7	23
6	8	13	33	8	45	－	7	21	43	9
7	0	21	7	30	15	32	－	0	21	13
8	6	6	20	29	17	29	11	－	17	1
9	10	2	0	29	12	7	48	23	－	24
10	19	45	51	40	32	34	34	0	44	－

表6：シミュレーション結果

第0世代（初期スケジュール）

No.	親1	親2	交叉点	スケジュール										適合値
1				1	2	3	4	5	6	7	8	9	10	426.00
2				3	1	5	2	4	9	8	7	10	6	464.00
3				9	3	6	1	5	8	2	7	4	10	486.00
4				10	6	1	4	2	9	3	5	8	7	474.00
5				3	8	9	6	10	5	1	7	4	2	532.00
6				8	7	1	3	2	10	4	6	5	9	494.00
7				1	2	6	9	5	7	8	10	3	4	566.00
8				5	6	8	7	10	3	4	9	2	1	435.00
9				6	9	2	8	4	10	5	1	7	3	571.00
10				7	4	10	5	3	2	9	6	1	8	544.00

最小適合値：426.00，最大適合値：571.00，平均適合値：499.20

表6：シミュレーション結果（つづき）

第1世代

No.	親1	親2	交叉点	スケジュール										適合値
1	10	4	5	7	4	10	5	3	9	1	2	8	6	544.00
2	10	4	5	10	6	1	4	2	5	9	7	3	8	508.00
3	8	6	1	5	8	1	3	2	10	4	7	6	9	522.00
4	8	6	1	8	5	7	6	10	3	4	9	2	1	539.00
5	3	8	8	9	3	6	1	5	8	2	7	10	4	464.00
6	3	8	8	5	6	8	7	10	3	4	9	1	2	469.00
7	8	5	8	5	6	8	7	10	3	4	9	2	1	435.00
8	8	5	8	3	8	9	6	10	5	1	7	4	2	532.00
9	9	5	9	6	9	2	8	4	10	5	1	7	3	571.00
10	9	5	9	3	8	9	6	10	5	1	7	4	2	532.00

最小適合値：435.00，最大適合値：571.00，平均適合値：514.60，交叉回数：5

第2世代

No.	親1	親2	交叉点	スケジュール										適合値
1	8	5	7	3	8	9	6	10	5	1	4	7	2	529.00
2	8	5	7	9	3	6	1	5	8	2	10	7	4	422.00
3	4	6	4	8	5	7	6	10	3	4	9	1	2	573.00
4	4	6	4	5	6	8	7	10	3	4	9	2	1	435.00
5	2	2	10	10	6	1	4	2	5	9	7	3	8	508.00
6	2	2	10	10	6	1	4	2	5	9	7	3	8	508.00
7	9	7	8	6	9	2	8	4	10	5	1	7	3	571.00
8	9	7	8	5	6	8	7	10	3	4	9	2	1	435.00
9	7	3	5	5	6	8	7	10	9	1	3	2	4	485.00
10	7	3	5	5	8	1	3	3	7	9	10	6	4	516.00

最小適合値：435.00，最大適合値：573.00，平均適合値：500.20，交叉回数：9

第3世代

No.	親1	親2	交叉点	スケジュール										適合値
1	4	4	3	5	6	8	7	10	3	4	9	2	1	435.00
2	4	4	3	5	6	8	7	10	3	4	9	2	1	435.00
3	4	8	1	5	6	8	7	10	3	4	9	2	1	435.00
4	4	8	1	5	6	8	7	10	3	4	9	2	1	435.00
5	6	10	7	10	6	1	4	2	5	9	8	7	3	494.00
6	6	10	7	5	8	1	3	2	7	9	6	4	10	510.00
7	7	1	2	6	9	8	5	10	4	1	3	7	2	561.00
8	7	1	2	3	8	2	9	5	10	6	1	7	4	474.00
9	5	5	6	10	6	1	4	2	5	9	7	3	8	508.00
10	5	5	6	10	6	1	4	2	5	9	7	3	8	508.00

最小適合値：435.00，最大適合値：561.00，平均適合値：479.50，交叉回数：14

表6：シミュレーション結果（つづき）

第4世代

No.	親1	親2	交叉点	ス	ケ	ジ	ュ	ー	ル					適合値
1	7	3	1	6	5	8	7	10	3	4	9	2	1	463.00
2	7	3	1	5	9	8	6	10	4	1	3	7	2	523.00
3	3	4	1	5	6	8	7	10	3	4	9	2	1	435.00
4	3	4	1	5	6	8	7	10	3	4	9	2	1	435.00
5	10	6	7	10	6	1	4	2	5	9	7	3	8	508.00
6	10	6	7	5	8	1	3	2	7	9	6	4	10	510.00
7	4	10	2	5	6	1	4	2	7	10	8	3	9	519.00
8	4	10	2	10	6	7	5	9	3	4	8	2	1	403.00
9	2	9	6	5	6	8	7	10	3	9	2	1	4	492.00
10	2	9	6	10	6	1	4	4	5	8	9	7	3	522.00

最小適合値：403.00，最大適合値：523.00，平均適合値：481.00，交叉回数：19

化を表7に示す。この結果を見ると適合値の最小値は，第4世代の第8番目のスケジュールで総所要時間C_{max}は403時間であった。また適合値は12，13世代で収束しているがそれ以後の世代では遺伝子は多様性を失い単調なスケジュールの繰り返しになっている。これは複製をランダムに行ったためで，期待値の高い個体を次世代に残すという保証がないからである。ルーレット方式による複製は，適合値に幅があるため期待値に差（スロットの差）ができ，次世代の親を選択して，そのスケジュールを決定すると，その段階で極小値を持つ親が選択される頻度が高くなる。そのために，その世代ですでに多様性が失われ探索範囲を狭める結果となるからである。

表7：適合値の変化

世代	平均値	最小値	最大値
0	499.20	426.00	571.00
1	514.00	435.00	571.00
2	500.20	435.00	573.00
3	479.50	435.00	561.00
4	481.00	403.00	523.00
5	493.20	435.00	519.00
6	488.70	446.00	515.00
7	495.10	456.00	515.00
8	505.80	483.00	562.00
9	507.40	485.00	562.00
10	502.70	485.00	562.00
11	500.20	485.00	562.00
12	490.10	465.00	510.00
13	490.10	465.00	510.00
14	490.00	465.00	510.00
15	491.10	465.00	510.00
16	497.50	465.00	510.00
17	506.10	485.00	510.00
18	506.00	485.00	510.00
19	507.30	465.00	538.00
20	514.40	508.00	538.00
21	514.40	508.00	538.00
22	511.40	508.00	538.00
23	508.40	508.00	510.00
24	508.20	508.00	510.00
25	508.20	508.00	510.00

4．結　論

本論文ではスケジューリング問題の最適化法として，遺伝的アルゴリズムを取り上げ，その有効性を数値計算例で示した。スケジューリング問題のような組合せ最適化問題に対して遺伝的アルゴリズムが有効であるのは，この手法が確率的探索や学習の概念に基づいているので，大域的な探索空間を効率よく探索して解を見つけだすことができるからである。このような組合せ最適化問題に対する遺伝的アルゴリズムの有効性に着目して，レイアウト計画［38］や工程計画問題［39］などの分野にも応用研究がなされている。

遺伝的アルゴリズムはこのように様々な分野で応用研究がなされているのであるが，すでに指摘されているようにいくつかの問題点がある。その1つは，解を染色体上でいかに表現するかというコーディング／デコーディングの問題および交叉の問題である。これらは研究者が職人芸的に決定しており，論理的な方法が示されていない。評価基準についても同様で，適合関数としてどの評価基準を選択するかによって解の収束速度や近似度に影響を与えるのであるが，その適切な選択基準がなく，これも職人芸的に設定されているのが現状である。また遺伝的アルゴリズムは大域的最適化に有効であるとされているが，他の大域的最適化手法［40］と比較し，その有効性を検討する必要がある。

遺伝的アルゴリズムのスケジューリング問題への適用を試みた研究の多くは，組合せ最適化問題の例としてジョブショップ・スケジューリングを取り扱っており，ジョブショップ・スケジューリングそのものの議論が希薄である。とくに，その多くはスケジューリング問題の評価基準としてC_{max}を採用しており，それ以外の評価基準を扱っているものは希である。また NP 完全なスケジューリング問題に遺伝的アルゴリズムを適用することによって多項式オーダーで近似解を得る保証はなく，その有効性を検証する必要がある。

先述したように遺伝的アルゴリズムは様々な分野に応用研究がなされているが，遺伝的アルゴリズムを適用するために非現実的な前提条件を設定したり，逆に現実的な条件を考慮していない研究も見受けられる。土居が［41］で指摘

しているように，このような応用研究の結果，矮小なモデルのみが残りモデルとしてほとんど発展していないという状況にならないようにしなければならない。

参考文献

[1] 人見勝人（監修）：CIM 総論－コンピュータによる設計・生産・管理 第2版，共立出版，p.173，(1993).

[2] 茨木俊秀：組合せ最適化－分枝限定法を中心として，産業図書，(1983).

[3] 鍋島一郎：スケジューリング理論，森北出版，(1974).

[4] M.S. Fox : *Constraint - Directed Search－A case study of job－shop scheduling*, Pitman, (1987).

[5] D.E. Goldberg : *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison－Wesley, (1989).

[6] J.H. Holland : *Adaptation in Natural and Artificial Systems 2nd ed.*, Univ. of Michigan Pr., (1992).

[7] 北野宏明（編）：遺伝的アルゴリズム，産業図書，(1993).

[8] 安居院猛，長尾智晴：ジェネティックアルゴリズム，昭晃堂，(1993).

[9] 米澤保雄：遺伝的アルゴリズム，森北出版，(1993).

[10] 特集 遺伝的アルゴリズムと免疫システム論，数理科学，No. 353，(1992).

[11] 特集 遺伝的アルゴリズム，計測と制御，Vol.32，No. 1，(1993).

[12] 遺伝アルゴリズム特集号，システム／制御／情報，Vol.37，No. 8，(1993).

[13] 小林重信：遺伝的アルゴリズムの基礎と応用 I，II，III，IV，オペレーションズ・リサーチ，Vol.38，No. 5，pp.256－261，No. 6，pp.311－319，No. 7，pp.352－357，No. 8，pp.419－429，(1993).

[14] 和田健之介：進化システム論 遺伝的アルゴリズムの基礎 (1)，(2)，(3)，(4)，(5)，(6)，(7)，Computer Today，No. 47，No. 48，No. 49，No. 50，No. 51，No. 52，(1992)，No. 53，No. 54，(1993).

[15] 特集 遺伝的アルゴリズム，オペレーヨンズ・リサーチ，Vol.38，No. 7，pp.328－351，(1993).

[16] 小林重信：遺伝的アルゴリズムの基礎と応用 I，オペレーションズ・リサーチ，Vol.38，No. 5，pp.256－261，(1993).

[17] 西川禕一：遺伝的アルゴリズムとその工学的意味，システム／制御／情報，Vol.37，No.8，pp.445－449，(1993).

[18] [5]，pp.28－33.

[19] [5], p.33.

[20] 山村雅幸, 小野貴久, 小林重信：形質の遺伝を重視した遺伝的アルゴリズムに基づく巡回セールスマン問題の解法, 人工知能学会誌, Vol. 7, No. 6, pp.1049−1059, (1992).

[21] 茨木俊秀：離散最適化法とアルゴリズム, 岩波講座応用数学, 岩波書店, (1993).

[22] [3], pp.50−84.

[23] V.A.Petrov : *Flowline Group Production Planning*, Business Publication, (1968).

[24] [3], pp.189−198.

[25] R.Conway, W.Maxwell, L.W.Miller : *Theory of Scheduling*, Addison−Wesley, (1967), 関根智明（監訳）：スケジューリングの理論, 日刊工業新聞社, (1971), pp. 9−11.

[26] S.French : *Sequencing and Scheduling : An introduction to the mathematics of the Job−Shop*, Ellos Horwood, (1982), pp.10−14.

[27] [25], p.10.

[28] E.H.Bowman : The Schedule−sequencing problem, *Operations Research*, Vol. 7, No. 5, (1959), pp.621−624.

[29] H.M.Wagner : An integer programming model for machine scheduling, *Naval Research Logistics Quarterly*, Vol. 6, No. 2, (1959), pp.131−140.

[30] A.S.Manne : On the job shop scheduling problem, *Operations Research*, Vol. 8, No. 3, (1960), pp.219−223.

[31] M.C.Gupta, Y.P.Gupta, A.Kumar : Minimizing flow time varianc in a single machine system using genetic algorithms, *European J. of Operational Research*, Vol.70, No. 3, (1993), pp.289−303.

[32] 西川禕一, 玉置久：GA のスケジューリング問題への応用, 講習会教材 [No. 930−43], 日本機械学会, (1993), pp.39−46.

[33] 山田武士, 中野良平：遺伝アルゴリズムとスケジューリング問題, システム／制御／情報, Vol.37, No. 8, (1993), pp.484−489.

[34] 小林重信, 小野功, 山村雅幸：GA によるジョブショップスケジューリング問題の解法, 第3回インテリジェント・シンポジュウム [No.930−60], 日本機械学会, (1993), pp.227−232.

[35] 三宮信夫：遺伝的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, (1993), pp.437−445.

[36] 三宮信夫：遺伝アルゴリズムのスケジューリング問題への応用, セミナー「遺伝アルゴリズム／ニューラルネット／ファジの新しい展開を探る」, システム制御情報学会, (1993), pp.33−42.

[37] 和田健之介：遺伝的アルゴリズムと機械の進化，数理科学，No. 328，(1990)，pp. 47－51.

[38] K. Y. Tam： Genetic algorithms, function optimization, and facility layout design, *European J. of Operational Research*, Vol. 63, No. 3, (1992), pp. 322－346.

[39] 皆川雅章，嘉数侑昇：GA によるプロセスプランニング問題へのアプローチ，第3回インテリジェント・シンポジュウム[No. 930－60]，日本機械学会，(1993)，pp. 233－238.

[40] たとえばH. Tuy： *Global Optimization*, Springer－Verlag, (1990).

[41] 土居洋文：個体発生と遺伝的アルゴリズム，数理科学，No. 353，(1992)，pp. 8－11.

[42] R. Kan： *Machine Scheduling Problems*, Martinus Nijhoff, (1976), pp. 131－135.