

多目的プロジェクトスケジューリング問題 のためのGA構成法に関する研究

徳島大学

滑川 達

徳島大学大学院

○ 吉田 健

By Susumu Namerikawa, Ken Yoshida

本研究では、プロジェクトスケジューリング問題のための遺伝的アルゴリズム (GA) の新たな構成方法を提案する。ここでは、GA による解探索の範囲を実行可能解集合の中でも特に局所最適解集合のみに限定させることにより、多目的関数が設定された問題に対しても高精度の解を効率的に求められるスケジューリングアルゴリズムを構築した。さらに、例題ネットワークを用いて、既存の GA 構成法や山崩し法との比較計算を行った。

キーワード：遺伝的アルゴリズム、プロジェクトスケジューリング、工程・工期

1. はじめに

建設工事のスケジュール作成において、工事用資源の増減を無計画に行うことは、工事遂行の実行可能性を低下させるとともに、コスト面でも高価になる場合が多い。また、労働力不足の問題や工期が非常に厳しい与件として存在することを考えると、工事用資源の投入量に対する量的制限を考慮したプロジェクトスケジューリング問題の高度な解析が必要となる。ここで、プロジェクトスケジューリング問題とは、プロジェクトの遂行に必要な各作業の所要時間や必要資源、及び作業間の順序関係が与えられたとき、工期制約・資源制約下で、工期の短縮化や資源配分の平滑化などの複数の計画目標を達成するように、各作業の開始時刻を求める問題と定義される。以上の問題は、一種の組合せ最適化問題として定式化され、これまでも山崩し法などを代表に多くの近似解法の研究がなされている。しかし、従来法である山崩し法をはじめとする優先順位法においては、トータルフロートのような PERT 計算によって求められるスケジュール指標に基づいて非常に限定的な形で資源制約に対する超過分を先送りする形で解消させていくため、最適解に対して大きく精度の劣るスケジュールしか求められない場合が少なくない。

そこで最近の傾向としては、多種多様な制約条件や計画目標の問題に対応可能で、かつある程度高精度な解を現実的な時間で計算することを指向した遺伝的アルゴリズム (GA) の適用が検討されている。GA をプロジェクトスケジューリング問題に適用し、効果的な探索過程を実現するためには、遺伝子型から表現型への変換が次のように行われていることが重要である。すなわち、a) GA による探索空間内の個体から問題空間内の解候補への変換 (対応) が“1対1”あるいは“多対1”になるように、問題の定式化段階における決定変数ならびに個体表現としての探索空間を定める、さらにb) 表現型への変換部分に、すべての制約条件を満足させるとともに、“スケジュールの代替案として検討に値する”ある程度良い解のみが求められる手続きを組込む、というアプローチが効果的である。

本研究においては、以上の事項を満足し得る①問題の定式化、②個体表現、③表現型への変換方法の④GAオペレーター⑤アルゴリズムの性能評価の検討を通して、プロジェクトスケジューリング問題のためのGAの新たな構成方法を提案する。

2. 既存研究

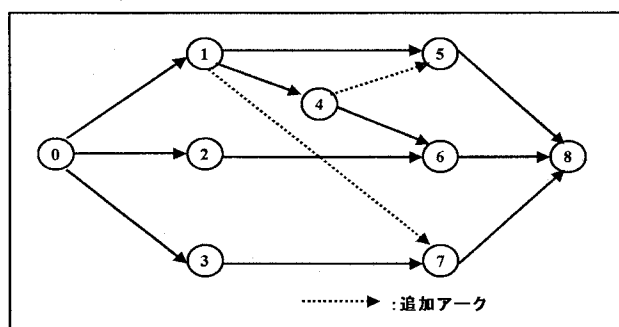
プロジェクトスケジューリング問題に対する最も一般的な手法としては、「PERT/MANPOWER」¹⁾における山崩し計算があげられる。この手法では、始めに工事用資源制約を考慮しない形で PERT 計算を行い各種結合点時刻を求める。その後、トータルフロート

*工学部建設工学科 TEL (088) 656-9877

**大学院工学研究科 TEL (088) 656-7578

などによる優先順位規則にもとづき資源制約を侵すことがないように山崩し計算を実行しスケジュールを決定するものである。このような方法では、優先順位規則という非常に限定された形で作業間のコンフリクトを解消していくため、求められるスケジュールの解精度が高くないことや作成される代替案の種類も限られてくることが挙げられる。

一方、春名、滑川²⁾は「カットネットワーク手法」にもとづく最適解法を提案している。この手法では、工程ネットワークの作業間順序関係を同時着手可能な並列作業群として求められるカット間の順序関係に等価変換したカットネットワークの作成方法を明らかにするとともに、このカットネットワークの最適経路を求めることで、理論的には最適解を求められることを示した。しかし、目的関数としては工期の最小化のみ対象としているとともに、ネットワークの規模が大きくなるとカットネットワークの作成に多くの時間が必要となる。

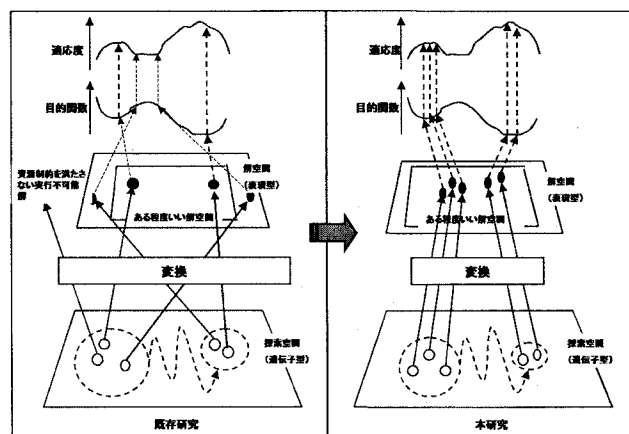


図－１ 既存研究における追加アーク例

このような中、解精度の向上と計算コストの両立をめざしたアプローチとして、メタヒューリスティクスを適用した研究が盛んになっている。例えば、日高ら³⁾は、ニューラルネットワーク理論の適用を試みているが、そこでの計算例としてはコンフリクトを起こす作業が２つの場合のみの解消方法しか示されていない。奥谷ら^{4) 5)}は、タブー探索法やGAの適用によりプロジェクトスケジューリング問題を解く方法を提案している。奥谷らによるこれらの研究では図－１のように追加アークを加えることによって、山積み図を変化させながら解探索を進める方法を採用しているとともに、納期遅れ、あるいは閉路が生じてしまう問題についても対策法を提案し、この問題を解決している。しかしながら遺伝子型から表現型へ変換した際、図－１のようにアークを追加したあとのスケジュール計算に

PERT 計算をもちいているために資源制約を満たさない実行不可能解が生成されたり、実行可能解の中でも検討に値しないような非効率な解の生成可能性が増大することで解探索に悪影響を及ぼすケースがあるものとする。

本研究においては、図－２に示したように、上述した既存のGA適用方法における構成上の冗長性を解決し、特に実行可能解集合のなかでもある程度良い解としての局所最適解のみの解探索に限定した効率的なプロジェクトスケジューリング問題のための新たなGA構成方法を開発する。



図－２ 本研究におけるGA達成イメージ図

3. 問題の定式化

ここでは、対象とするプロジェクトネットワークの形式として、アロー・ダイアグラム型のネットワークを考える。また問題の入力データとして、作業*i*、作業の所要日数*d_i*、作業の必要資源数*w_i*、資源制約*W*を与える。作業数*m*をもつプロジェクトスケジューリングは、作業間の順序関係からみても資源の投入量制約からみても「開始可能」となった時刻にすぐさま作業を開始するものと仮定することにより、*m*×*m*のマトリックス（以下スケジュールマトリックス）として表すことができる。すなわち、実行可能なバーチャート工程表は、以下のような制約条件を満たしていなければならない。

$$\sum_{k=1}^m a_{ik} x_k = d_i \quad \text{for all } i \quad (3.1)$$

$$\sum_{i=1}^m a_{ik} w_i \leq W \quad \text{for all } k \quad (3.2)$$

$$\text{if } a_{ik} = 1 \cap a_{rk} = 1 \\ \text{then } R_{ir} = 0 \quad \text{for all } k \quad (3.3)$$

$$x_k \geq 0 \text{ for all } k \quad (3.4)$$

ここで、 a_{ik} ；スケジュールマトリックスの k 列における作業 i の実施状態、(1or0)、 x_k ； k 列の区間長で式(3.1)の連立方程式を解くことによって求められる、 R_{ii} ；工程ネットワークの可達行列の要素である。また、今回取り扱う計画目標となる目的関数は以下のようなものである。

(1)工期 T_d の最小化：プロジェクトの早期完了を目的とする。

$$\min T_d = \sum_k x_k$$

(2)最大投入資源量 W の最小化：プロジェクトコストへの負担を軽減することを目的として、最小化を図る。

$$\min W = \max_k \sum_i a_{ik} \cdot w_i$$

(3)投入資源の平滑度 H の最小化：投入資源数をできるだけ一定にして、遊休化を防ぐことを目的とする。

$$\min H = \sum_k \left\{ \sum_i (a_{ik} \cdot w_i) \right\}^2 \cdot x_k$$

(4)資源使用効率 E の最大化：必要資源の稼働率を向上させることを目的に、最大化を図る。

$$\max E = \sum_i w_i \cdot d_i / T_d W$$

またこれらの計画目的を次のように標準化し、適合度を計算することとした。

(1)式～(4)式に対応する適合度を次のように表す。

$$f_i = (X_{\max} - X) / (X_{\max} - X_{\min}) \quad (i=1,2,3)$$

(4)式の E は既に0と1の間で変動しているため、そのまま適合度関数となる。

$$f_4 = E$$

さらに、次のようにして総合的な適合度関数を表す。

$$F = \alpha_1 f_1 + \alpha_2 f_2 + \alpha_3 f_3 + \alpha_4 f_4 \quad \sum_{i=1}^4 \alpha_i = 1$$

また、 α_i については、施工計画者の当該工事の特性を考慮し設定するものとする。なお、以上の定式化は、基本的に奥谷⁵⁾らの既存研究に準拠しており、これを本研究が採用しているスケジュールマトリックスの形式に等価変換したものである。

4. 個体表現

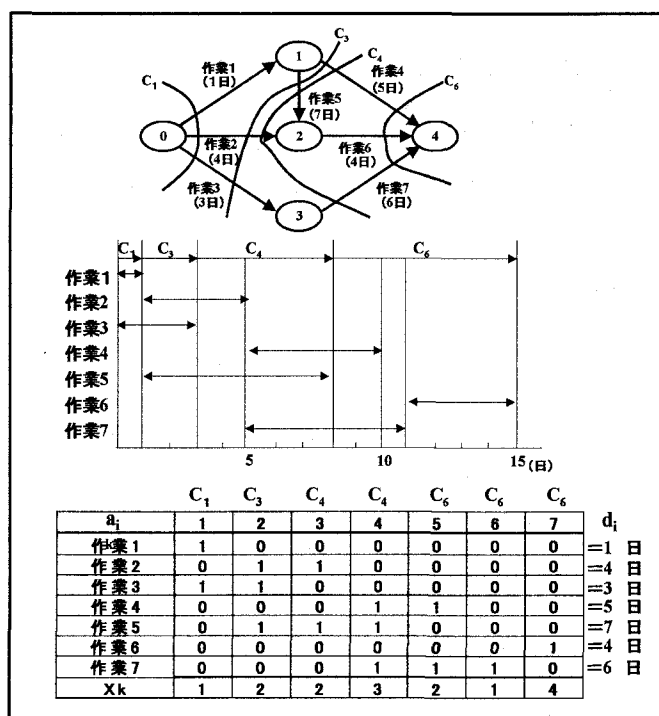


図-3 カットパスとスケジュールの関係

まず、プロジェクトスケジューリング問題にGAを適用する場合に必要なことは生物集団の要素である個体に対応するものをどのように表現するかということである。ここでは、前述したカットネットワーク手法におけるカットパスに着目し、実行可能なスケジュールマトリックスとの関係を考察する。まず、図-3に示したように、実行可能なスケジュールマトリックスはカットパスの遷移過程として捉えることが可能である。換言すれば、任意のカットパスのカット間順序関係に従ってスケジュールマトリックスの各列の実施状態を決定していくことにより、式(3.3)のような作業間順序関係に関する制約を満たすスケジュールが必ず求められるとともに、カットパスを変化させることにより、様々な代替案を作成することが可能となる。このため本研究では、このカットパスを個体として採用する。

5. 表現型への変換方法

ここでの表現型への変換とは、個体としてのカットパスにもとづいて、代替案としてのスケジュールマトリックスを求めることを意味する。すなわち、まず生成されたカットパスにおいて、a) 現レベルから次レベルへのカットの遷移を決定している作業を順次各列

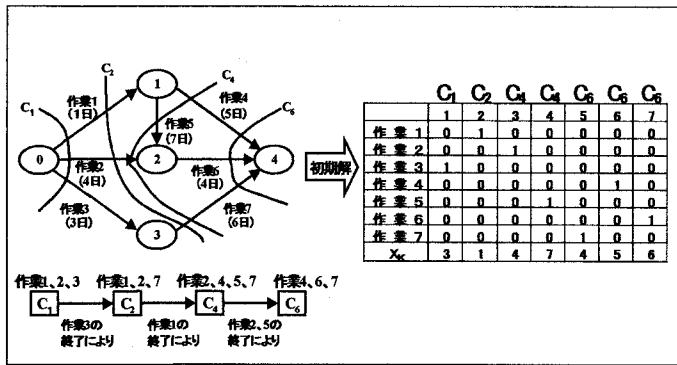


図-4 カットパスに基づく初期解の設定

へ配分する。b) 遷移を決定している作業が複数存在する場合は、それらの作業がはじめて現れるカットのレベルが小さい作業を優先して配分することにより、図-4のような初期スケジュールとマトリックスの各列が対応するカットを設定する。なお、この初期スケジュールはカットパスごとに完全にユニークである。

上述したスケジュールマトリックスに対して図-5のような列生成法による改良収束計算を行う。この過程により、局所最適解集合のみの解探索が可能となる。

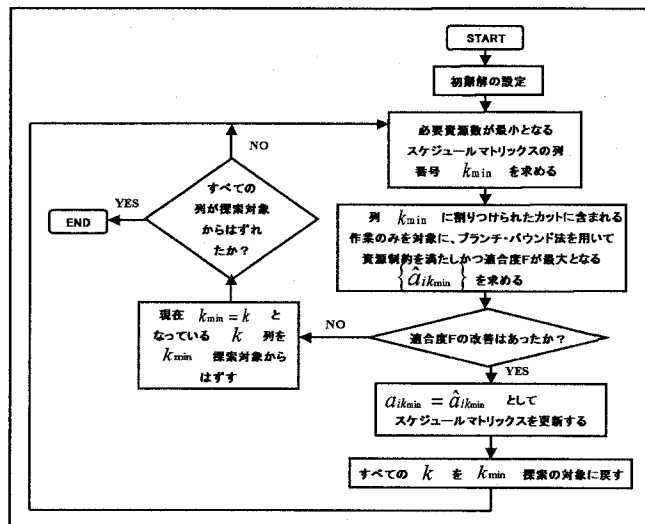


図-5 列生成による改良収束計算フロー

7. GA オペレータの概要

ここでは、本モデル中の GA の基本的動作に関して説明を加える。

a) 初期解群の生成

ここでの初期解の生成とは、設定した個体数分のカットパスを作成することをいう。まず、ランダムにノード順序列を作成する。つづいて、以下のような接続行列 (δ_{ij}) の列ベクトルを作成したノード順序列に従

って並べ替える。

$$\text{接続行列 } (\delta_{ij}) = \begin{cases} 1; \text{ノード } j \text{ が作業 } i \text{ の始点であるとき} \\ -1; \text{ノード } j \text{ が作業 } i \text{ の終点であるとき} \\ 0; \text{その他のとき} \end{cases}$$

このとき、図-6に示すように開始結合点から最終結合点へと接続行列の列ベクトルを加算していき、すべての段階における加算結果に“-1”の要素が出現しないようなカットパスが実行可能なカットパスになっており、この手順を個体数分のカットパスが生成されるまで繰り返す。

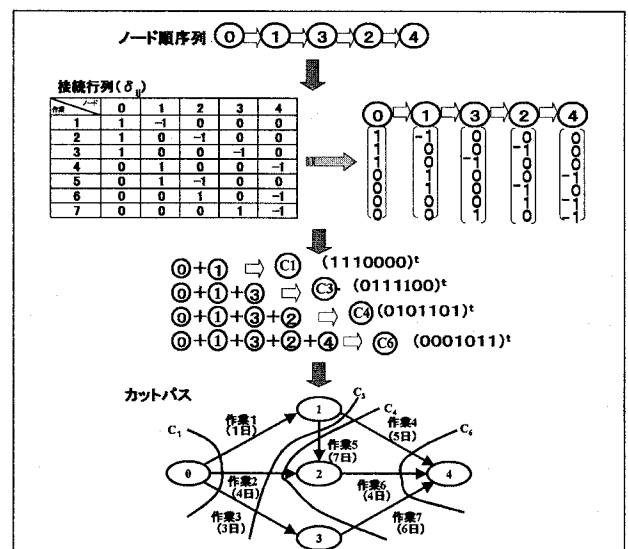


図-6 初期解の設定方法

b) 再生

再生とは、個体の中で問題への適合度が高い個体は増殖し、逆に低いものは淘汰されることをいう。今回は再生のバランスを重視して、ランキング選択⁶⁾を採用する。

c) 交叉

交叉の目的は、親の遺伝子情報を子供に伝えることにある。まず、2つの親個体をランダムに選択し交叉確率 P_c で交叉を起こす。本研究では、個体としてカットパスマトリックスを採用しているため、1点交叉や2点交叉などの標準的な交叉方法を適用した場合、個体の実行可能性を保持することは難しい。このため、カットパスの構造特性を考慮した新たな交叉方法を開発する。基本的には、任意の同レベルにある2つのカ

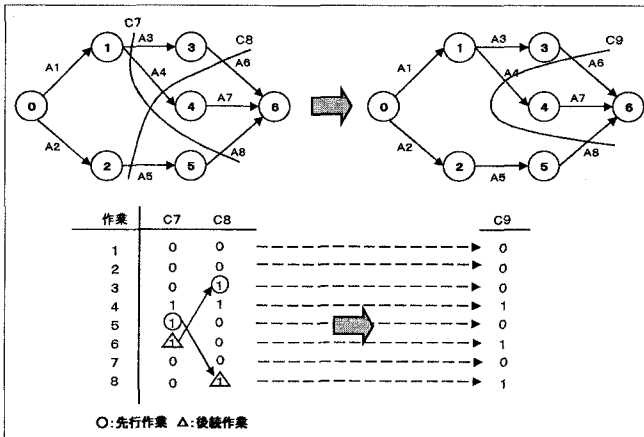


図-7 カットX交叉

ットに対して図-7のように、それぞれのカットが交差している点より右側の部分を、次レベルの新しいカットとして生成することを繰り返し、2つの子個体を作成することを考える。具体的には、次のような方法で交叉を行う。いま、子1のレベル*i*のカット子1 C_i を生成することを考える。このとき、レベル*i*-1のカットは、親1 C_{i-1} 、親2 C_{i-1} 、及び既に生成されている子1 C_{i-1} の3つが存在する。ここで、親1 C_{i-1} と合成する相手として、子1 C_{i-1} を優先し合成する。但し、親1 C_{i-1} =子1 C_{i-1} ならば親2 C_{i-1} との合成を行い、さらに親1 C_{i-1} =親2 C_{i-1} ならば、子1 C_i =親1 C_i とする。なお子2を生成する場合には、親2を中心として同様のことを行う。これにより、親1の影響を強く受けた子1と親2の影響を強く受けた子2の2つの子個体が生成される。以上の詳細なプロセスと簡単

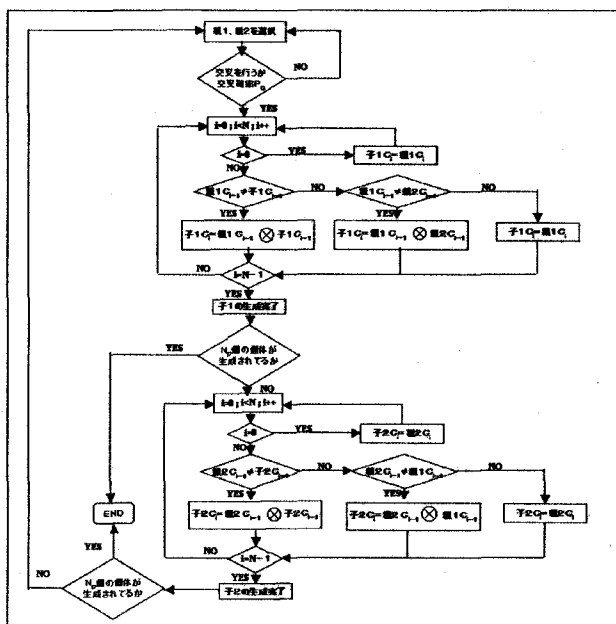


図-8 交叉のフロー

なネットワークにおける交叉手順例を図-8、図-9にそれぞれ示す。なお、図中の⊗は、図-8に示した交叉するカットの合成処理を表すこととする。

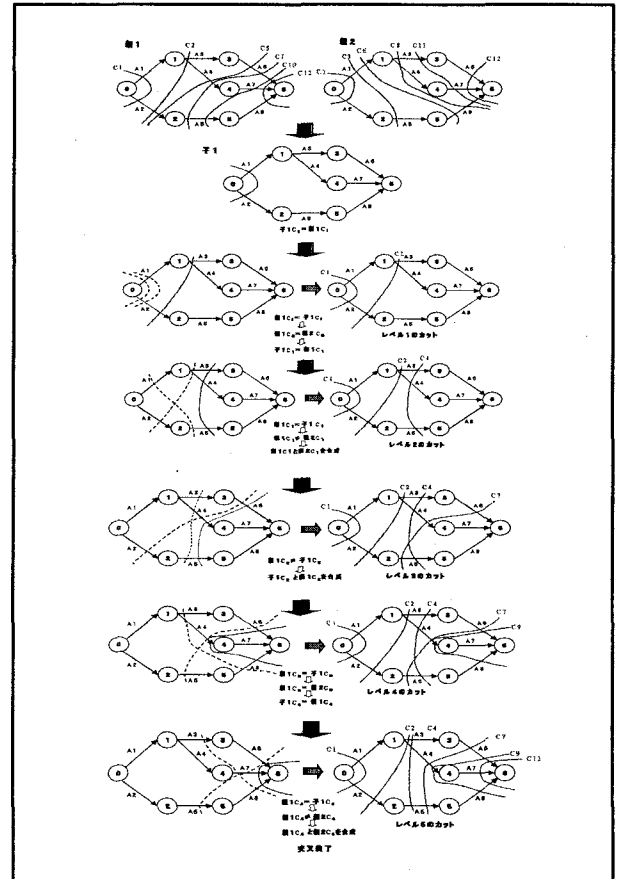


図-9 交叉手順の一例

d) 突然変異

交叉によって新たに生成された N_p 個の子個体に対して、突然変異確率 P_m で突然変異を起こす。しかし、交叉と同様、標準的に採用されているランダムな位置での対立遺伝子の転換方法では、個体の実行可能性を保持することが困難となるため、本研究では以下のような方法で突然変異を行う。

- 1) 図-10のようにカットパスをノード順序列へ変換する。
- 2) 2つのノードをランダムに選択し、それらを先頭と最後尾として部分順序列を取り出す。
- 3) 取り出した部分順序列の順序行列を作成する。
- 4) 順序行列の列がすべて0の（先行作業のない）ノードの中からランダムに1つ選択し、元のノード順序列の要素が空になっている先頭に加える。
- 5) 選択されたノードの行と列を順序行列から削除する。
- 6) 4) ~ 5) をすべてのノードが削除されるまで繰

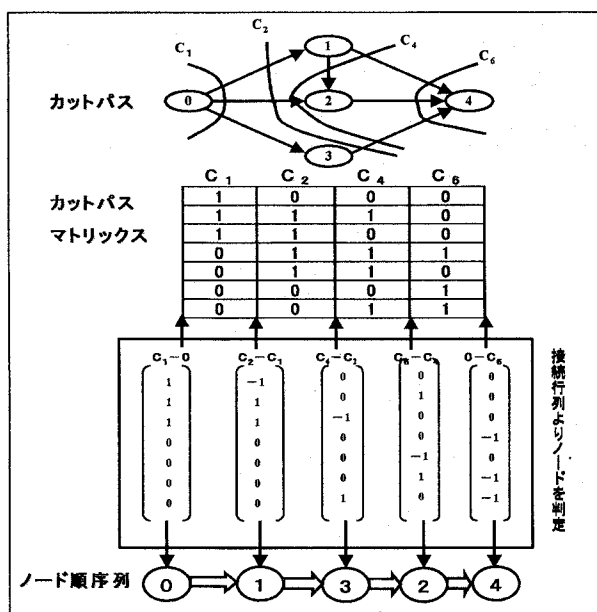


図-10 カットパスからノード順序列への変換

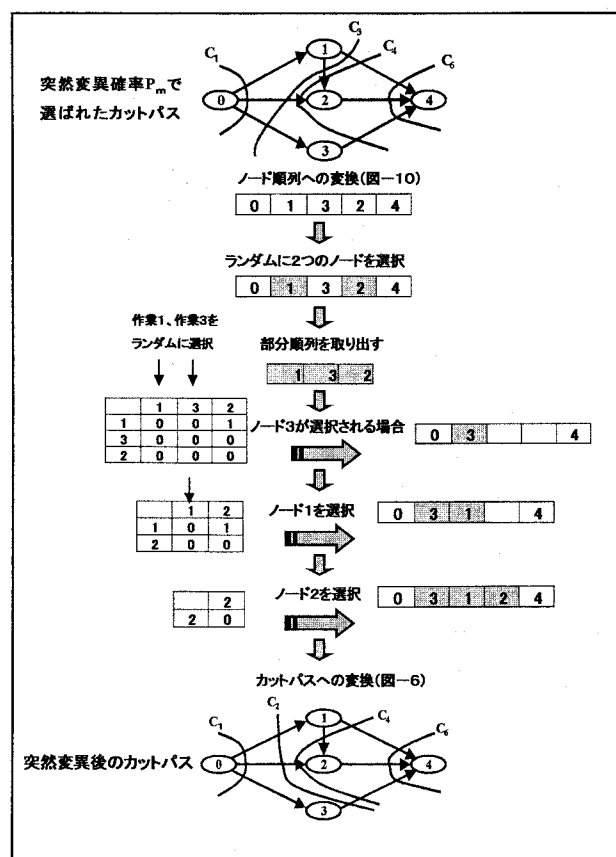


図-11 突然変異処理の一例

り返す。

7) 新たに作成されたノード順序列を図-6の方法でカットパスに変換する。

e) 世代の更新

世代の更新は、現世代の子個体を親個体として世代を1つ進めることをいう。これは当然、設定された終

了世代数 X まで繰り返し行われるが、本研究ではもし現世代において前世代よりも最良の適合度をもつ個体が出現するとその時点でその個体をエリート保存する。そして、後の世代においてこの最良の適合度をもつ個体を超越する個体が出現しない場合は、その世代における最悪の適合度を持つ個体と保存しているエリート個体とを入れ替えることとしている。

また、これまでに説明を加えてきた、個体表現・表現型への変換方法・GAオペレーターにおける作業の詳細な手順を、整理した全体スケジューリングアルゴリズムのフローを図-12に示す。

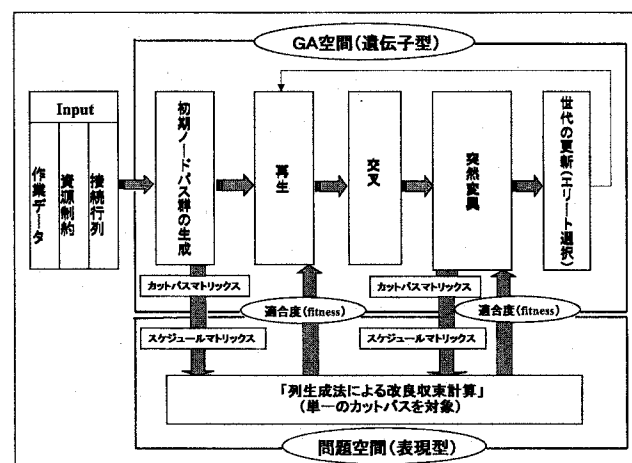


図-12 全体フロー

8. 開発アルゴリズムの性能評価

この節では、開発したスケジューリングアルゴリズムを例題ネットワークに適用し、従来法である山崩し法および、奥谷⁵⁾による既存のGA構成法との比較を行う。なお、今回の山崩し法計算は、次の優先順位規則を設定したものを用いている。1) TFの小さい作業を優先する。2) 作業日数の小さい作業を優先する。例題ネットワークとしては、作業数65ノード数41のネットワークを用いることとし、①アロー・ダイアグラム型(図-13)と②プロジェクト型(図-14)の2種類を用意した。①を開発アルゴリズムに②を山崩し法計算、既存GAアルゴリズムに適用する。また、②における終始ノードはダミー作業とする。作業データ(表-1)には、各作業に所要日数と必要資源をそれぞれ設定した。このデータに対し本研究では、工期制約を210日、資源制約を15に設定し数値計算を行った。また、GAパラメータの設定として、交叉確率 $P_c=0.4$ 、突然変異確率 $P_m=0.1$ 、各世代の固体数

$N_p=10$ と設定した。また、適合度関数の $\alpha_1 \sim \alpha_4$ については、すべて 0.25 に設定した。

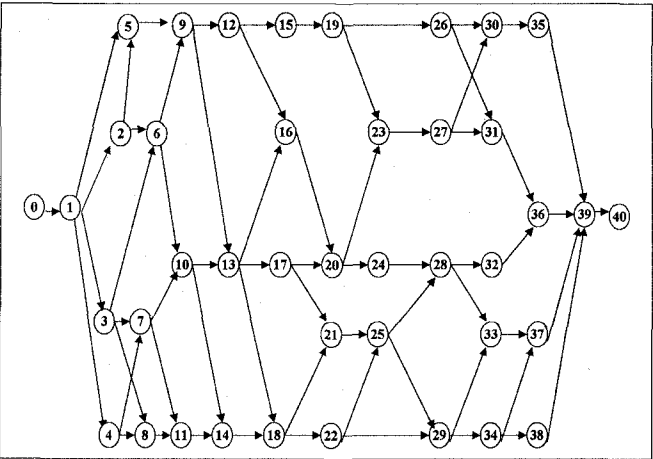
$$F = 0.25f_1 + 0.25f_2 + 0.25f_3 + 0.25f_4$$

$$(\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.25)$$

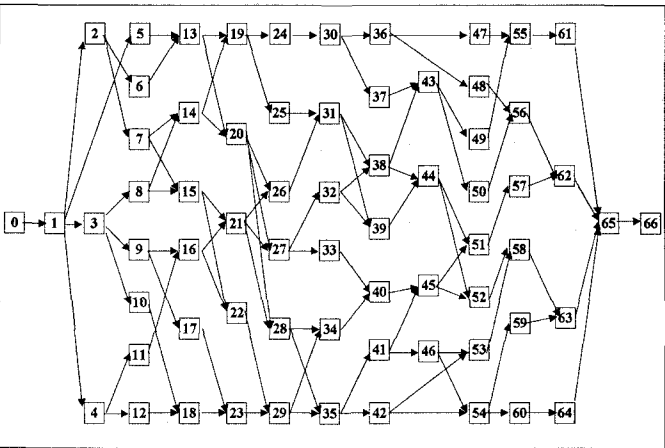
適用計算の結果、最良解として求められたスケジュール案の各目的関数値及び適合度を表－2 に示す。

なお、既存の GA 構成法においては、48 時間の計算をおこなったが、今回の問題設定では実行可能な初期解をランダムな過程の中で見つけることができなかった。これは、今回の問題においては、追加アークの候補となる変数の数が 952 と比較的多くなることにより、単純なランダム過程では、工期制約・資源制約を満たす実行可能解を生成することが困難であったためだと思われる。このため、表－2 には、山崩し法との比較のみを示している。なお、この開発アルゴリズムの結果は、60 分の計算を行った時点の最良解のものである。まず、工期についてであるが、山崩し法では、工期制約として設定した 210 を満足するスケジュールは求めることはできなかった。それに対し、開発アルゴリズムでは 199 と工期の短縮化が図られている。また、資源使用効率についても、山崩し法では約 78% であるのに対して、開発アルゴリズムでは、約 84% と使用効率の高いスケジュールが求められている。しかし、資源の平滑度については、山崩し法より劣る結果が得られた。これは、平滑度が同一工期あるいは非常に近い工期となるスケジュール代替案の資源平滑化の程度を比較するための指標であり、上述のように工期が大きく異なるスケジュール案の比較には適さないためである。事実、資源平滑化に関する別の指標として用いられる各日の使用資源量の分散値で両スケジュールを比較した場合には、山崩し法では 6.19 であり、開発アルゴリズムでは 4.89 と評価は逆転することとなる。今後、現場技術者の定性的評価と指標による定量評価の相関分析等を通して、より効果的な資源平滑化指標の検討が必要である。加えて、開発アルゴリズムの最終世代で求められた全ての個体が、山崩し法

よりも高い適合度を示しており、多くの高精度な代替案を作成することができたことがわかった。



図－13 アロー・ダイアグラム型例題ネットワーク



図－14 プロジェクトグラフ型例題ネットワーク

表－1 アクティビティデータ

作業	日数	資源数	作業	日数	資源数	作業	日数	資源数
(0, 1)	3	5	(11, 14)	5	6	(25, 28)	9	12
(1, 2)	5	7	(12, 15)	3	4	(25, 29)	8	11
(1, 3)	4	5	(12, 16)	5	6	(26, 30)	6	8
(1, 4)	6	8	(13, 16)	6	8	(26, 31)	9	13
(1, 5)	1	3	(13, 17)	2	2	(27, 30)	5	6
(2, 5)	2	4	(13, 18)	8	10	(27, 31)	3	3
(2, 6)	4	5	(14, 18)	3	5	(28, 32)	1	2
(3, 6)	3	5	(15, 19)	4	6	(28, 33)	5	7
(4, 7)	3	6	(16, 20)	5	8	(29, 33)	4	6
(4, 8)	4	7	(17, 20)	3	4	(29, 34)	7	9
(5, 7)	6	8	(17, 21)	7	9	(30, 35)	3	5
(5, 8)	3	4	(18, 21)	5	7	(31, 36)	5	6
(5, 9)	4	5	(18, 22)	4	5	(32, 36)	6	8
(6, 9)	5	6	(19, 26)	7	10	(33, 37)	4	4
(6, 10)	5	6	(19, 23)	5	6	(34, 37)	8	9
(7, 10)	7	9	(20, 23)	6	8	(34, 38)	7	7
(7, 11)	6	7	(20, 24)	2	4	(35, 39)	8	9
(8, 11)	4	5	(21, 25)	1	2	(36, 39)	4	5
(9, 12)	3	4	(22, 25)	4	6	(37, 39)	5	7
(9, 13)	9	11	(22, 29)	8	11	(38, 39)	7	8
(10, 13)	4	5	(23, 27)	5	7	(39, 40)	6	9
(10, 14)	11	13	(24, 28)	6	8			

表－2 計算結果の比較

手法	計画目標	Td	W	H	E	適合度
開発アルゴリズム		199	15	32530	0.83953	0.734613
山崩し法計算結果		213	15	30802	0.78453	0.685781

9. おわりに

本研究は、資源制約を満たす実行可能解集合の中でも特に局所最適解集合のみを対象とした解探索を実現させるとともに、多様な計画目的に対して精度の高い計画案を、効率的に求めることのできるスケジューリングアルゴリズムの開発を目指して研究を行った。本研究で得られた成果として次が挙げられる。

- ①：個体表現してカットパスを用い、列生成による収束計算を行うことで局所最適解集合のみを対象とした解探索を実現した。
- ②：性能評価においても、既存アルゴリズム及び山崩し法計算結果よりも、工期、資源使用効率および総合的な適合度において高精度な代替案を求めることができることがわかった。

また、今後の課題としては、次が挙げられる。

- ①：性能評価について、1種類の例題ネットワークに対する評価であるため今後としては、より多くの数値実験を行ったうえで、本アルゴリズムの有効性と適用上の限界を明らかにする必要がある。
- ②：実際工事への適用計算を行う必要がある。
- ③：資源山積み図とともに重要計画情報となる出来高曲線を扱うことが可能な手法へ発展させる必要がある。

【参考文献】

- 1) (i) Wiest, J. D. ; A Heuristic Model for Scheduling Large Project with Limited Resource, Management Science, vol. 13, No. 6, pp. B-359-B37 7, 1967 .
(ii) 坂本実；プロジェクト・ネットワーク上の資源配分問題－ヒューリスティック・プログラミングによる解法－, IE, vol. 14, NO. 3, pp. 101-108, 1975.
- 2) 春名攻, 滑川達：PERT/MANPOWER 問題の最適解の開発研究－カットネットワークにおける最適資源配分問題への変換を用いた新しい解法－土木計画学・論文集 No. 15, pp. 41-48, 1998.
- 3) 日高ほか：ニューラルネットによる資源配分を考慮したスケジューリング問題の開放, 土木学会論文集, No. 458, 101, 1993.
- 4) 奥谷巖ほか：マンスケジューリングにおけるタブー探索法の適用性, 土木計画学・論文集 No. 13, pp. 323-330, 1996.
- 5) 奥谷巖ほか：マンスケジューリング問題における遺伝的アルゴリズムの適用性, 電気学会論文誌 C, No. 4 vol. 114-C, pp. 450-455, 1994.
- 6) 坂和正敏, 田中正博：遺伝的アルゴリズム, 朝倉書店, 2000

A Study on Multi-object project scheduling utilizing GA

In this study, It is developed as follows that new-type systematic scheduling procedure by use of GA, which is very general in that the procedure can take multiple constrains as well as various forms of goal into account: At the first stage, a optimal scheduling model with limited resource problem is formulated as problem of integer programming to obtain the execution state defined by using “number of activities”×” number of activities” matrix. At the second stage, topological characteristics of the project network are analyzed mathematically focusing on the cut set structure. Thus relationship between this cut set structure and execution state which represent as “number of activities”×” number of activities “matrix studied at the above stage is discussed. At the third stage, coding, decoding and GA-operator of new type GA for project scheduling problem are proposed in accordance with cut-network concept which is established through analysis of characteristics of the project scheduling problem at the above stage. At the final stage, it is demonstrated in the model testing using example project network that the proposed procedure can improve the disadvantages of the existing methods compared.