

遺伝的アルゴリズムによる軌道保守スケジューリング

Application of Genetic Algorithm to the Track Maintenance Scheduling Problem

三和雅史^{**}・内田雅夫^{**}

By Masashi MIWA^{**}・Masao UCHIDA^{**}

1. はじめに

軌道は、円滑な列車走行を実現するための案内路の役割を果たす重要な構造物である。特に、道床バラストを有する有道床軌道は、列車の繰返し通過によって軌道面の不整の発生・成長、いわゆる軌道狂い進みを伴うため、軌道形状の復元・補修作業が必要である。この作業には、一般に、マルチプル・タイタンパ（MTT）が用いられるが、良好な軌道状態を維持しながら、費用を最小とする効率的な保守活動の実現を目指す上で、この MTT を適切に運用することが極めて重要である。そこで、本研究では、この MTT の運用計画問題に対して、遺伝的アルゴリズム¹⁾（GA:Genetic Algorithm）を用いた解法を示し、得られる解の性質を調べる。

2. MTT 運用計画問題のモデル化

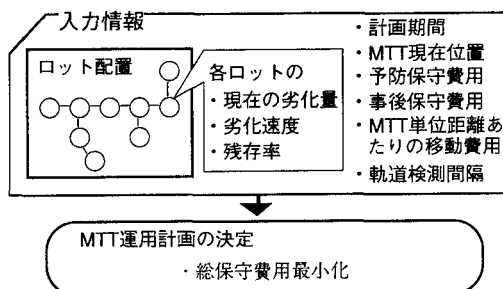
（1）用語・記号の定義

- ロット：作業区間の単位。ロットが線状につながったものが路線になる
- 期：保守計画の単位。1 期間中に 1 ロットの MTT 作業を行なうことができる
- 劣化量：軌道検測により得られる軌道の劣化の程度を表す量（ex. P 値、標準偏差）
- 劣化速度：劣化量の単位時間あたりの増加量
- 残存率：保守前の劣化量に対する保守後の劣化量の割合
- 事後保守：軌道検測の結果、劣化量が緊急に保守が必要な限度（事後保守限度）を超えていると診断された場合に行なう保守

予防保守：劣化量が事後保守限度を超過していない場合に予防保全的に行なう保守

- t ：時間変数
- T ：計画対象とする期の数
- τ ：期の長さ（定数）
- t_m ：軌道検測間隔（定数）
- k_n ：ロット n の劣化速度（定数）
- g_n ：ロット n の残存率（定数）
- $D_n(t)$ ：時刻 t におけるロット n の劣化量
- D_b ：事後保守限度
- C_p ：1 回あたりの予防保守費用
- C_b ：1 回あたりの事後保守費用
- L ：MTT の単位距離あたりの移動費用
- C_t ：総保守費用
- N ：ロット数
- M ：個体数
- M_u ：規定個体数
- C_i ：個体 i の総保守費用
- F_i ：個体 i の適応度
- U ：制御変数
- x ：世代数
- x^* ：規定世代数
- P_m ：突然変異確率

（2）想定する問題と計画の決定



図－1 MTT 運用計画決定法

* キーワーズ：計画手法論、土木施設維持管理

** 正員、(財)鉄道総合技術研究所 軌道技術開発推進部
(東京都国分寺市光町 2-8-38、TEL 0425-73-7278、FAX 0425-73-7296)

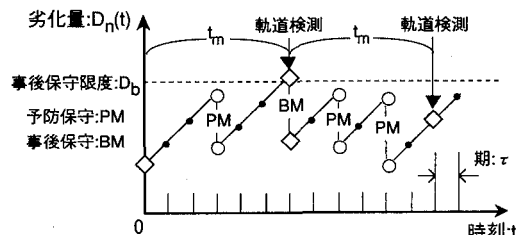


図-2 任意のロットにおける劣化量推移のイメージ

本研究では、MTTをいつ、どのロットに投入するかを決定する手法を提案する。ここでは、図-1に示す情報が与えられたとき、計画期の総保守費用が最小となるようにMTTの運用を決定する。

期 1 2 3 4 5 T-2 T-1 T
 個体 (0, 3, 0, 15, 0, ..., 0, 2, 1)
 遺伝子 $\nearrow \nearrow \nearrow \nearrow \nearrow$
 4 期目にはロットNo.15を
 保守する (T-2)期目にはどかも
 保守しない

圖一 3 個體表現

$$+ (\text{計画期間中の MTT 移動距離}) \times L \quad (3)$$

3. GAの適用

(1) GAの概要

本問題のようなロット数や計画対象とする期の数が大きなスケジューリング問題を、動的計画法のような最適解法で解くことは、組み合わせ数が膨大なことから困難である。よって、現実的には、最適解が得られなくても、それに近い近似解が効率よく得られれば十分であると考えられ、その解法の1つにGAがある。GAは生物の進化とそれを支える遺伝のメカニズムを模倣して最適化問題を解く方法であり、特に、目的関数の値さえ得られれば、目的関数の勾配情報が無くても探索可能であり、また、モデル作成の自由度が高い。更に、多点探索であるために、目的関数の値がほぼ同じ解を複数得ることができ、意思決定者は好みの解を選択することができる。以下に、GAの本問題への具体的な適用方法を示す。

(2) 個体表現とアルゴリズム

a) 個體表現

本問題に対する個体表現は図-3に示すような各期のMTT投入箇所の順列で表す。この順列はロットNo. $\{1, 2, 3, \dots, N-1, N\}$ の重複を許す順列である。ここで順列の各数字のことを遺伝子と呼ぶ。GAでは個体表現が与えられれば目的関数の値を得ることができ、それに対応して定義する適応度により、各個体が次世代に生き残る、あるいは子孫を残す確率を決定することができる。

h) アルゴリズムの概要

- STEP1 初期個体を M 個ランダムに作成する。
- STEP2 各個体 i の適合度 F_i を計算する。
- STEP3 適応度最大の個体を選択する。
- STEP4 STEP3で選択した個体以外の全個体の適応度の和 ΣF_i を計算し、個体群の中から2つの個体を選択する。ただし、個体 i が選ば

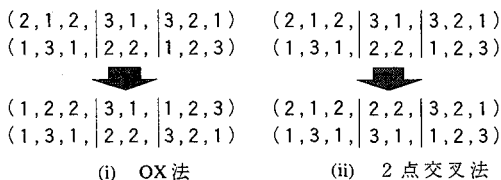


図-4 交叉の例

※確率 P_m で突然変異を起こすことを決定

任意の遺伝子を選択

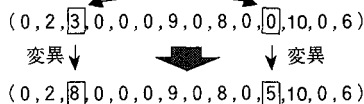


図-5 突然変異の例

れる確率は $F_i/\Sigma F_i$ とする。そして、その2つの個体から新たに2つの個体を生成する(この操作を交叉と呼ぶ)。この操作を個体数が $(M_u - 1)$ 個になるまで繰り返す。

STEP5 STEP4で生じた各個体に対して確率 P_m である変化を与えて新たな個体を生成する(この操作を突然変異と呼ぶ)。

STEP6 STEP3で選択された個体とSTEP4～5で生成された個体 M_u 個の各個体の適応度を計算し、適応度の大きい順に M 個選び、次世代に残す。

STEP7 世代数 x が予め決められた値 x^* に達したならば、最も適応度の高い個体を解として終了する。達していないならば $x = x + 1$ としてSTEP2へ戻る。

以上のように、GAでは交叉や突然変異といった遺伝演算を実行し、世代交代を繰り返して最適解を得ようとする。遺伝演算の内容を以下に示す。

c) 交叉

交叉の目的は親の遺伝情報を子孫に伝えることにあり、適応度の低い個体を排除し、探索領域を絞る役割を担っている。ここでは、個体 i の適応度 F_i を各個体の総保守費用 C_k の($k = 1 \sim N-1$)最小値、最大値と個体 i の総保守費用 C_i を用いて、次のように定義する²⁾。

$$F_i = U \min C_k + \max C_k - C_i \quad (3)$$

ここで、 U は制御変数で次のように与える。

$$U = -0.45x / x^* + 0.5 \quad (4)$$

交叉方法には各遺伝子の前後関係の保存を重視す

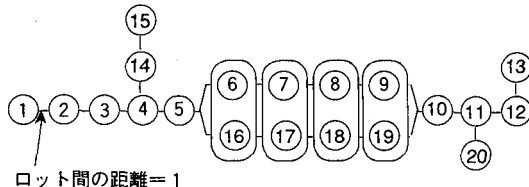


図-6 ロット配置

表-1 数値例

ロットNo.	1	2	3	4	5
$D_n(0)$	1.517	2.647	2.362	2.199	2.994
k_n	0.06944	0.06944	0.06944	0.06944	0.06944
	6	7	8	9	10
$D_n(0)$	3.099	1.940	2.593	2.655	2.769
k_n	0.05556	0.05556	0.05556	0.05556	0.06944
	11	12	13	14	15
$D_n(0)$	2.232	3.087	1.234	2.149	3.345
k_n	0.06944	0.06944	0.06944	0.08333	0.08333
	16	17	18	19	20
$D_n(0)$	2.669	2.364	2.590	1.945	3.469
k_n	0.05556	0.05556	0.05556	0.05556	0.04167

$g_n=0.4$ $D_b=4$ MTT現在位置:4 $C_p=1$ $C_b=2$ $L=0.2$
 $\tau=10$ $t_m=90$ $T=40$ $N=20$ $M=21$ $x^*=5000$ $P_m=0.1$

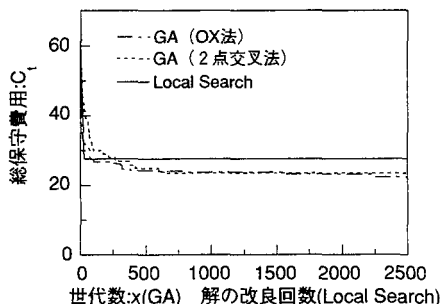


図-7 世代と総保守費用の例

るOX法³⁾と2点交叉法の2つを用いる。交叉の例を図-4に示す。

d) 突然変異

突然変異の目的は、交叉で生成する子孫が親の遺伝子に依存してしまうという弱点を補い、個体群の多様性を維持することにある。ここでは、1個体中の任意の2つの遺伝子を、確率 P_m で他の数字に置き換える。突然変異の例を図-5に示す。

4. 数値実験

(1) 方法

図-6に示すロット配置、表-1に示す数値例を用いて数値実験をした。

図－8 MTT運用計画の例

好みの解を選ぶことができる

解1 (0,0,0,3,0,5,15,0,6,16,11,12,20,10,8,9,18,0,17,0,4,0,0,0,14,2,0,0,0,0,0,0,0,0,0,0,0,0)	22.4	2.65
解2 (0,0,0,3,5,0,0,15,16,6,11,12,20,10,8,18,9,0,17,0,4,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0)	22.4	2.63
解3 (0,0,14,3,5,0,0,15,16,6,11,12,20,10,8,18,9,17,0,0,4,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0)	22.4	2.63
⋮	⋮	⋮

表－3 解法別の総保守費用

		平均値	標準偏差	最小値	最大値
GA	OX法	23.6	1.12	21	25.8
	2点交叉法	24.2	1.43	22	28.4
	Local Search	27.4	1.27	24.6	31

※各50回の計算結果による

ところで、ここでは、GAの有効性を検討するために、Local Searchにより得られる解との比較を行なう。Local Searchは以下に示す方法で行なう。

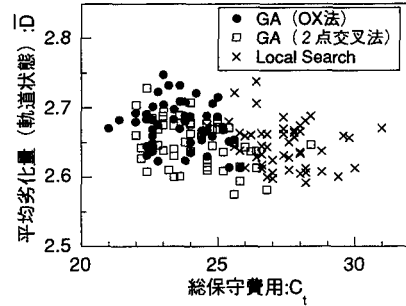
【Local Search】

- STEP1 初期解として MTT投入箇所の順序を1通りランダムに生成する。
- STEP2 作成した順序中の数字列の1箇所だけを別の数字に置き換える（解の改良）。ただし、この置き換えは、1箇所の数字を置き換えたときの総保守費用減少の効果が最大となるように行なう。
- STEP3 保守費用を減少させる入れ換えができなくなった場合、または、STEP2の操作回数が予め定めた値に達したならば、そのときの順序を解として計算を終える。

(2) 結果

まず、世代と総保守費用の関係の例を図－7に示す。これから、世代の増加に伴って費用が一定の値に収束していく過程がわかる。GAの2つの交叉方法は同じ様な収束をしている。Local Searchの場合も横軸を解の改良回数として示したが、何回かの解の改良の後、局所最適解に陥ってしまっている。なお、このとき得られた MTT運用計画（OX法の場合）を図－8に示す。

次に、解法別の総保守費用を表－3に示す。これは各々50回ずつ計算した結果である。Local Searchは初期解の周辺領域しか探索しないため、局所最適解に陥り、GAに比べると総保守費用が大きくなる傾向にある。GAの交叉方法による差は、OX法の方が全項目で値が小さい。



図－9 解と軌道状態の関係

最後に、得られた解に対応した期間中の全ロットの平均劣化量（軌道状態）を図－9に示す。Local Searchの解では、GAより大きな費用でGAによる解と同じレベルの軌道状態となっていることから、GAの優位性がわかる。次に、GAの2つの交叉方法を比較すると、同じ費用で実現する軌道状態は2点交叉法の方が良い傾向にあるが、同じ軌道状態に対する費用のバラツキは大きい。

5. おわりに

本研究では、GAを用いた MTTの運用計画法を提案した。また、数値実験により得られる解の性質を調べ、本手法の実用性を検討した。

さて、実際の MTTの運用計画は、今回想定した問題以上に複雑であり、本手法を実際の MTT運用計画に適用できる形への改良が、今後の課題である。

参考文献

- 1) Goldberg, D. E. : Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, 1989.
- 2) 飯間等・三宮信夫：遺伝アルゴリズムによる製品投入順序問題の解法, 計測自動制御学会論文集, Vol28, No11, pp.1337-1344, 1992.
- 3) 三宮信夫：スケジューリング問題に対する遺伝アルゴリズム, オペレーションズ・リサーチ, 39, 12, PP.659-664, 1994.