Genetic Algorithms による航空乗務ペアリング

非定期便を含めた統合的アプローチ

The Fully Dated Airline Crew Pairing by Genetic Algorithms

佐藤 眞木彦

Makihiko Sato

富士通 計算科学技術センター 知的システム研究部

Intelligent Systems Laboratory, Computational Science and Engineering Center, Fujitsu LTD.
maki@strad.se.fujitsu.co.jp

松本 俊二

Shunji Matsumoto

(同上)

shun@strad.se.fujitsu.co.jp

寺本 陽彦

Youhiko Teramoto

(同上)

tera@strad.se.fujitsu.co.jp

安達 統衛

Nobue Adachi

富士通研究所 コンピュータシステム研究所 知能システム研究部

Intelligent Systems Laboratory, Computer Systems Laboratories, Fujitsu Laboratories LTD.
adachi@ias.flab.fujitsu.co.jp

keywords: crew pairing, fully dated, genetic algorithms, airline

Summary

Crew Pairing is one of the most important and difficult problems for airline companies. Next to fuel costs, the crew costs constitute the largest cost of airlines, and the crew costs depend on the quality of the solution to the pairing problem.

Conventional systems have been used to solve a daily model, which handles only regular flights with many simplifications, so a lot of corrections are needed to get a feasible solution and the quality of the solution is not so high.

A fully dated model, which handles regular flights and irregular flights simultaneously, is extremely hard and has not been solved directly hitherto. The number of irregular flights tend to increase in Europe and Japan, hence the resolution of the practical fully dated pairing system is desired.

This paper presents a new approach which solves directly the fully dated crew pairing by Genetic Algorithms. GA is improved with stochastic processes to attack the fully dated model, and many heuristics are included in the decoding mechanism of the GA. For several hundred flights per day for fully dated 2 months scheduling period, the system found better or equivalent solutions to the human scheduler's without any simplification from 5 to 20 times faster.

1. はじめに

航空会社において運航乗務員の費用は、燃料費の次に大きなコストを占め、その削減は大きな問題である。このためには与えられたダイヤに効率良く乗務員を割り当てる必要がある。この問題はクルーペアリング Crew Pairing (CP) と呼ばれており、多くの研究事例があるが、これまではダイヤの非定期性をうまく扱うことができていた。我々は Genetic Algorithms (GA) を適用して非定期便を含む CP を統合的に解き、CP の専門家に匹敵する解が得られるシステムを開発した [寺本 99, Matsumoto 99, 松本 00]。

このシステムでは、1 日あたり数百便でスケジュール期間は 2 ヶ月の規模の問題を、問題を単純化することなく扱うことが可能である。本稿では、まず CP の問題設

定と CP で非定期便を扱う困難さを示し、従来手法について説明した後に、非定期便を含んだ CP にヒューリスティクスと確率的デコード手法を導入した GA をいかに適用したかを述べる。

2. クルーペアリング

クルーペアリング (Crew Pairing : CP) とは、1 ヶ月～2 ヶ月間に渡る所与の運航ダイヤに対して、個々の便に一人の乗務員を割り当てる問題である [Andersson 97]。ここでの乗務員とは機長、副機長、及び航空機関士のことである。CP の目的は、乗務員へ課せられる多くの制約条件を満たしながら、パターン数 (1 日当たりの運航に要する総乗務員数) を最小化することである。パターン数の少ない CP とは、乗務員の稼働効率が高いという事

を意味する。

乗務員は所属する基地空港から出発し、乗務を重ねながら、自らの基地まで帰還する。乗務員は必要に応じて出先で宿泊し、定められた日数以内に帰還する。これらの一連の乗務の割り当てをペアリング (pairing) という。

乗務員は、次の乗務や基地への帰還に、(自社の) 航空便や列車で移動する場合があります、これを便乗 (deadhead) という。deadhead は乗務員を全ての便に割り付けるため、つまり実行可能な解を得るために必要であるが、一方、探索空間を莫大にする要因ともなっている。

航空機は機種によっては 1 日当たり最大 10 便程度運航される。一方、乗務には様々な制約条件があり、航空機の一日の運航便全てに一人の乗務員が乗務することはできないため、複数の乗務員を個々の航空機の 1 日の運航に割り当てる必要が生じる。乗務員の人件費 (訓練費等も含む) は高額なため、良い CP の解が得られればコスト削減効果は極めて大きい。

CP には、以下のような特徴がある。

- CP 対象の便には、過不足なく乗務員が割り当てられなくてはならない。
- 乗務員には、離発着回数、乗務時間、勤務時間等の法規制上、及び、労働協約上の制約を厳守しなければならない。これらの制約の中には、2 日間に渡ってペアリングを行なった後でなければ制約違反と判定できないものもある。
- 機種毎に乗務員の免許が異なるため、機種別にペアリングする必要がある。また CP は各乗務員タイプ (機長/副機長/航空機関士) に解かれる。
- 宿泊可能な空港、deadhead に鉄道を使っても良い空港等、様々な制約条件が指定されている。
- 無駄な deadhead を避けるため、連続した deadhead はしてはならない。また deadhead だけの勤務日を含むペアリングは、不可避な場合以外は行なわない。
- 機長は割り当てられる便の路線毎の離発着免許を取得している必要がある。
- 航空会社では、顧客の需要に応えるため、2ヶ月に一回程度ダイヤの改正を行なう。CP のスケジューリング期間は、この期間に対応している。改正されたダイヤ間の調整の問題も別途生じるが、その問題は本稿の CP の対象外である。

3. daily モデル

3.1 daily モデル

daily モデルでは、連日同一のダイヤが運航されるものと仮定し、最小のパターン数で全ての便に乗務員を割り付ける問題である。daily モデルでは、後述の非定期便は対象としていない。

図 1 は daily モデルのペアリング例である。左方向への矢印は、宿泊して翌日に繋がっていることを示している。

図 1 daily モデルのペアリング例

ここでは、東京を基地とするある乗務員は

- 1 日目 航空機 A で便 1 → 便 2 → 便 3 と乗務、長崎泊
- 2 日目 航空機 B で便 4 → 便 5 と乗務、大阪泊
- 3 日目 航空機 C で便 8 → 便 9 と乗務し帰還。

大阪を基地とするある乗務員は、

- 1 日目 航空機 B で便 6 → 便 7 と乗務、札幌泊
- 2 日目 航空機 C で便 9 を deadhead、便 10 → 便 11 と乗務し帰還。

ある 1 日の冒頭を取ってみれば、便 1 に乗務を開始する 1 人がおり、便 4 には前日便 1 から乗務を開始した 1 人が、更に便 8 には、前日に便 4 からの乗務を行なった 1 人がおり、このペアリングを通して毎日 3 人の乗務員を要する。また、便 6 に乗務を開始する 1 人と、便 9 で deadhead する 1 人がおり、このペアリングには毎日 2 人の乗務員を要する。したがって全体で、このダイヤのパターン数は 5 となる。

3.2 weekly モデル

weekly モデルは運航周期を 1 週間としたモデルである。北米に多い形態の運航では、週末の運航は、平日の運航の一部を間引いた形態をしており、weekly モデルの 1 つとして扱える。この場合の weekly モデルは weekly exception problem として、daily モデルの解をベースに週末便を間引いた問題設定がなされている。ヨーロッパや日本では、不規則性が高いため、weekly モデルでは扱えない。

4. 非定期便と fully dated モデル

実際のダイヤは毎日同じ機種で運航する定期便 (regular flight) 以外に、日によって運航が異なる非定期便 (irregular flight) が混在する。

全ての定期便、非定期便を取り入れたモデルを fully dated モデルと言い、休日の運航や機種の変更、臨時便が全て含まれる実用に則したモデリングである。

一見定期便でも日によって運航機種が異なるものは、CP では非定期便として扱わねばならないため、実際のダイヤでの非定期便はかなり多い。非定期便の数は、北米地域はさほどではないが、ヨーロッパや日本ではかなり多く全体の 3 割程度を占める。顧客の需要になるべく応えるため、非定期便は増加する傾向にあり、fully dated モデルを解く意義は大きくなる一方である。

図 2 非定期便が入ったペアリング例

4.1 fully dated モデルの困難さ

図 2 は非定期便が入ったペアリングの例で、“航空機 C” は奇数日のみ運航される。

東京発の便 1 からの乗務は、3 日目の奇数日は“航空機 C”に乗務するが、3 日目の偶数日は“航空機 C”が運航されないため、便 5 から“航空機 D”の便 12, 13 の偶数日に接続して基地に帰還することになる。

この際、“航空機 D”は定期便なので、便 5 からの接続を受けない奇数日の便 12, 13 への割り当てを別途考慮する必要がある。

また、“航空機 B”便 7 からの 2 日目の接続は、同様に接続可能な“航空機 E”便 14, 15 に偶数日に接続する必要がある、残った運航日が上と同様の問題を引き起こす。

このような非定期便が新たな非定期便を作り出す、非定期便の波及効果が生じるために、問題はますます複雑になってしまう。

非定期便でも、ある往復だけが日によって異なるだけのもがあり、このような非定期便を統合的に取り扱えれば望ましいが、単純にまとめて扱うと制約違反が生じる可能性があるため、無条件には行えない。

また、乗務日と運航日の擦り合わせも大きな問題である。例えば、乗務 1 日目が偶数日なら、その乗務員の 2 日目は奇数日だけである。つまり非定期便を扱うためには、乗務員の乗務日の非定期的なズレまで考慮する必要がある。

問題の難しさの指標の 1 つは問題規模である。daily モデルでは、大雑把には問題規模は“1 日の便数”であったが、fully dated モデルでは、非定期便が存在するために“1 日の便数”×“スケジュール日数”になる。つまり daily モデルに比べて大きく難しいといえる。

fully dated なペアリングのパターン数は、以下の式で計算される。

$$\text{パターン数} = \frac{\text{ペアリングの延べ運航日数}}{\text{スケジュール期間}} \quad (1)$$

この例での“航空機 C”が絡む部分のパターン数は、運航が隔日であるため、スケジュール期間を 30 日とすれば東京発のものが $\frac{15 \times 3}{30} = 1.5$ 、大阪発のものが $\frac{15 \times 2}{30} = 1.0$ で、計 2.5 パターンとなる。“航空機 D”、“航空機 E”が絡む部分まで入れば、5 パターンとなる(“航空機 D”、“航空機 E”の残った運航日の部分は除く)。

5. 従来手法

CP は長年 OR で扱われてきており、既存の研究事例の多くが、daily モデルの上で解かれている。従来手法では、daily モデルを集合分割問題 (Set Partitioning Problem) や、deadhead を含めて重複した割当てとみなす集合被覆問題 (Set Covering Problem) として扱っている。最適化手法としては LP (Linear Programming)/IP (Integer Programming) が用いられ、規模としては数百便に及ぶものもあるが、非定期便を扱っていない。

例えば [Vance 97] では問題を 2 段階に分けて、問題空間の縮小を図った提案をしているが、定期便のみの扱いで便数は 174 である。また [Andersson 97] では、CARMEN という CP システムを商用化しており、これは制約条件を扱うルールベース・システムと IP を用いた最適化システムを組み合わせたものであるが、非定期便は扱っていない。

fully dated の例として、[Tajima 97] において、先ず daily モデルを数理計画法を用いて解き、それに逐次非定期便を付加する手法が提案されている。しかし真に fully dated な取扱いをした場合は非定期便をうまく組み合わせることで必要な乗務員数を削減することができるがあるが、非定期便を後付けする手法ではこのような総合的最適化が期待できないと思われる。また [Andersson 97] は、fully dated モデルをコンピュータ上で実際に解くことは非常に困難であると述べている。

一方、実際に航空会社の専門家が fully dated な CP を解く場合は、その専門家は自社のダイヤの構造を熟知しており、便をどのように乗り継いだら良い結果が得られるかのパターンを経験的に把握している。専門家は、それらを踏まえて全体を見渡ししながら、非定期便を定期便と同時に扱って実行可能解を作成し、その解に推敲を積み重ねて最終的な解を出していく。その結果、専門家の出す解は必ずしも最適解ではないが、既知のものとしては最も良いものとなっている。

6. 確率的デコードを用いた GA による解法

fully dated モデル CP 問題の性質や特徴を検討し、下記のような特長を持つ確率的デコードを用いた GA を構成した。

- (1) ヒューリスティクスを用いた確率的デコード
- (2) 遺伝子の多重化

この構成を採用した背景をはじめに説明する。

(1) ヒューリスティクスを用いた確率的デコード

fully dated な CP のような問題では解空間が膨大であり、単純に解空間へ一意にマッピングを行なうようなコーディングは長大となり、また様々な制約が複雑に絡んでおり、GA の遺伝操作だけで探索するのは効率的ではない。

図 3 通常の GA と確率的デコードを用いた GA

そこで、様々な制約を効果的に充足して求解を高速化するために、それぞれの染色体に対して解空間の 1 点ではなくて解空間の部分範囲を対応 (1 対多マッピング) させることで、染色体空間を縮退させ、デコードの過程にヒューリスティクス探索を組み入れ、1 つの染色体の指す解部分空間の中から 1 つの解を求めることにした。さらにこの探索手法として、個々の染色体のデコードでバックトラックなどを用いると計算時間がかかること、複雑な制約を効率的に解消できるヒューリスティクスを用いたデコーダを設計するのが一般に容易でないことから、探索途中で現れる条件分岐において確率で分岐していくことで、短時間に 1 つの解を与える確率的手法を導入した。

本システムでは、後で染色体表現やデコード処理に述べるように、解 (ペアリング) の主要な構成要素であるセグメントや、セグメントの接続に関する情報のみを染色体上に表現している。これにより、解と一意に対応させる従来流の染色体表現と較べてコンパクトになり、染色体空間の大きさは小さいものとなっている。更にデコードには、ヒューリスティクスを反映させた確率テーブルをもちいた確率的分岐による探索手法を採用している。

(2) 遺伝子の多重化

探索を丹念に行ないたい部分については、対応する遺伝子の対立遺伝子を複数個染色体上に用意し、ある対立遺伝子でのトライが失敗した時に、次々と代替対立遺伝子を試みる仕組みを用意することにより、効率的に探索できることが、我々の試行錯誤の過程で分かってきた。これは探索の範囲を、等確率に取り出す場合より、解がありそうな領域に絞るために有効であり、確率的デコードだけでは引き当てにくい良い解を探しやすくするという相補的效果を持つと考える。

本システムでは、接続候補遺伝子の多重化として実装されている。デコード過程でセグメントを順次接続することでペアリングを探索していくが、そのセグメントが他のセグメントと競合して接続できない場合、別の代替セグメントの接続を試みる。1 つの接続だけを染色体にコーディングされている場合、確率的探索だけでは有望な代替セグメントを選びにくい。そこで、複数個の接続を染色体上に表現することで探索を助けている。さらに、成功した接続を染色体に反映させるラマルク遺伝的

仕組みを付加している。

確率的デコードを用いた GA の特徴

図 3 は、通常の GA と確率的デコードを用いた GA の染色体空間と解空間のマッピングを示したものである。図の左のように通常の GA では、染色体空間の方が解空間より等しいか大きく、ある染色体に対して出力値が一意になるのに対して、図の右の確率的デコードを用いた GA では、染色体空間より解空間の方が大きく、ある染色体はデコード毎に確率的に異なる出力値を与える。

この手法の利点として、複雑な問題において染色体空間が縮退しているため、染色体の設計が簡素になり、デコード時の探索処理で複雑かつ明示的なインプリメントが不要になる点が挙げられる。

複雑な制約を持つ実問題では、問題固有なヒューリスティクスがあったとしても、多くの場合部分的・不完全なものであり、制約条件等を効率的に解消するデコード処理の設計が非常に難しい。これを難しくしている要因は、デコード処理の過程で現れるさまざまな論理的分岐において、分岐先を決定論的に決める処理である、と思われた。そこで、明白なヒューリスティクスを適用しにくい場合に、分岐先をヒューリスティクスを取り入れた確率によって決めることにより、デコード処理の設計が簡素化できると考えている。ヒューリスティクスを特定しにくい部分を確率的探索に委ね、その結果の取捨選択を GA でおこなうことはごく自然なことと思われる。

また、デコード時の探索に用いられるヒューリスティクスが反映された確率過程により、探索が確率が高い領域に集中するので、GA の探索効率の改善がなされている。さらに、確率が低い領域も完全に捨てずに探索を行っているため、GA 探索の持つ柔軟さも保持されている。このように、この手法は GA とヒューリスティクスを互いに良い点を損なわずにハイブリッドさせている。

この手法の弱点として、染色体から一意に解が再現しないことが挙げられるが、常にその時点での最良の染色体とその解を、その世代までの最良解として保存することで、この点は容易に補われる。

本手法は CP を解くために考案したが、その考え方は CP に限らず他問題に応用可能と思われ、例えば、通常的手法では事前に的確なコーディングが分からない実問題などに有効と考える。

6.1 デコードのおおまかな流れ

本システムは、GA の最適化システムではあるが、上記で述べたようにデコードの処理が非常に大きな役割を演じている。

本システムでは、ある個体の染色体は、同一航空機で連続して運航する便を 1 まとめにしたセグメントを中心に、以下のようにデコードされる。セグメントについては、6.4 節で詳しく説明する。

図 4 染色体

- (1) セグメント化 — いくつかの便をセグメントにまとめる.
- (2) ペアリング順リスト作成 — ペアリングの先頭として取り出すセグメントの順番をきめる.
- (3) ペアリング — ペアリング順リストから先頭のセグメントを取り出し、そこに制約条件をチェックしながら接続情報に従ってセグメントを接続していく.
- (4) 未割り当てセグメントが無くなるまで、(3)を繰り返す.

ここで、探索の効率化のために前処理を行なって 2 つのテーブル、セグメントテーブル (上記 (1) で使用)、便接続ベーステーブルを作成した. さらに、便接続ベーステーブルをもとに、セグメント動的接続テーブルを (3) の実行時に作成して利用した. 詳しくは後述する.

6.2 染色体表現

染色体は、上記の (1), (2), (3) に対応して、3 つの部分から構成し、これらが配列として順に並んだものを用いている.

図 4 は、1 個体の染色体の説明図である.

●セグメント遺伝子

- 各航空機の一日の便についていくつかのセグメントに分けるその分け方が指定される.
- 航空機の数だけ用意される.
- 対立遺伝子は、セグメントテーブルの制約を満たした切断のインデックスが入る (6.4 節参照).

●ペアリング順遺伝子

- ペアリング順リストを作る際に用いる.
- ペアリング対象の便数だけ用意される.
- 対立遺伝子は、0~4095 までの整数で、その大小によりこの便を先頭とするセグメントのソートに用いる.

●接続遺伝子

- ペアリング対象の個々の便に、複数の接続候補が列挙されている (接続候補遺伝子の多重化).
- 不定長、かつ、基地への帰還もコーディングされる.
- 便接続ベーステーブルのインデックスが入る.

6.3 遺伝操作と世代交代

ベースとなる GA の枠組として、集団の多様性を保ちながら、良解を得るため、GA の世代交代モデルとして MGG [佐藤 96] (Minimum generation Gap) を採用した. これは、家族 (親 2 個体、子 2 個体) から 2 個体を残

C.N.	羽田	新潟	羽田	福岡	羽田
0000	()	()	()	()	X
0001	()	()	()	()	0
0100	()	()	()	()	0
0110	()	()	()	()	0
1000	()	()	()	()	0
1001	()	()	()	()	X
1100	()	()	()	()	X
1110	()	()	()	()	X
1111	()	()	()	()	X

図 5 セグメントテーブル

す手法である.

評価関数はパターン数を用い、最小化問題として扱う. 選択は、最良のものを 1 個体、他の評価基準で 1 個体残すようにしている. GA のオペレータとしては、単純な 2 点交叉と、対立遺伝子の中からランダムに選択される突然変異を採用した.

GA 起動時には、各染色体は、セグメントテーブルやセグメント動的接続テーブル等を用いた初期解が設定される.

6.4 セグメントとセグメントテーブル

個々の便単位でペアリングを行なう場合、組合せの数が膨大になる. また、一日の乗務の途中で他の航空機に乘換える場合、航空機の点検等に余分な時間が必要となり、効率が悪いペアリングになりがちである. そこで “1 つの航空機に割り当てられた便をなるべく乗り続けたほうが良い” という経験則により、ペアリングの単位を個々の便ではなく、同一航空機で運航する連続した複数の便を 1 つの塊とすることにした. これをセグメントと呼ぶ.

セグメント内の便はまとめてペアリングされ、セグメントの最後の便から他のセグメントの最初の便への接続が有効となる. セグメントの途中からの接続と、セグメントの途中への接続は、探索する必要が無くなるため、セグメントの採用は探索空間の削減に大きな効果がある.

§1 セグメントテーブル

セグメントテーブルは、各航空機毎に作成され、個々の切断の他便への乗換え可能性、乗務回数、勤務時間等の制約条件を満たしたセグメントの組合せを全て列挙したテーブルである.

図 5 はセグメントテーブルの例であり、“乗務回数は 3 回まで”、“新潟で乗換えてはならない”、という制約があると仮定する. ここでは、切断可能な部分が 3 つあるため、 $2^3 = 8$ 通り (CN : Cutting Number) の切断が可能であり、図では “()” で囲んだ部分が 1 つのセグメントである. 全ての切断候補の中で、4 便連続乗務の CN 000 及び、新潟で切断した CN 100,110,111 がこの制約に触れ、テーブルから外れる (×印).

さらに、許された組合せのそれぞれに、選ばれやすさの数値を付加しており、まず長いセグメントを組み合わせた切断に高い基礎ポイントを与えている. さらに各空港での切断について、宿泊不可空港データや便の接続状況などから負の値も含むポイントが計算され、この切断

された空港の個々のポイントと前述の基礎ポイントとの和が、その切断の組合せの選ばれやすさの数値として設定される。このセグメントテーブルは、初期解作成と GA の突然変異の際に確率テーブルとして使用される。

デコード時のセグメント化の処理では、染色体のセグメント遺伝子の指定によってテーブルから 1 つのセグメントの組合せが選ばれる。

6.5 ペアリング

§1 非定期便の処理

ペアリング時の非定期便の処理についてははじめに説明する。セグメントは運航情報としてスケジュール期間の各日の運航日をビット列として持っている。例えば定期便は、11111...11 のような運航日情報を持ち、奇数日の運航は、10101... のような運航日情報を持つ。

ペアリングの際には、複数の運航日情報のビット列の AND がそのペアリングの乗務日となる。数日に渡るペアリングでは、日数経過分のシフトがされた乗務日が設定される。

図 8 の CP1 の Seg.1, Seg.2 の定期便が Seg.3 の非定期便に接続する例で、AND 処理をすると以下のようになる。(表 2 も参照)

セグメントの運航日		ペアリングの乗務日	
Seg.1	111111111...	010010010...	1 日目
Seg.2	111111111...	001001001...	2 日目
Seg.3	100100100...	000100100...	3 日目

スケジュール期間の両端では、乗務日のズレのために不整合が生じるが、これはペアリング時に適宜解消する。(例では、Seg.3(3日目)の第1日目の運航は、スケジュール期間内に対応する乗り継ぎがない。)

§2 便接続ベーステーブル

個々の便は、基地空港との接続のため、予め全ての基地空港からの出発 (deadhead を含む)、全ての基地空港への帰還 (deadhead を含む) がチェックされる。さらに、以下の接続可能な全ての便をリストアップし、これらの結果から便接続ベーステーブルを作成する。

- 当日の同一空港での他の航空機への乗り継ぎ (同一航空機への乗り継ぎは無条件に可能である。) 可能な便
- 翌日の便 (宿泊は最後の便の到着空港) への乗り継ぎ可能な便
- 当日 deadhead で他空港へ移動し (宿泊は deadhead の到着空港), その空港からの乗り継ぎ可能な便
- 翌日 deadhead で移動し (宿泊は最後の便の到着空港), その空港からの乗り継ぎ可能な便

接続遺伝子は、このテーブルのインデックスを指し表す。便接続ベーステーブルでは、個々の接続に対して、その好ましさも数値的に取り入れており、図 6 は便接続ベーステーブルのある便からの接続の好ましさを重み付き矢

図 6 重み付き接続

図 7 セグメント動的接続テーブルとその再構成

印で示したものである。ここでは、便が到着してから所定の休息時間以上で、なるべく短い時間に接続された方が好ましいので、その様子を図示してある。この重み付けには、この時間間隔の他に接続先の便への他の接続便数などから算出した値が用いられる。この図では、deadhead 無しの接続を示しているが、deadhead も入れれば更に可能な接続が増える。

ここで設定する便接続ベーステーブルは、乗り継ぎ可能性についての前チェックであり、重み付きの値の計算はデータ読み込み直後に一度だけ行なわれる。

§3 セグメント動的接続テーブルの初期化と再構成

このセグメント動的接続テーブルの初期化は、各染色体のデコード時のセグメントが決まった後に行なわれる。各セグメントの、最後の便の接続がそのセグメントからの接続となり、そこから他のセグメントの先頭を指している有効な接続が、便接続ベーステーブルの接続の重みデータと一緒にまとめられる。この重みを用いて、特定のセグメントから接続するのが望ましいセグメントへの接続には大きく、そのような必要性の少ない接続には小さくなるような確率が、セグメント動的接続テーブルに設定される。この重みは、全体が 1.0 になるよう規格化され、確率テーブルとして用いられる。

セグメントの接続は、接続遺伝子の指定する値に従って選択される。染色体の複数ある接続候補を使いきった後は、このセグメント動的接続テーブルの確率に従った接続が行なわれる。ここで得られた新たな接続は、接続遺伝子に付加され、獲得形質を後世に伝えるラマルク遺伝的な仕組みも取り入れている。

ある選択肢がペアリングされて無くなると、残った選択肢からセグメント動的接続テーブルが再構成される。図 7 は、セグメント動的接続テーブルを再構成する様子

図 8 非定期便の波及抑制

を示したものである。ペアリング時に、セグメント A の全ての運航日が確定したペアリングに組み込まれると、それ後のそのセグメント A への接続は許されないため、セグメント A への接続は削除され、セグメント動的接続テーブルが図の様に再構成される。再構成では、残った接続確率の総和が 1.0 になるよう規格化される。

6.6 ヒューリスティクスの導入

探索に効果があったヒューリスティクスを述べる。

§1 確率的再ペアリング

局所的には制約違反ではない (セグメントの制約違反は前チェックで除いてある) が、前後の日の接続の状況によって、制約違反になる場合がある。

例えば $n+1$ 日目の勤務時間がある制限を越えた場合、 n 日目の勤務終了から $n+1$ 日目の勤務開始までの時間をより開けなければならない。しかし、制限を満たすかどうかは、実際に $n+1$ 日目のスケジューリングを行なって見なければ判らない。

このような制約違反があった場合は、最大 2 日目まで遡ってペアリングがやりなおされる。何日間遡るかは、様々な確率で選択される。

§2 非定期便の波及抑制

非定期便の波及を抑制するために以下のような処理を導入した。

スケジューラは、セグメントをいくつか制約違反無く接続できれば、そのペアリングを確定して解として取り込む。定期便を繋いでいっても、一部のセグメントが非定期便であれば、このペアリングは非定期なものとなり、定期便の非定期化が生じる。これを放置すると非定期化の波及が生じるため、定期便が非定期化したことをトリガーとして、非定期便の波及抑制処理が起動され、直前のペアリングの結果非定期化した定期便のセグメントについて、バックトラック的ペアリングが行なわれる。

図 8 は、非定期便の波及抑制処理を示した図である。図で、Seg.1、Seg.2(太い実線)は定期便のセグメントで“111111...”のような連日の運航パターンを持つ。Seg.3(太い点線)は非定期便のセグメントで“100100100...”のような 3 日に 1 度の運航パターンを持つとする。

- (1) スケジューラが、Seg.1 → Seg.2 → Seg.3 のペアリングを行なう (CP-1)。
- (2) CP-1 は 3 日に 1 度の運航なので、残りの日の Seg.1、Seg.2 の運航は非定期便とみなして、別のペ

アリングを作る必要がある。

- (3) ここで非定期便の波及抑制機能が起動される。

- 残りの Seg.1 と Seg.2 を繋ぎ、この後につながる、これを補完するセグメントが探される。
- なければ図のように deadhead で基地への帰還をペアリングする (CP-2)。

このような仕組みを用意することで、非定期便の波及を抑制することが可能となる。また可能な限り、直前のペアリングと同じセグメントを接続することで、解の質を損なうことなく CP の結果を人間が見て理解やすくなる。これはシステムを実際に使用する上では非常に重要なことである。

7. デコード手法の詳細

本システムで用いたデコード手法をまとめておく。

- (1) セグメント遺伝子に従って、各航空機の便がセグメントに分けられる。
- (2) セグメント化の結果と、ペアリング順遺伝子から、ペアリング順リストが作られる。
- (3) セグメント化の結果から、セグメント動的接続テーブルの初期化が行なわれる。
- (4) ペアリング順リストの先頭のセグメントから、ペアリングが開始される。
 - ペアリングの開始セグメントでは、ある規則で出発基地が決められる。
- (5) 次のセグメントを繋いでいく処理を、出発基地に帰還するまで続ける。
 - ペアリングは、割り当てられたセグメントの運航日を考慮して行なわれる。ペアリングの乗務日は、日のズレ入りの AND がとれた運航日となる。
 - ペアリングの終端セグメントの接続情報に従って繋いでいく。
 - 接続遺伝子上に繋がるセグメントが無ければ、セグメント動的接続テーブルから繋がるセグメントを確率的に選択して接続していく。成功した接続は、接続遺伝子に付加され、次世代に受け継がれる。
 - 制約チェックを行ない、違反が無ければ次に進む。途中で制約違反を生じたら、確率的再ペアリングを試みる。
 - 最終セグメントを数種類の条件で決定し、出発基地へ帰還させる。
- (6) 制約違反なく基地に帰還したら、そのペアリングを確定する。
 - 確定したペアリングは解の一部となる。
 - 残った各セグメントから確定したペアリングの運航日を取り除く。運航日が無くなれば、そのセグメント自体が取り除かれる。

- (7) (6) で確定したセグメントの残りの状況を判断し、場合によっては非定期便の波及抑制処理を起動する。
- (8) 未割当セグメントが残っていれば、(4) へ。
- (9) 終了

8. 評価

ここで扱っている CP は、多くの非定期便を含む実問題である。従来手法の枠組みでは定期便しか扱えず、この問題を事実上取り扱うことができないため、本稿では、航空会社の専門家によって作成され実運用された解を対照させて検討していくこととする。

試行データとして、ある航空会社より提供を受けた繁忙期や閑散期を含む 2 年分の 12 種類のダイヤデータを用いた。機種は A～E の 5 機種あり、合計で 60 種について CP を試行した。試行データ 1 においての便数は 39 から 386 で、それを表の 2 行目に示したが、各データ間の各機種の便数 (問題規模) の差異は数%しかなく、他のデータは省略した。スケジュール期間は 60～61 日であり、非定期便は全体の 3 割ほどを占めている。

本システムは C++ 言語で記述し、CPU 296MHz の UNIX ワークステーションを用いて開発した。GA のパラメータは、集団サイズ 1024、突然変異率 0.005/遺伝子座 で試行した。最大規模 (機種 E) の問題のパターン数は約 100 であり、専門家が解くのに要する時間は 1 週間程度である。なお、1 パターンは 1ヶ月当たり 30 人日に相当し、乗務員数に換算すると 2～3 人に相当する。

表 1 に、GA の解と実運用の解の差、すなわちパターン数の差を示す。表の値が正の部分は GA 解の方が良い結果であったことを意味する。試行において、同一ケースに関しての試行間のバラツキは、中小の問題ではほとんどなく、大きな問題 (機種 E) で 1 パターン程度であった。

表 1 において、差が極端に大きいケースが見られず、また、GA 解の方が優っているケースは全体の 26/60、劣っているケースは 14/60 であり、総合的には実運用解と同等と考えられる。比較的小規模な問題 (機種 A～C) のケースに較べて、大規模な問題 (機種 E) のケースで、GA がやや及ばない傾向がみられる。問題が中小規模のもので、専門家がそれを実際に解くには、数時間～数日かかっており、これらの規模の問題である程度信頼できる結果が得られれば、システム化の意義があると言える。

GA が専門家に優るのは、GA が良い部分解を見つけてパターン数を下げているからである。例えば、専門家がその先入観や見落としのため想定しなかった空港でのセグメントの切断と接続を行なうことで、パターン数が下がったケースがあり、ある部分では専門家よりも広い探索をしているといえる。

一方、GA 解が及ばないケース (特に機種 E) では、専門家はそのノウハウを駆使して乗換を行い、パターン数を下げている。例えば、経験則ではなるべく長いセグ

メントを用いる方が良いとされ、本システムでは長いセグメントに高い評価を与えることでこれを実装しているが、専門家は場合によっては短いセグメントをうまく繋ぎ、トリッキーな航空機の乗換を行なうことで、パターン数を下げている。本システムでは、その乗換を固定部として指定することで、求解性能を向上させることができ、表 1 で GA が及ばなかったケースでも専門家と同等のパターン数が得られた。但し、表 1 にはこの結果は入っていない。このノウハウの一部分を明示的にインプリメントしたところ、ある程度有効な結果は見られたが、無条件に専門家より優れたスケジューラを構成するには至っていない。

なお、GA の試行は 2 時間で打ち切った。これは航空会社のシステム化の要請からくる制限である。解を得るまでの時間は、機種 A のケースで約 5 分、機種 E のケースでおよそ 1 時間であり、専門家による作成の約 1 週間に比べて大幅に短い。

前述のように他システムとの比較を行なうことはできないが、本システムでは数々の制約条件を近似無しで取り入れており、解はそのまま実行可能となっている。一方、IP/LP で実用規模の問題を定式化する場合、多くの近似を取り入れて変数を減らすモデリングがなされるのが常であり、この点でも本システムのアプローチ手法はより有効であるといえる。

9. まとめ

本稿では、fully dated な CP を、GA を用いて解くシステムを示した。本システムは、経験を積んだ専門家に頼った CP を自動化することができ、また従来手法では扱いにくい、現場で要請されるさまざまな条件を簡略化することなく考慮した、実用的な fully dated な CP スケジューラとしては初のもと思われる。

fully dated な取り扱いをしたことに起因して探索空間が非常に大きくなるため、確率的なデコード、遺伝子の多重化、を特長とした GA の枠組みを考案し、問題の特性に着目したいろいろな技法と組み合わせることにより探索の効率化・高速化を図った。

本手法を 2 年分のデータを用いて試行し、その結果と航空会社専門家によるペアリング結果を比較した。一部分専門家に優る解もあるが及ばない解もあって改善の余地が残されているものの、制約を簡略化しないで作成が自動化されたこと、作成期間が短縮されたことなどの点から、実効あるシステムであると考えられる。

本手法は CP という実問題を対象として開発されたが、本手法の考え方や枠組みは、CP だけでなく、勤務スケジュールリングなどの他の実問題への応用が可能と思われる。これについては今後の課題としたい。

表 1 テスト結果

	A	B	C	D	E
\ (便数)	39	47	67	122	363
1	1.0	0.8	1.0	0.6	0.7
2	0.0	-1.0	0.0	0.0	0.0
3	0.0	0.0	0.0	-0.1	-1.3
4	1.0	0.0	-0.2	0.1	0.1
5	0.0	0.8	0.1	1.0	0.6
6	0.0	0.3	0.0	-1.5	-1.7
7	0.0	-0.7	1.0	0.6	1.0
8	0.0	0.2	1.0	0.0	-1.2
9	0.3	0.0	1.0	-0.6	-2.1
10	0.0	-1.0	0.0	0.1	0.0
11	1.0	0.0	0.1	1.3	-2.4
12	0.0	-0.4	1.8	-0.4	1.3

Pairing — A
 TYO base 0.870968 人日
 — TS 4 265 625
 B767 HND 0855 OIT 1025
 A300 OIT 1150 SDJ 1850
 — TS 2 145 420
 A300 SDJ 1550 FKS 1850
 # FKS 1920 HND 2150

Pairing — B
 TYO base 1.12903 人日
 — TS 4 320 625
 B767 HND 0855 OIT 1025
 A300 OIT 1150 SDJ 1850
 — TS 2 145 420
 A300 SDJ 1550 FKS 1850
 # FKS 1920 HND 2150

表 2 非定期便を含むスケジューリング例

運航日/乗務日 → (H)
 1000011100001110000111000011100 ← Pairing-A の運航日
 1000011100001110000111000011100 ← 第 1 日目の乗務日
 ← deadhead
 1000011100001110000111000011100 ← 非定期便のセグメント
 1100001110000111000011100001110 ← 第 2 日目の乗務日
 1111111111111111111111111111111 ← 定期便セグメントへの接続
 ← deadhead(railway)

運航日/乗務日 → (H)
 0111100011110001111000111100011 ← Pairing-B の運航日
 0111100011110001111000111100011 ← 第 1 日目の乗務日
 ← deadhead
 0111100011110001111000111100011 ← 非定期便のセグメント
 0011100011110001111000111100011 ← 第 2 日目の乗務日
 0011100011110001111000111100011 ← Pairing-A の残りの運航日
 ← deadhead(railway)

◇ 参 考 文 献 ◇

- [Andersson 97] Andersson, E., E.Housos, , N.Kohl, , and D.Wedelin, : *OR in the Airline Industry, chapter Crew Pairing Optimization*, Kluwer Academic Publishers (1997).
 [Matsumoto 99] Matsumoto, S., Sato, M., Teramoto, Y., and Adachi, N.: A Genetic Approach to a Practical Crew Pairing System, in *Proceedings of APIEMS '99*, pp. 327-330 (1999).
 [Tajima 97] Tajima, A. and Misono, S.: Solving Airline Crew Scheduling Problem with Many Irregular Flights, in *ISMP97 International Symposium on Mathematical Programming* (1997).
 [Vance 97] Vance, P. H., Barnhart, C., Johnson, E. L., and Nemhauser, G. L.: Airline Crew Scheduling: A New Formulation and Decomposition Algorithm, *Operation Research*, Vol. 45, No. 2, pp. 188-200 (1997).
 [佐藤 96] 佐藤, 小野, 小林: 遺伝的アルゴリズムにおける世代交代モデルの提案と評価, 人工知能学会誌, Vol. 12, No. 5, pp. 734-744 (1996).
 [寺本 99] 寺本, 佐藤, 松本, 安達: GA による航空乗務パターンスケジューリング, 生産スケジューリング・シンポジウム'99 講演論文集, pp. 47-52 (1999).
 [松本 00] 松本, 佐藤, 寺本, 安達: 複雑問題への遺伝的アプローチ (事例紹介: ミニ特集, ブレークスルーとしてのソフトコンピューティング), 計測と制御, Vol. 39, pp. 194-197 (2000).

[担当委員: 木下哲男]

2000 年 10 月 12 日 受理

◇ 付 録 ◇

A. ペアリング例

表 2 は, 非定期便と定期便のスケジューリング例の特徴的な一部を取り出したもので, 30 日間をまとめて扱っている。

Pairing-A では, 羽田 (HND) から大分 (OIT) に deadhead の後, 非定期便のセグメントに乗務し, 翌日の仙台 (SDJ) から福島 (FKS) の定期便のセグメントに接続後, deadhead(電車) で帰還している。

2 日目の乗務日が 1 日目の乗務日と 1 日ずれていることに注意されたい。この際, 2 日目のセグメントは, 2 日目乗務日が 1 の日しか接続されておらず, 残りが非定期便となる。

Pairing-B では, Pairing-A と相補的な非定期便のセグメントが, Pairing-A と同様に割当られ, 2 日目には Pairing-A で運航日が残り, 非定期便になったセグメントに接続されている。

Pairing-A と Pairing-B のセグメントの内容は其々 OIT → NGO → OIT → NGO → SDJ, OIT → OKA → OIT → NGO → SDJ となっており, 冒頭の 2 便が日によって行き先が異なってい

る。(2 日目のセグメントは, SDJ → CTS → FKS である。)

尚, 各日の乗務日の行の TS の後の数字は, 乗務便数, 乗務時間 (分), 勤務時間 (分) を表示したものである。

—— 著 者 紹 介 ——

佐藤 眞木彦(正会員)

1982 年 東北大学理学部物理学科卒業, 1984 年 理学部原子核理学博士課程前期修了, 1986 年 理学部原子核理学博士課程後期中退, 同年 富士通株式会社入社。プログラム相談システム/暗号/ニューラルネットワーク/GA システム/バイオインフォマティクス等の研究開発に従事。人工知能学会会員

松本 俊二(正会員)

1983 年 東京大学工学部物理工学科卒業, 同年富士通株式会社入社, ルールベースシステム/事例ベース/GA システム/自然言語応答システム等の開発研究に従事。人工知能学会, 情報処理学会会員

寺本 陽彦

1994 年 東北大学理学部物理学科卒業, 1996 年 東北大学理学研究科物理学専攻修士課程修了, 同年富士通株式会社入社, スケジューリングシステム/自然言語応答システム等の開発研究に従事。

安達 統衛

1975 年 東京工業大学工学部制御工学科卒業, 1977 年 東京工業大学大学院理工学研究科修士課程修了, 同年富士通株式会社入社, 計算機科学/生物情報/進化アルゴリズム等の研究開発に従事。情報処理学会, 電子情報通信学会, 日本物理学会会員