

Levy Flight に基づく進化計算によるレイアウト問題の  
解法

2018年 9月

趙 冬青

Levy Flight に基づく進化計算によるレイアウト問題の  
解法

趙 冬青

システム情報工学研究科

筑波大学

2018年 9月

## 要旨

本論文では、Levy Flight (LF) に基づく進化計算手法を提案する。また、施設レイアウト問題を対象として、適応度が高い複数の異なる近似解を得るため、複数のデコード化を有する島モデルも提案する。

施設レイアウト問題 (FLP) とは、施設内に配置物 (設備、機械、備品、家具、部署、職場など) を仕事の効率や作業時間が最適になるように配置する探索問題である。適切な配置は仕事の運営費を 50%削減できるといわれており、古くから FLP に対する研究が多数行われている。FLP の計算複雑さは NP 困難であることが知られている。このため、問題を分類した上でメタヒューリスティックにより近似解を求めるというアプローチが主流となっている。

本論文では、複数の長方形の施設内に大きさが不揃いな複数の長方形の配置物を目的関数値が最小となるように配置する問題を扱う。この問題の解法としては、遺伝的アルゴリズム (GA)、焼き鈍し法 (SA)、アントコロニー最適化法 (ACO)、タブーサーチ (TS)、粒子群最適化法 (PSO)、カオス力学的モデルを用いる方法などの近似解法が提案されている。この問題の解法としては、メタヒューリスティックと MIP などをハイブリッド化する方法が広く研究されている。

最近、膨大な探索空間の中でランダムに存在する解を効率的に探索する方法として、LF が注目されている。LF に基づく探索方法は、通常の Random Walk (RW) に基づく探索よりも効果的であることが知られている。本研究では、まず FLP のベンチマーク問題を対象として、通常の進化アルゴリズム (EA) に LF を導入することにより、GA、ACO、カオス力学的モデルよりも目的関数値の小さい解を発見できることを示した。

次に、現実的な FLP の一例として、住宅の複数の部屋に大きさが不揃いな複数の家具を配置する「室内レイアウト問題」を取り上げる。室内レイアウト問題とは、家具を与えられた領域内に制約条件を満たすように配置する問題である。また、作成されたレイアウトはユーザの希望を反映していることが望ましい。従来、対話型の遺伝的アルゴリズム (IGA) などが提案されているが、制約充足度とユーザの好みを両立できていない。そこで、本論文では、ユーザの嗜好・快適性・利便性を考慮して進化計算における集団を複数の部分集団 (島) に分割し、部分集団毎に異なるデコード化法を採用することにより、集団全体の多様性の維持を図る。最終世代の個体にばらつきを持たせることを主な目的としている。提案手法と通常の GA との比較実験により、本手法の有効性を示す。

# 目次

第一章 序論.....	1
1.1 研究の背景.....	1
1.2 研究の目的.....	2
1.3 本論文の構成.....	3
第二章 研究分野の概要.....	4
2.1 施設レイアウト問題.....	4
2.1.1 施設レイアウト問題概要.....	4
2.1.2 施設レイアウト問題の定式化モデル.....	6
2.2 室内レイアウト問題.....	11
2.3 施設レイアウト問題の解法.....	13
2.4 Levy Flight.....	21
第三章 ベンチマーク問題への適用.....	25
3.1 提案手法.....	25
3.1.1 コード化.....	25
3.1.2 突然変異.....	26
3.1.3 Levy Flight による解の更新方法.....	31
3.1.4 アルゴリズム.....	32
3.2 ベンチマーク問題の評価実験.....	33
3.2.1 実験方法.....	33
3.2.2 EA における突然変異の回数.....	44
3.2.3 交換回数の計算に用いた分布関数の比較.....	45
3.2.4 Levy Flight の有効性.....	49
3.2.5 従来手法との比較.....	51
3.3 まとめ.....	53
第四章 室内レイアウト問題への適用.....	55
4.1 提案手法.....	55

4.1.1	コード化 .....	55
4.1.2	デコード化.....	56
4.1.3	適応度関数.....	62
4.1.4	遺伝の操作.....	66
4.1.5	解の更新方法 .....	67
4.1.6	アルゴリズム .....	68
4.2	室内レイアウト問題の評価実験 .....	69
4.2.1	実験方法 .....	69
4.2.2	室内レイアウト問題の設計.....	70
4.2.3	適応度の評価 .....	74
4.2.4	多様性の評価 .....	76
4.3	まとめ .....	78
第五章	結論 .....	79
5.1	研究の結論 .....	79
5.1.1	第三章「ベンチマーク問題への適用」の結論.....	79
5.1.2	第四章「室内レイアウト問題への適用」の結論 .....	79
5.2	本研究の工学的意義.....	80
5.3	今後の課題.....	81
謝辞	.....	82
参考文献	.....	83

# 第一章 序論

## 1.1 研究の背景

施設レイアウト問題 (FLP) とは、施設内に配置物 (設備、機械、備品、家具、部署、職場など) を仕事の効率や作業時間が最適になるように配置する探索問題である。適切な配置は仕事の運営費を 50%削減できるといわれており、古くから FLP に対する研究が多数行われている。FLP の計算複雑さは NP 困難であることが知られている [Drira 07]。これらの問題はそれぞれに専門的な知識が必要であり、その多くが計算機上で表現することが困難である。また、様々な制約条件を考慮しなければならないことから、膨大な解候補の組み合わせから望ましい解を探索する必要があり、膨大な計算時間と記憶容量を要する。全解探索が理論的に不可能な問題や、経路探索のように即時性を求められる問題に対して、高速に準最適解を探索する解法が求められてきた。さらに、それぞれの事例ごとに特徴が異なるため有効な解法がないことや、実面的には厳密に表現できない曖昧な制約条件があることから、問題解決は人手に頼らなければならないのが現状である[清水 01]。

近年、研究者たちはこの問題に対して定式化方法と近似解法の組み合わせる方法を改良している。定式化方法としては、離散方法と連続方法を分けている。レイアウト問題に関する研究で多く利用したのは QAP (Quadratic Assignment Problem) 法[Kusiak 87]は離散表現方法の一つである。この方法は平面図を小さなブロックで区切り、配置物はこのブロックを基準に配置する方法である。ブロックの大きさはレイアウト (配置図) の生成に大きな影響がある。他のはまだ CRAFT(Computerized-Relative Allocation of Facilities Technique)[ Armour 63]、グラフ理論モデル[Hassan 87]などの手法が提案されている。以上の手法は主に小規模な問題に解決できる。近年、連続表現方法として、配置物の位置を実数値として、全ての制約条件を数式で表現する MIP (Mixed Integer Programing) モデル[Montreuil 90]が主流となっている。この方法は制約少ない、表現できるレイアウトが多いという利点がある。また、メタヒューリスティックと MIP をハイブリッド化する方法が広く研究されている。

この問題の近似解法としては、遺伝的アルゴリズム (GA) [Gonçalves 15][Liu 07]、焼き鈍し法 (SA) [Tam 92]、アントコロニー最適化法 (ACO) [Komarudin 10] [Guan 16]、タブーサーチ (TS) [Scholz 09]、粒子群最適化法 (PSO) [Konaka 11]、カオス力学的モデルを用いる方法[大森 13] などの解法が提案されている。これらの研究においては、ベンチマーク問題[Komarudin 10] (主な制約条件としては、①配置物は施設からはみ出さない、②配置物は重なってはいけない、という単純なものである。) を対象として、各配置物間の物流コストや距離の全合計 (目的関数) がなるべく小さくなる配置を求めることを主な目的としている。最近、膨大な探索空間の中でランダムに存在する解を効率的に探索する方法として、LF (Levy Flight) が注目されている。今まで Levy 分布は、物理学、生物学、金融、統計、経済の分野の最適化問題に応用されていて LF に基づく探索方法は、通常の Random Walk (RW) に基づく探索よりも効果的であることが知られている[Ali 15] [Viswanathan 08]。

現実なレイアウト問題を考えると、目的関数も制約条件も複雑であり、施設を利用する人々の個性、直感、快適性、利便性などを考慮して問題をすべて定式化することは難しいと思われる。探索にユーザの好みを反映させる方法として、対話型遺伝的アルゴリズム (IGA) [是永 00]が広く用いられている。しかし、IGA を本問題に適用すると探索空間が広いため、探索が収束するまでに多数の世代ステップが必要になり、ユーザの対話回数が多くなりすぎる。ユーザとの対話を数世代毎にして回数を減らすと、収束時に集団の多様性が低くなり、ユーザの選択肢が少なくなる。この結果、ユーザの好みを反映したレイアウトを求めることができない可能性がある。

## 1.2 研究の目的

実用的かつ効率的なレイアウトシステムを開発するため、本研究の提案手法は 2 つの部分に分けている。第一部分は Levy Flight (LF) に基づく進化アルゴリズム (EA) を用いて、探索効率の向上を求める。第二部分は複数のデコード化を有する島モデルを提案して多様なレイアウトを生成することを期待する。この二つ部分により、探索性能の向上かつレイアウトの多様性の維持を目指すことが本研究の目的です。

本論文では、Levy Flight の有効性を確認するため、まず多くの従来研究で取り扱われている FLP のベンチマーク問題を対象として、通常の進化アルゴリズム (EA) に LF を導入することにより、GA、ACO、カオス力学的モデルよりも目的関数値の小さい解を発見できることを示した。EA は GA、ACO などのアルゴリズムより、操作はただ突然変異つまり最も単純なアルゴリズムなので、LF の探索効果をはっきり見える。従って、EA は LF の有効性を証明する最適なアルゴリズムと考える。

次に、現実的な FLP の一例として、住宅の複数の部屋に大きさが不揃いな複数の家具を配置する「室内レイアウト問題」を取り上げる。実用するため、市販されている家具と住宅のカタログを参考にして部屋の構造、および使用した家具の属性を設定した。また、探索を通じて生成されたレイアウトがユーザになるべく多い選択肢を提供する。つまり、適応度が高い複数の異なる近似解を得るのは本研究の第二目的である。従って、Levy Flight (LF) に基づく進化計算手法と複数のデコード化を有する島モデルを提案する。そして、どの解を採用するかはユーザが決定するという戦略をとる。ユーザの嗜好を探索に取り入れる方法としては、対話型進化計算[是永 00]が広く用いられているが、本問題は探索空間が膨大であるため解候補の集団サイズが大きくなり、対話型の計算には向いていないと考える。

### 1.3 本論文の構成

本論文は全五章からなり、第一章は序論、第五章は結論とした。第二章では本研究の対象問題である施設レイアウト問題のベンチマーク問題と実問題の一例「室内レイアウト問題」並びに Levy Flight の基礎事項について述べる。第三章ではベンチマーク問題を通じて提案アルゴリズムの有効性を示す。第四章では、現実な施設レイアウト問題の例への適用の基礎事項を述べる。ユーザの好みを反映する制約条件の設定、異なるデコード化とレイアウトの生成方法などを詳しく説明する。実験結果から、提案手法の特徴と利点について考察する。



## 第二章 研究分野の概要

### 2.1 施設レイアウト問題

#### 2.1.1 施設レイアウト問題概要

施設レイアウト問題 (Facility Layout Problem: FLP) は、プリント基板の設計、病院・学校・空港などのレイアウト、機器内部品の配置、倉庫詰め問題、水力タービンの設計など、幅広い分野に見られる組み合わせ最適化問題である [Singh 06]。FLP の計算複雑さは、NP 困難であることが知られており、施設内の通路の種類 (列状通路、ループ状通路、自由通路)、施設の数 (部屋数、階数)、配置物の形状 (同一形状 / 不揃い、長方形 / 凸凹形) などで問題を分類した上で解法が研究されている [Drira 07]。適切なレイアウトは作業時間の短縮、見通しが良くなり、人力運搬が減ることなどの利点がある。この問題は環境の特徴、問題の制約、解決の方法などの要因に依存する。また、すべての条件：能率がよく運営でできること、弾力性、面積が効率良く使われること、運搬しやすいことを十分満足するのは難しい。レイアウト設計の初期段階、施設の形状と規模を考慮し、生産用の材料の適切な場所への配送、運搬装置方法 (施設の通路仕様) の選択も注意しなければならない [遠藤 71]。

施設内の通路の種類は図 2.1~図 2.3 を示すような列状通路、ループ状通路、自由通路がある。列状通路、ループ状通路は部品ごとの作業工程の順に職場を並べる型で配置する。自由通路は同じ作業方法を用いる職場を集めて配置する。職場の場所は **backtracking** と **bypassing** の方法によって施設の配置順番を変える [Drira 07]。また、配置の始点位置と終点位置を決める必要がある。図においては、番号を付いている円形は配置物である。施設の数から分類すると一つ部屋から複数部屋、一階の建造物から多層ビルなども対象となる。図 2.4~図 2.5 は施設の形状の例である。



図 2.1 列状通路

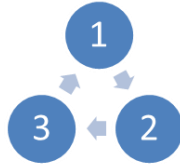


図 2.2 ループ状通路

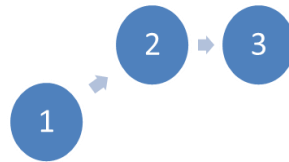


図 2.3 自由通路

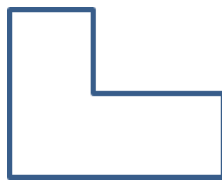


図 2.4 凸凹形の施設



図 2.5 長方形の施設

全体レイアウトの確定は建屋の形、敷地への配置及び建屋内の職場分類や通路種類の選択、出入口の位置などを左右される。実施後変更は困難なので、工費の大部はこの段階で決める。

### 2.1.2 施設レイアウト問題の定式化モデル

複雑なレイアウト問題を応用するため、色々な改良モデルを提案した。現実なレイアウト問題特に工場の生産ライン、オフィスの部署などの配置問題を設計する時、人の移動距離を減ることと運搬作業を軽減するため、各部屋の長さや幅また場所の決定が難しい。従って、一つの長方形の施設内に面積が不揃いな長方形の配置物また配置する位置と縦横比（面積は固定）を決定する問題をよく研究されている。

代表的な定式化モデルとしては、QAP（Quadratic Assignment Problem）として扱うモデル[Kusiak 87]、配置物をグラフのノードとして表現するグラフ理論モデル[Hassan 87]、MIP（Mixed Integer Programming）で扱えるように配置物の位置を実数値として、全ての制約条件を数式で表現する MIP モデル[Montreuil 90]などが挙げられる。表 2.1 は各モデルの比較を示す。

表 2.1 各モデルの比較

手法	特徴	職場の特徴
QAP (Quadratic Assignment Problem)	格子状の配置領域に職場を配置する	面積固定、主に正方形
グラフ理論モデル	重み付きグラフの与えられた2つノード	面積、形状固定
MIP (Mixed Integer Programming) (1990)	職場たちが重ならないように、たくさん二値変数を使う、自由平面で、9 職場まで	面積、形状自由
MH(Meta-Heuristics)+MIP をハイブリッド(2005)	35 職場、大規模問題に適用	MIP と同じ

### (1) QAP (Quadratic Assignment Problem)

この分野よく使っている QAP (Quadratic Assignment Problem) は平面図を小さなブロックで区切り、配置物はこのブロックを基準に配置する手法である。この手法はブロックの大きさはレイアウトつまり配置図の生成に大きな影響がある。表現できないレイアウトが存在する。

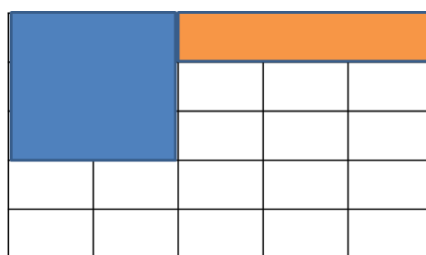


図 2.6 QAP の配置図

### (2) グラフ理論モデル

グラフ理論モデルは重み付きグラフの与えられた 2 つノード間を結ぶ経路の中で、重みが最小の配置順を求めるという方法である。このモデルは主に小規模問題に適用できる[Hassan 87]。

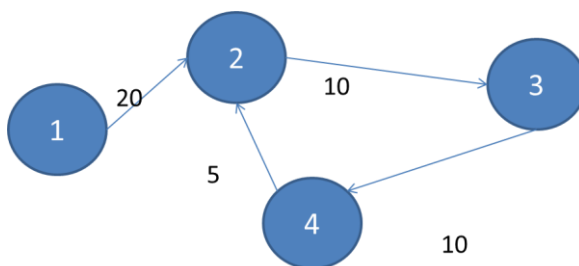


図 2.7 グラフモデル

### (3) MIP (Mixed Integer Programming)

近年、配置物の位置を実数値として、全ての制約条件を数式で表現する MIP モデルが主流となっている。非常に柔軟な解表現が可能となり、離散モデルでは表現できなかったレイアウトが表現可能となる。FLP0(1990)[Montreuil 90]は初期の連続、ただ境界制約変数付きモデルである。精度上がるため、FLP1(1999)[Meller 99]モデルで職場長さや幅の制約(上限と下限など)を加えと修正した。FLP2(2003)[Sherali 03]は形状制約の線形近似を行った(形状の自由度の向上を求める)。

① 基本定式化[大森 11]

初期提案した施設レイアウト問題のモデルの制約が簡単で、職場は施設内に配置することや職場同士重なってはいけないようなものである。

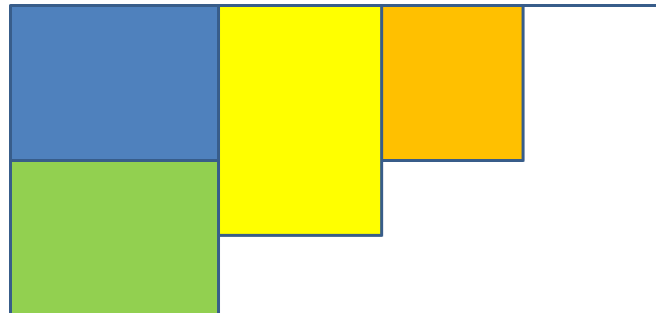


図 2.8 連続表現

目的関数

$$\text{Minimize } \sum_{i=1}^M \sum_{j=1}^M D_{ij} F_{ij}$$

Subject to

$$X_{ij} Y_{ij} = 0 \quad (2.1)$$

$$\frac{\omega_i}{2} - x_i \leq 0 \quad (2.2)$$

$$\frac{h_i}{2} - y_i \leq 0 \quad (2.3)$$

$$x_i - \left(W - \frac{\omega_i}{2}\right) \leq 0 \quad (2.4)$$

$$y_i - \left(H - \frac{h_i}{2}\right) \leq 0 \quad (2.5)$$

但し、

$$D_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (2.6)$$

$$X_{ij} = \max\left\{0, \frac{\omega_i + \omega_j}{2} - |x_i - x_j|\right\} \quad (2.7)$$

$$Y_{ij} = \max\left\{0, \frac{h_i + h_j}{2} - |y_i - y_j|\right\} \quad (2.8)$$

入力変数：

$M$ ：職場の数

$W$ ：施設の幅

$H$ ：施設の高さ

$\omega_i$ ：職場  $i$  の幅

$h_i$ ：職場  $i$  の高さ

$F_{ij}$ ：職場  $ij$  間の物流量

出力変数：

$x_i$ ：職場  $i$  の  $x$  座標

$y_i$ ：職場  $i$  の  $y$  座標

目的関数は職場間の総運搬コストの最小値を求める。式(2.1)は施設内の職場が重ならないための制約である。 $X_{ij}$ と $Y_{ij}$ は水平方向と垂直方向への職場 $i, j$ 間の相対位置を示す変数。職場 $i, j$ の水平方向と垂直方向の距離は職場 $i, j$ の高さや幅の和の半分より大きくなれば、0となる。式(2.2)~(2.5)は職場が施設内に配置するための制約である。

## ② 改良モデル

Mellerら[Sherali 03]は大規模な問題（職場数9以上の問題）へ応用するため、FLP2のモデルを提案した。職場の面積に関する制約を線形近似するために式(2.20)を導入された。これにより、線形計画法（Linear Programming）[並木 08]を適用できる。数学的には線型計画問題は、目的関数と制約条件がすべて線型の最適化問題である。線形計画法の解法の一つ単体法(シンプレックス法)は、実行可能解の1つから出発して目的関数の値をなるべく大きく(小さく)するようなところに移動させていく動作を繰り返して最適解を見つけ出す方法である。各ステップで必ず目的関数の値は改善されるという特徴がある。

大森ら[大森 13]はFLP2モデルの相対位置の計算の制約を軽減して高速化を図った。以下は文献[大森 13]の定式化方法である。

$$\text{Minimize} \quad \sum_{i=1}^N \sum_{j>i}^N f_{ij} (d_{ij}^x + d_{ij}^y)$$

Subject to

$$d_{ij}^x = |c_i^x - c_j^x| \quad (2.9)$$

$$d_{ij}^y = |c_i^y - c_j^y| \quad (2.10)$$

$$l_i^x \leq c_i^x \leq L^x - l_i^x \quad (2.11)$$

$$l_i^y \leq c_i^y \leq L^y - l_i^y \quad (2.12)$$

$$lb_i \leq 2l_i^x \leq ub_i \quad (2.13)$$

$$lb_i \leq 2l_i^y \leq ub_i \quad (2.14)$$

$$c_i^x + l_i^x \leq c_j^x - l_j^x \quad (\text{if } H_{ij}=1) \quad (2.15)$$

$$c_i^y + l_i^y \leq c_j^y - l_j^y \quad (\text{if } V_{ij}=1) \quad (2.16)$$

$$\alpha_i l_i^x + 4\bar{x} l_i^y \geq 2\alpha_i \bar{x} \quad (2.17)$$

入力変数 :

$f_{ij}$ : 職場  $i$ 、 $j$  間の物流量

$N$ : 職場の数

$\alpha_i$ : 職場の面積

$\partial_i$ : 職場の最長の辺と最短の辺の比率

$L^x, L^y$ : 施設の横幅と長さ

$lb_i$ : 職場  $i$  の横幅と縦幅の下限

$ub_i$ : 職場  $i$  の横幅と縦幅の上限

$\bar{x}$ : 職場の形状制約の外接点 (計算方法は、式(2.20)参照)

出力情報（決定変数）：

$d_{ij}^x, d_{ij}^y$ : 職場  $i, j$  間の  $x, y$  方向の距離

$c_i^x, c_i^y$ : 職場  $i$  の  $x, y$  座標

$l_i^x, l_i^y$ : 職場  $i$  の長さ と 横幅 の 半分

$H_{ij}, V_{ij}$ : 水平方向 と 垂直方向 への 職場  $i, j$  間の 相対位置 を 示す 変数

式(2.9)と(2.10)は職場間の距離、式(2.11)と(2.12)は職場を施設内に配置するための制約、式(2.13)と(2.14)は職場の長さ と 横幅 は 上限 と 下限 の 範囲 を 抑える ような 制約 である。ここで、上限 ( $ub_i$ ) と 下限 ( $lb_i$ ) は、式(2.18)と(2.19)で計算する。

$$ub_i = \min\{\sqrt{\partial_i \alpha_i}, \max_s\{L^s\}\} \quad (2.18)$$

$$lb_i = \alpha_i / 4ub_i \quad (2.19)$$

式(2.15)と(2.16)は、各職場が重ならないための制約である。また、式(2.17)は職場の面積を一定にするための制約である。ただし、職場の形状制約の外接点の数は式(2.20)で計算する。また、サポート点を多くすることで近似の精度を高めることがわかった [Sherali 03]。

$$\bar{x} = lb_i + \frac{r}{(\Delta-1)}(ub_i - lb_i) \quad \forall r = 0, 1, \dots, \Delta - 1 \text{ for } \Delta \geq 2 \quad (2.20)$$

従来研究 [Sherali 03] により、 $\Delta=20$  のとき、誤差は十分小さいことが示されている。

## 2.2 室内レイアウト問題

室内レイアウト問題とは、テーブル、タンス、テレビ、ベッドなどの家具一式の配置を決定する問題である。ただし、その配置にはドアや窓の位置、コンセントやモジュラージャック、家具の向きなどを考慮する必要がある。室内レイアウト問題を計算機で解くために、この問題は制約充足問題としてよく定式化されている。制約充足問題 (Constraint Satisfaction Problem) とは「離散値をとるいくつかの変数に割り当てられる組合せのうち、与えられた制約をすべて満たす組合せを発見する探索問題」である。



以下に示すような多種多様な制約を考慮しながらレイアウトを作成する必要がある。

- 家具は室内に配置する
- ドアの周囲にスペースをあける
- 家具は配置図上で重複してはならない
- 窓はできるだけふさがないように配置する
- 出入り口と各家具を結ぶ通路が存在する
- 家具の機能を妨げないように配置する
- 壁際に配置するのが好ましい家具は壁際に配置する
- 家具を部屋の内側に向けて配置する

それぞれの制約に重み付けを行い、これらの合計を目的関数とする最適化問題と考える。ここで目的関数は次節で定義する制約違反点数の合計であり、この値を最小化することが問題解決につながる。室内レイアウト問題を重み付き制約充足問題として図 2.9 のように定式化する。ただし、家具の数を  $i$ 、制約の数を  $k$  とする。

対象問題  $= (N, F, R, C, W)$

$N = \{ 1, 2, 3, \dots \}$

家具の配置順

$F = \{ F_1, F_2, \dots, F_i \}$

配置する家具の集合

$R = \{ \text{面積, ドア, 窓, } \dots \}$

配置する部屋の情報

$C = \{ C_1, C_2, \dots, C_k \}$

レイアウト問題の制約

$W = \{ w_1, w_2, \dots, w_k \}$

各制約に課せられた違反点数

図 2.9 定式化

レイアウト問題に対して Kim ら [Kim 00] は IGA (interactive genetic algorithm) を用いたファッション デザインシステムを提案した。このシステムは、一世代ごとに、生成した個体をユーザに提供し、ユーザで評価する。このように、ユーザの好みや感情を有効的に結合に探索できる。しかし、レイアウト問題は、探索空間が広くて、評価回数は数え切れないほど多く、すべての染色体はユーザで評価するのは不実現である。

脇下ら [脇下 96] は、制約に基づく対話型室内レイアウトシステムを開発した。このシステムは既存のレイアウトを初期個体として探索が始めてユーザはインターフェー

スの違反情報ナビにより、自分の好みに従い、家具の場所と向きを変更できる。このシステムはユーザの気に合う配置図が作れるが、作れた配置図は既存した配置図と類似する場合が多く、個人の発想が難しいという問題点がある。

## 2.3 施設レイアウト問題の解法

### (1) アントコロニー最適化法 (ACO)

アントコロニー最適化法 (ACO) とは、蟻の採餌行動をモデル化したマルチエージェント型の最適化手法である。ACO は探索領域の情報をフェロモンと見なし、複数の蟻がフェロモンに基づいて解を作成する。そして、作成された解の情報がフェロモンにフィードバックされ、更新されたフェロモンを利用して蟻が解を作成することで探索が進む[Dorigo 91]。

Komarudin ら[Komarudin 10]は、解の表現として *slicing tree* を用いた Ant System を FLP に適用した。Slicing tree に従い、施設の空間を順番に分割し、職場を配置する。また、この手法を局所探索法とハイブリッド化することにより性能の向上を図っている。図 2.10 は染色体の構造を示す。

図 2.11 は木構造表現の一例である。H は縦方向、V は水平方向に配置することを意味する。図 2.12 は図 2.11 の木構造によって配置したレイアウトである。まず 3 番の職場は 1 の横方向に配置して、つぎに 3 と 1 番の上に 2 と 4 番の職場を縦方向に配置する。



図 2.10 Komarudin の手法の染色体の構造

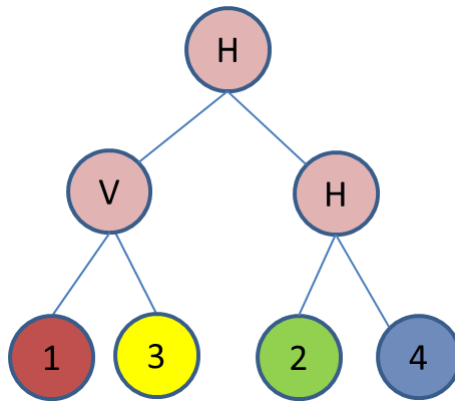


図 2.11 木構造の表現

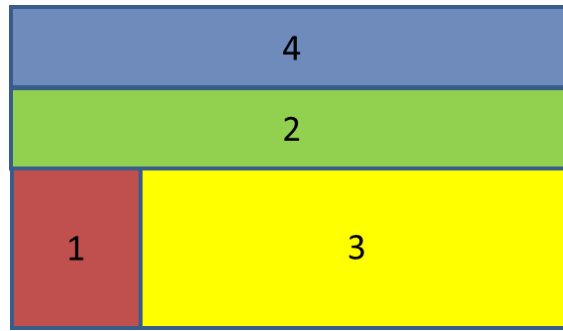


図 2.12 レイアウトの表現

(2) 力学モデル[大森 13]

大森らは、FLP における職場間の物流量の大きさをバネと見なし、「カオス挙動」を加えることで相対位置を効率的に求めるヒューリスティックを提案した。この方法は MIP と組み合わせて大規模な問題に対応でき、かつ職場の移動における局所解への収束を抑制した。また、線形計画法を適用するため、制約を軽減することにより高速化を図った。

図 2.13 はアルゴリズムの流れを示す。この方法は大域探索、相対位置抽出、局所探索のステップを繰り返す。初期の大域探索は職場が相対位置関係を得られる。その後、LP 方法の一つ「主双対内点法」を用いて、職場の形状を最適化する。

```

初期点一つをランダムに選択する；
for(終了条件){
    「カオス力学的モデル」による初期配置(職場の座標)を決定する；
    職場座標から職場同士の相対位置関係 (H,V) を抽出する；
    「主双対内点法」による職場位置，形状 (職場の座標、幅と長さ) の最適化
}

```

図 2.13 力学モデルのアルゴリズムの流れ

### (3) 遺伝的アルゴリズム(GA)

遺伝的アルゴリズムは、自然淘汰や、突然変異といったいわゆる生物の進化過程をモデル化した確率的探索アルゴリズムである[北野 93]。GA は問題の探索空間に依存することなく準最適解へ高速に収束できることから、さまざまな最適化問題に適応されてきた。GA では、問題の解候補を個体に見立て、それら個体により構成される集団内で選択・交叉などの遺伝的操作を行うことによって探索を進める。GA は、ダーウィン進化論に基づく生物進化をモデル化した、確率的探索手法であり、大域的探索能力が優れているという特徴があるため、複雑な問題への適用が検討されている。

Liu らは、sequence-pair 表現を用いて MIP と GA をハイブリッド化した手法を提案し、大規模な問題に適用して職場同士の相対位置をバイナリ変数、職場の座標・形状を連続変数として定式化を行った[Liu 07]。従来方法は大量な二値変数を使って、データ処理の量が膨大なので  $n$  個職場が  $2^{2n(n-1)}$  種の 0-1 組み合わせが存在する。また、職場が施設内に配置することが約束しないため、MIP モデルに適用しない個体を生成する可能性がある。

そこで、Liu らは遺伝子配列ペアを用いてデータの処理量の削減を目指す。図 2.14 はコード化方法を示す。遺伝子は職場番号と選択する二つ個体でのこの職場の配置順とする。この方法は新しい位置関係の作りと保存がやすいというメリットがある。

遺伝操作として、交叉は Order-based Crossover 方法、突然変異は two-exchange ランダムで交換する方法を使った。

職場番号	3	2	1	4	5	6
職場配置順のペア	(3,5)	(6,3)	(2,2)	(4,6)	(1,4)	(5,1)

図 2.14 Liu のコード化法

図 2.17 は GA 手法の流れである。施設からはみ出した職場が含む新個体を排除するため、GA で生成した個体は FLP モデルへ検証する。また、探索性能を評価するため、問題 O7、O8、O9 に限って aspect ratio を 4 に固定し、厳密解を求めた。これにより、提案手法で得られた解の探索時間と厳密解を得られた時間を比較できる。

Gonçalves ら[Gonçalves 15]は職場の配置順，職場の縦横比，最初に配置する職場の座標で染色体を表現し，random-key GA を使って効率的な配置順を生成する方法を提案する。染色体は図 2.15 で示す。

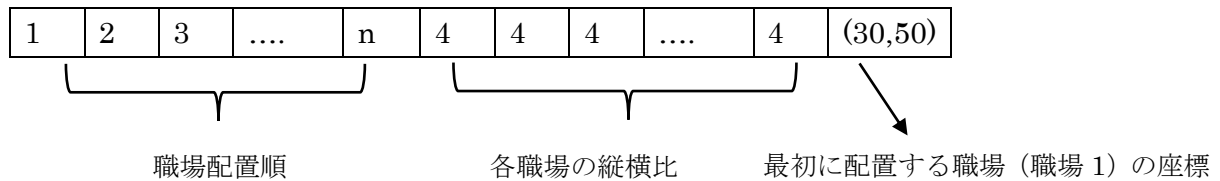


図 2.15 Gonçalves の手法の染色体

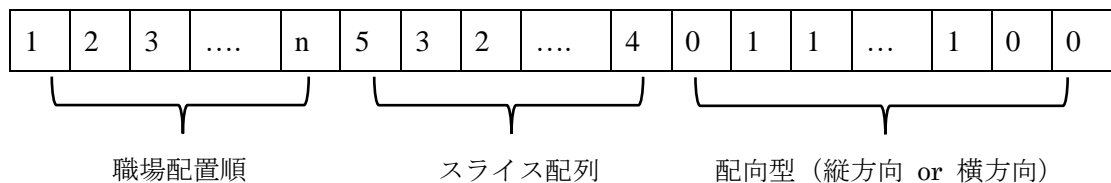


図 2.16 Kazi の手法の染色体

Kazi ら[Kazi 11]は、slicing tree を通して配置順と分割演算子を組み合わせた染色体を用いている。図 2.16 は使用する染色体を示す。また、多目的 GA を導入して、計算の速さと適応度の改善を求める。式(2.21)は職場間の総運搬コストを計算できる。式(2.22)は職場間の位置関係を示す。CR Value は職場間の関連度に応じて値を設定される。表 2.2 は関連度を示す。

その後、探索効率を向上するため、[Kazi 13]で 3 点交叉方法を用いて遺伝操作を改良した。

$$F_1 = \sum_{i=1}^n \sum_{j=1}^n f_{ij} d_{ij} \quad (2.21)$$

$$F_2 = \sum_{i=1}^n \sum_{j=1}^n r_{ij} \quad (2.22)$$

Subject to

$$r_{ij} = \begin{cases} \text{CR Value} & \text{if 職場 } i \text{ が職場 } j \text{ の隣} \\ 0 & \text{otherwise} \end{cases} \quad (2.23)$$

表 2.2 CR Value の設定

CR Value	Clossness
6	Absolutely Necessary
5	Especially Important
4	Important
3	Ordinary Important
2	Unimportant
1	Closeness Not Desirable

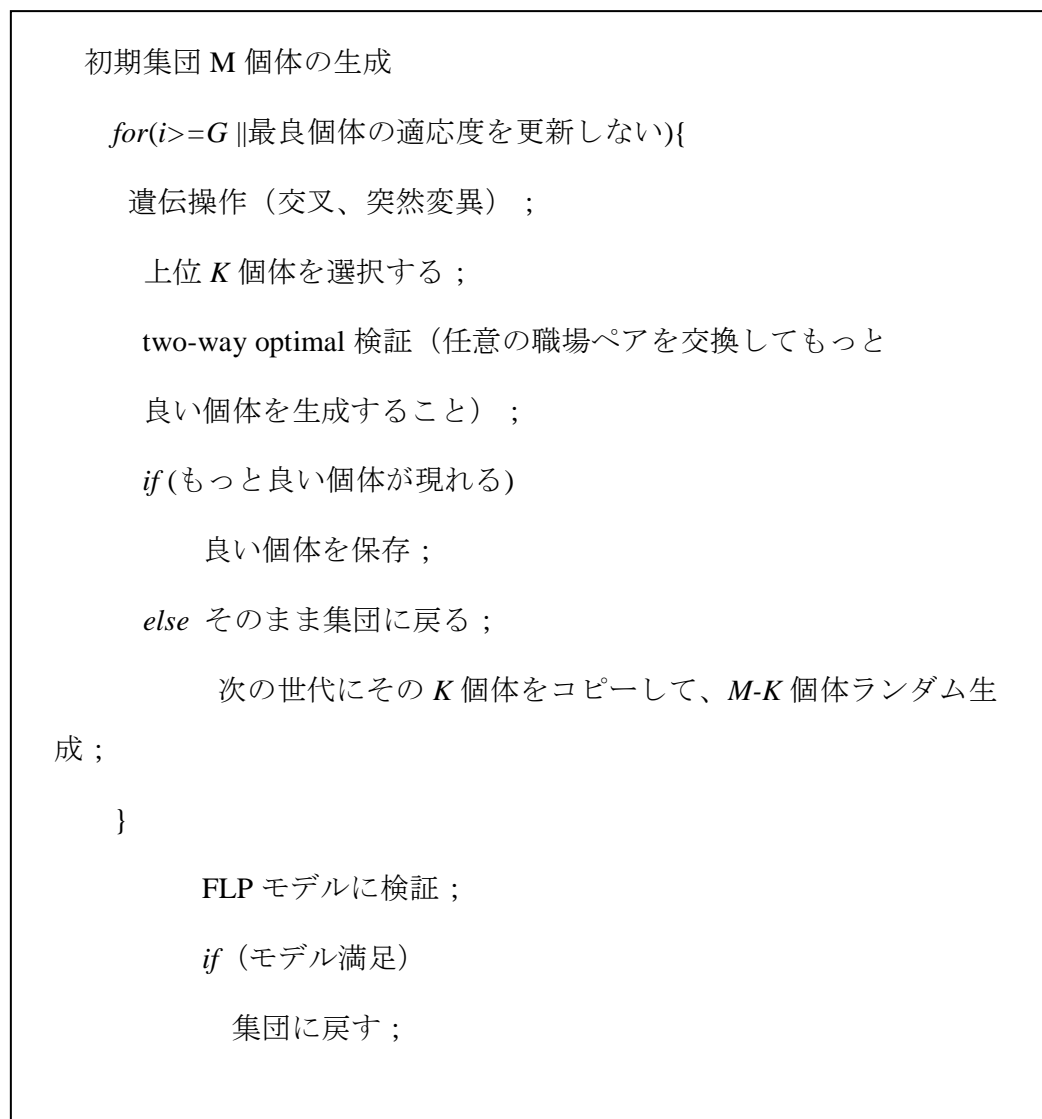


図 2.17 Liu の GA の流れ

#### (4) ES

進化戦略 (ES) は、実数空間を定義域とする非線形最適化問題を解くことを目的に、Rechenberg と Schwefel によって 1960 年代に開発されたアルゴリズムである[長尾 00]。

ES における探索の主な特徴としては、

- 扱う問題として  $n$  次元実数空間  $\mathbf{R}^n$  上での最適化問題を想定する
- 正規分布に従った摂動を加える形で突然変異を導入しており、これを主な探索手段としている
- 選択（淘汰）は適応度の上位から確定的に行う
- 突然変異を主演算とする

などが挙げられる。

TSP においては、ES と次のように定式化する

- (1)  $\mu$  個の個体からなる親集団を生成する。
- (2) 突然変異などにより  $\lambda$  個の子孫を生成する。
- (3) 適応度の高い順に  $\mu$  個を親として選び次世代集団とする。

また、 $(\mu+\lambda)$ -ES はエリート保存による多点探索という特徴がある。

主な突然変異方法は図 2.18 のように、逆位 (inversion)、挿入 (insertion)、交換 (2-exchange)、移動 (shifting) 4 種類がある。矢印はランダムで選択した位置である。

Asl らは、施設レイアウト問題に対して ES、PSO、GA の 3 つの手法の比較実験を行って、ES が PSO や GA より良い結果を得ることができることを示した [Asl 16]。



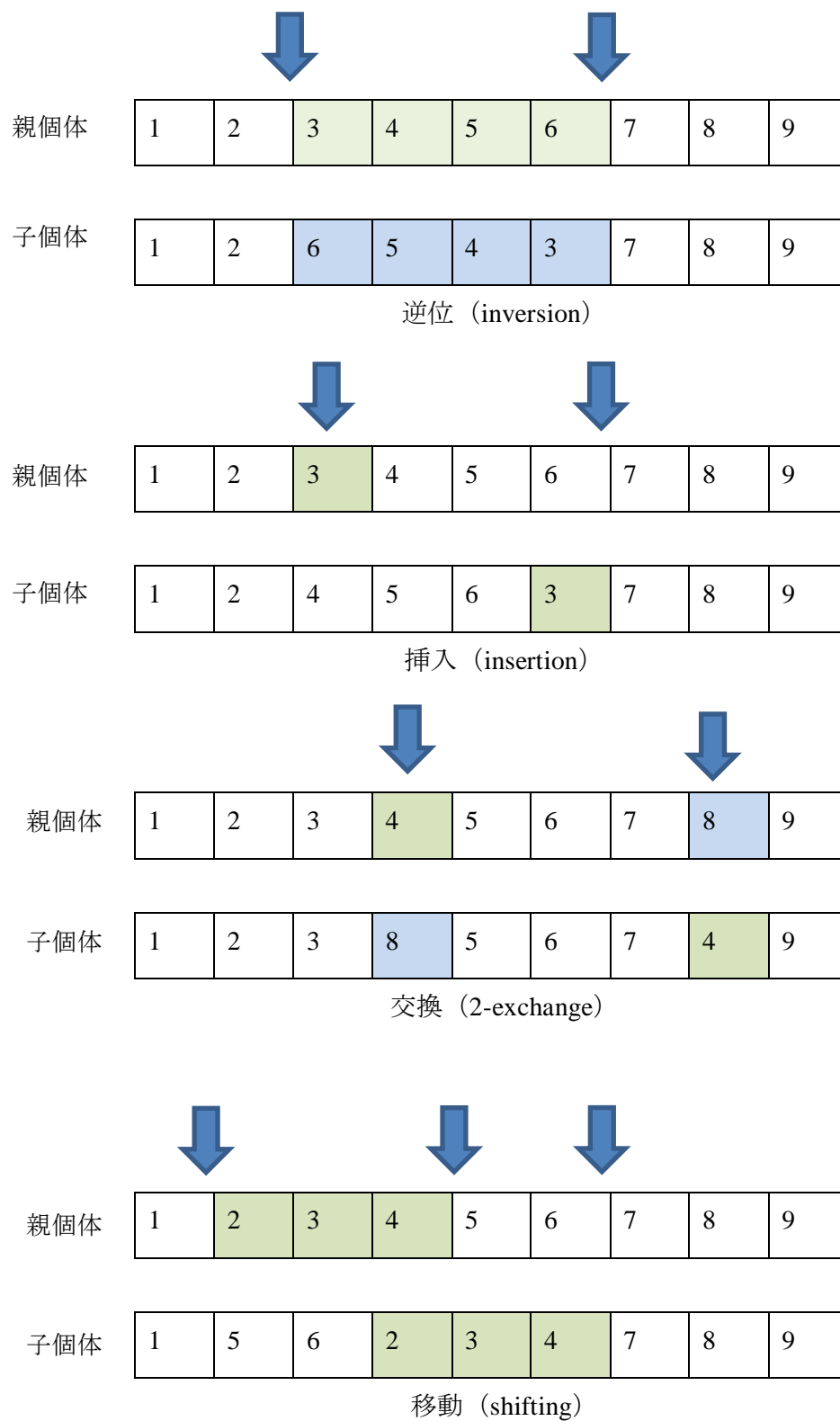


図 2.18 突然変異方法

## 2.4 Levy Flight

動物は変化する自然環境の中で、獲物の居場所をどうやって効率的に探しているのか、という疑問は古くから研究されてきた。獲物は、広い範囲にランダムかつまばらに分布している。また、獲物の居場所に関する捕食者の知識は曖昧である。このような状況下で、動物は自然淘汰の結果として LF による採餌行動を行っているので、LF の探索効率は最適である (LF 採餌仮説) と仮定する[Viswanathan 11]。

LF では探索空間における探索点の 1 ステップの移動は式(2.24)の分布に従う[Viswanathan 11]。ただし、 $l$ は移動距離 (ステップ幅)、 $u$ は範囲  $1 < u \leq 3$  の定数である。 $u > 3$  の時、Random Walk モデルになることをわかった。 $u < 1$  の時、正規分布となる[Buldyrev 93]。

$$P(l) \sim l^{-u} \quad (2.24)$$

次の確率密度関数で定義される標準 Levy 分布は式(2.25)と図 2.19 で示す。移動距離  $l$ はこの確率密度関数に従う乱数 (Levy 乱数[四辻 10]) として生成することになる。

$$P(l) = \frac{1}{\sqrt{2\pi}} l^{-\frac{3}{2}} e^{-\frac{1}{2l}} \quad (2.25)$$

通常のランダム探索では、同じ場所を繰り返して探索してしまうことが指摘されているが、LF に基づく探索では、遠方への大域探索を織りまぜて実行することになるので、動物の採餌行動に類似して効率よく探索することができると考えられている。図 2.19 は標準正規分布( $l > 0$ )と標準 Levy 分布の比較図である。 $P(l)$ は確率、 $l$ は距離である。近年、探索の分野で Levy 分布が注目されている理由は、従来から利用されていた正規分布や一様分布にはない「ヘビーテール」という特徴があるからである。

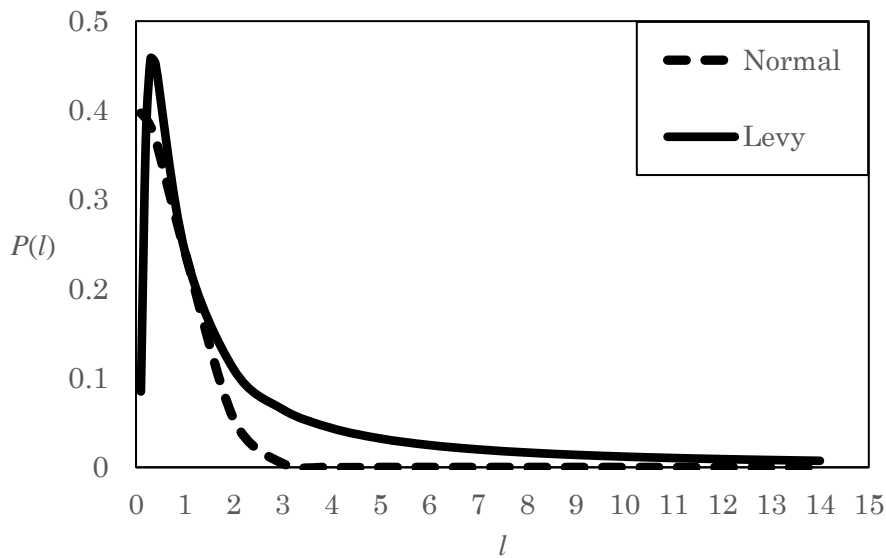


図 2.19 標準正規分布( $l > 0$ )と標準 Levy 分布

LF は、物理学、生物学、金融、統計、経済の分野の最適化問題に応用されている [Kamaruzaman 13]。LF は、実数値の探索空間における探索点の移動方法と位置づけられるので、メタヒューリスティックと組み合わせて関数最適化問題に直接適用することができる [Lee 04][Bartumeus 05] [Yang 09]。しかし、組み合わせ最適化問題への応用は調査範囲内にあまり見られない。

二つの図 2.20、図 2.21 は正規分布に従った random walk と Levy 分布に基づく探索方法のルートを示す。こちらの図から見ると、同じの範囲で、通常のランダム探索では、同じ場所を繰り返して探索してしまう。LF に基づく探索では、遠方への大域探索を実行することになるので、動物の採餌行動に類似して効率よく探索できると想定できる。

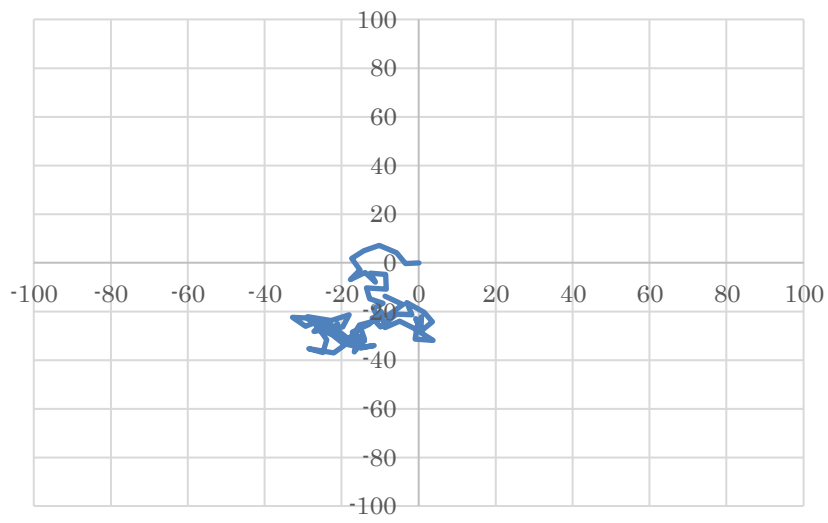


图 2.20 Random Walk

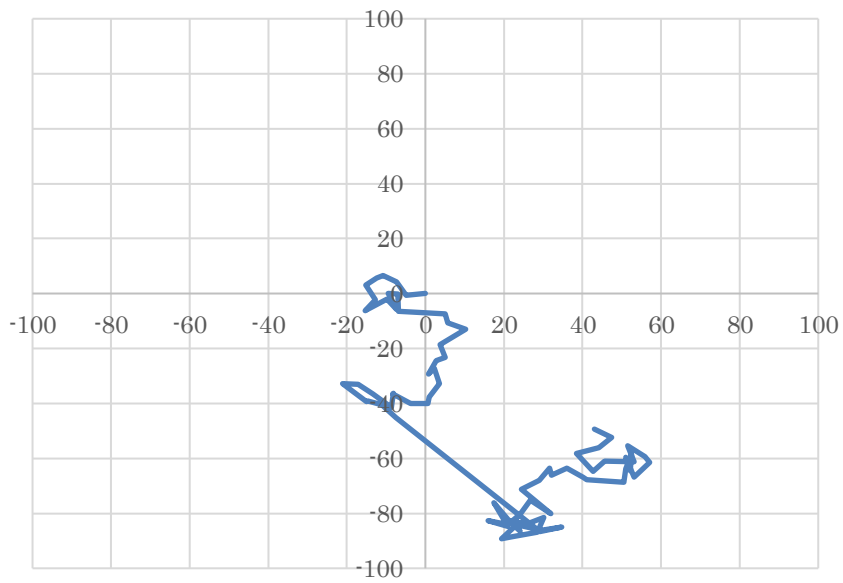


图 2.21 Levy Flight

\*Random Walk の図 2.20 の生成方法：

- ① ランダムに  $N$  個角度  $\theta$  ( $0^\circ \sim 360^\circ$ ) を生成する。
- ② 正規分布に従って、 $N$  個移動距離  $l$  を生成する。正規分布の平均値は 5、標準偏差 1 ( $Z(5,1)$ ) とする。つまり、移動距離の平均値は 5 である。
- ③ 極座標を式と式によって直角座標に変換し、次の座標を生成する。
- ④ 生成した座標により、Excel を用いて図を作成する。

$$x_i = \cos(\theta_i * \text{PI}/180) * l_i + x_{i-1}$$

$$y_i = \sin(\theta_i * \text{PI}/180) * l_i + y_{i-1}$$

(ただし、 $i=0$  の時、 $x_{i-1}=0$ ;  $y_{i-1}=0$ 。)

Levy Flight の図 2.21 の生成方法：

- ① ランダムに  $N$  個角度  $\theta$  ( $0^\circ \sim 360^\circ$ ) を生成する。
- ② Levy 分布(式)に従って、 $N$  個移動距離  $m$  を生成する。levy 分布の平均値は 5、標準偏差 1 とする。

$$m = \frac{1}{z(0,1)^2} + 5$$

- ③ 極座標を式と式によって直角座標に変換し、次の座標を生成する。
- ④ 生成した座標により、Excel を用いて図を作成する。

$$x_i = \cos(\theta_i * \text{PI}/180) * m_i + x_{i-1}$$

$$y_i = \sin(\theta_i * \text{PI}/180) * m_i + y_{i-1}$$

(ただし、 $i=0$  の時、 $x_{i-1}=0$ ;  $y_{i-1}=0$ 。)

## 第三章 ベンチマーク問題への適用

この部分の主題は、探索における LF の有効性を示すことなので、遺伝的演算としては、最も単純な突然変異のみを用いる。

### 3.1 提案手法

#### 3.1.1 コード化

施設レイアウト問題で職場の配置順は総運搬コストの計算（目的関数値）に重要な影響があるので、本研究ではコード化として巡回セールスマン問題でよく利用されている単純な順序表現を用いる。図 3.1 に染色体の例を示す。職場番号を遺伝子とする。

職場番号	2	1	4	5	3	6	8	7	9
------	---	---	---	---	---	---	---	---	---

図 3.1 染色体の例

この例では、職場 2,1,4,5,3,6,8,7,9 をこの順番で施設に配置することを意味する。図 3.2 に示した矢印の方向に従って空いているスペースを探す。施設上には左上から右下に向けて施設内に収まるように職場の配置順に従って配置し、職場がはみ出した場合は職場 4 のように下に配置する。

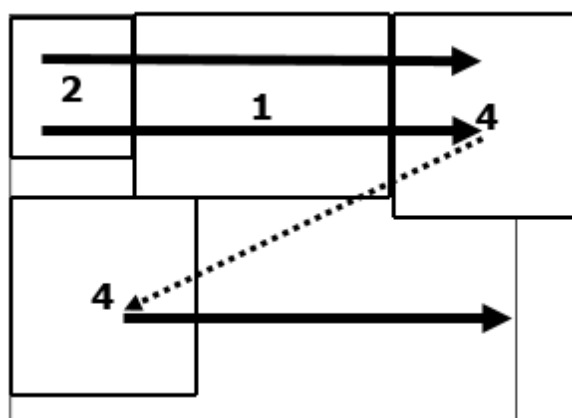


図 3.2 施設内に職場を配置する方向

### 3.1.2 突然変異

#### (1) 突然変異方法の選択

突然変異方法は ES に対してとても重要なので、予備実験を行った。順序表現に対する突然変異の方法としては、逆位 (inversion), 挿入 (insertion), 交換 (2-exchange), 移動 (shifting) が良く用いられている [Beyer 02]. 大森らの論文 [大森 11] に示されているベンチマーク問題 Y10 を用いて、実験を行った。図 3.4 は、問題 Y10 に対して ES を用いて同じ実験条件 ( $\mu = 200$ ,  $\lambda = 40$ ) で逆位 (inversion), 挿入 (insertion), 交換 (2-exchange) の結果を示す。

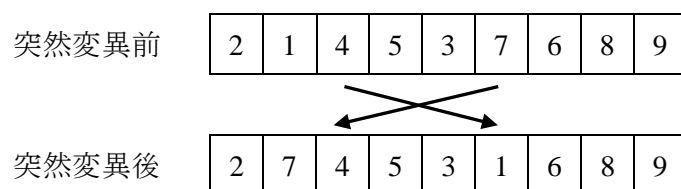
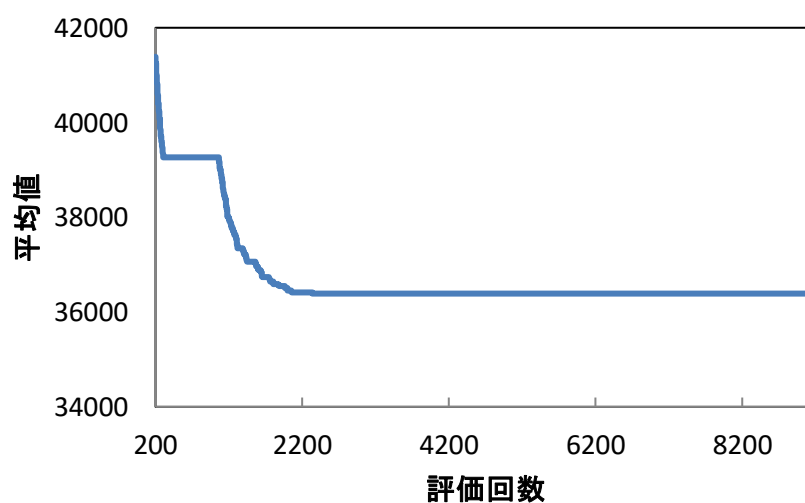
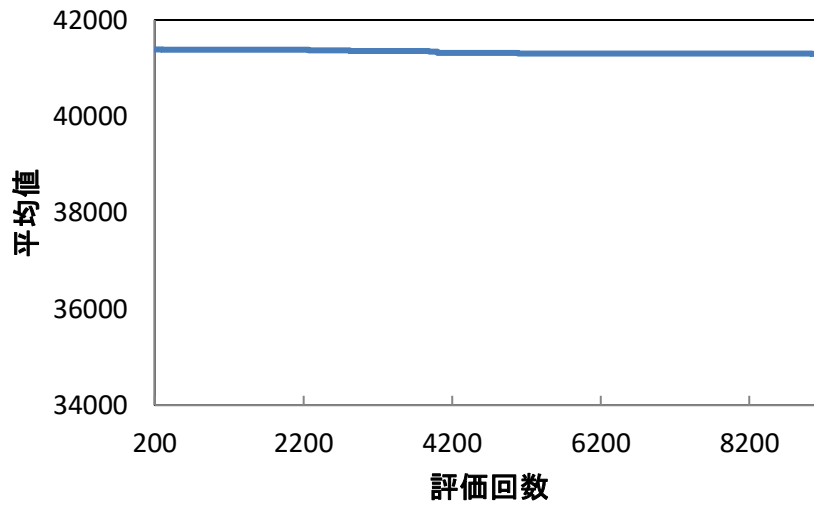


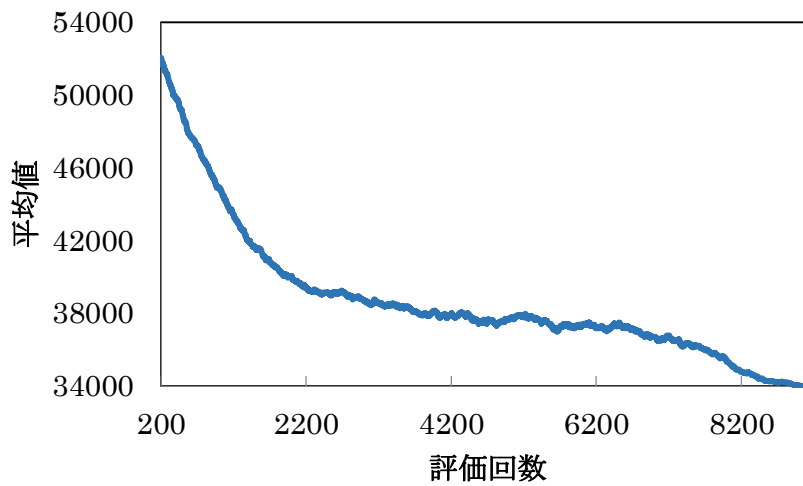
図 3.3 突然変異の例



逆位 (inversion)



挿入 (insertion)



交換 (2-exchange)

図 3.4 突然変異方法の比較

移動 (shifting) は挿入 (insertion) 方法と似ている部分があったので、結果は挿入方法ほぼ同じで、平均適応度をあまり改善しなかった。本研究では、これらの演算を予備実験により比較した結果、最も有効だった交換を用いる。すなわち、図 3.3 で示すような、ランダムに選んだ 2 つの遺伝子座の遺伝子を交換する。



## (2) 厳密解に関する考察

予備実験で使ったベンチマーク問題 Y10 の職場数は 10、かつ各職場の長さや幅は定数、また探索空間のサイズは  $10!=3628800$  なので、厳密解を計算できると想定できる。これにより、探索の挙動を考察できると考える。図 3.5 は上位 20 個体の適応度と遺伝子配列を示す。一番目の個体は得られた厳密解である。

	A	B	C	D	E	F	G	H	I	J	K
1	32564.64	4	8	3	7	9	5	1	0	6	2
2	32761.2	4	8	5	0	3	1	7	9	6	2
3	32761.2	4	8	5	0	3	7	1	9	6	2
4	32761.2	4	8	5	0	3	7	9	1	6	2
5	32838.76	3	5	6	0	1	7	9	2	4	8
6	32857.83	1	8	3	4	7	9	5	0	6	2
7	32857.83	1	8	3	7	4	9	5	0	6	2
8	32861.32	4	8	5	0	3	1	9	7	6	2
9	32861.32	4	8	5	0	3	9	7	1	6	2
10	32861.32	4	8	5	0	3	9	1	7	6	2
11	32862.89	6	3	4	5	2	9	8	7	0	1
12	32873.98	6	5	4	8	3	7	9	0	2	1
13	32873.98	6	5	4	8	3	9	0	7	2	1
14	32873.98	6	5	4	8	3	9	0	2	7	1
15	32873.98	6	5	4	8	3	9	0	2	1	7
16	32873.98	6	5	4	8	3	9	7	0	2	1
17	32873.98	6	5	4	8	7	3	9	0	2	1
18	32895.67	6	5	3	9	8	4	2	7	1	0
19	32895.67	6	5	3	9	8	4	2	7	0	1
20	32935.85	6	5	4	8	7	2	9	0	1	3

図 3.5 上位 20 個体の適応度と遺伝子配列

表 3.1 はある一回の ES の上位 50 の各職場は各遺伝子座で出現した回数を示す。表 3.2 は上位最良解の中で各遺伝子座の出現頻度が一番と 2 番高い職場番号を示す。下線を引いている数字は厳密解で出現した職場番号である。出現頻度が 2 番高い職場番号の順番と厳密解の職場順番が近いことをわかった。

表 3.1 上位 50 の各職場は各遺伝子で出現した回数

遺伝子座 職場番号	1	2	3	4	5	6	7	8	9	10
0	0	0	0	7	4	2	16	16	3	2
1	2	4	0	0	1	2	3	5	10	23
2	7	0	4	0	17	4	2	3	4	9
3	1	3	5	0	11	3	3	9	9	6
4	11	0	34	1	1	2	0	0	1	0
5	0	34	6	3	0	1	2	0	2	2
6	29	0	1	0	6	3	2	0	9	0
7	0	0	0	10	6	9	7	9	4	5
8	0	9	0	27	3	1	3	6	0	1
9	0	0	0	2	1	23	12	2	8	2

表 3.2 出現頻度が一番と 2 番高い職場番号

遺伝子座 上位個体範囲		1	2	3	4	5	6	7	8	9	10
		1~50	①	6	5	4	8	2	9	0	<u>0</u>
	②	<u>4</u>	<u>8</u>	5	<u>7</u>	3	7	9	(3,7)	( <u>6</u> ,3)	<u>2</u>
1~40	①	6	5	4	8	(2,3)	9	0	<u>0</u>	( <u>6</u> ,3)	1
	②	<u>4</u>	<u>8</u>	5	0	(6,7)	7	9	7	1	<u>2</u>
1~30	①	6	5	4	8	3	9	(0,9)	<u>0</u>	<u>6</u>	<u>2</u>
	②	<u>4</u>	<u>8</u>	5	0	2	7	7	7	<u>6</u>	0
1~20	①	6	5	4	(8,0)	3	9	9	<u>0</u>	<u>6</u>	<u>2</u>
	②	<u>4</u>	<u>8</u>	5	0	3	7	(9, <u>1</u> )	9	<u>6</u>	<u>2</u>
	最良解	4	8	3	7	9	5	1	0	6	2

図 3.6 はある一回の ES 探索した最良解と実際存在する最良解の区別を示す。  
この差はただ一回交換と挿入によって厳密解に辿れる。

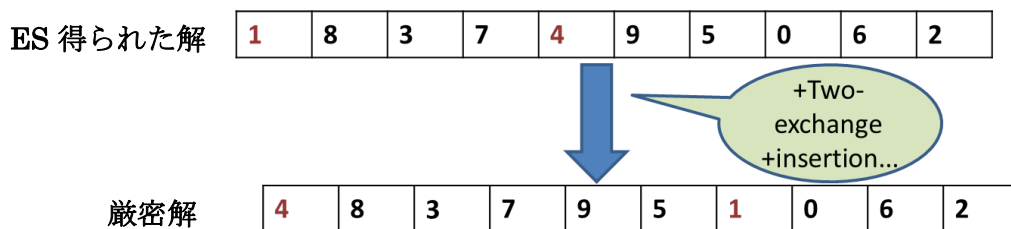


図 3.6 最良解と厳密解

従って適切な突然変異の回数は提案手法にとって重要である。

### 3.1.3 Levy Flight による解の更新方法

LF に基づく手法は本来連続値問題を対象としているが、FLP の遺伝子は離散値をとるので、LF の確率密度関数をそのまま適用することはできない。そこで、本論文では個体間の距離を以下のように定義する。一つの個体 A に遺伝子の交換（突然変異）を  $m$  回施して生成した個体を B とし、 $m$  を A-B 間の距離と見なす。すなわち、交換の回数を探索空間上の移動距離とする。下の例（図 3.7）を示すような、交換回数  $m=2$  の時、一つの個体 A に交換（職場 4 と 5、3 と 0）を 2 回施して生成した個体を B とし、2 を A-B 間の距離である。

以上より、探索の 1 ステップにおける解の更新を次の式(3.1)の  $m$  を用いて行う。ただし、 $L$  は染色体の長さ（職場数）、 $[l]$  は  $l$  の整数部分を表わす。なお、交換の回数が染色体長の半数を超えると同じ遺伝子座が複数回選択される確率が高くなると考え、 $l$  のしきい値を  $L/2$  と設定した。標準 Levy 分布により、 $[0,1]$  の levy 乱数が生成される確率が高いので、+1 する必要がある。式(3.1)で整数部分に 1 を足しているのは、切り上げるためである。また、 $m=0$  のときは乱数の生成をやり直す。

$$m = \begin{cases} [l] + 1 & \text{if } l < \frac{L}{2} \\ 0 & \text{otherwise} \end{cases} \quad (3.1)$$

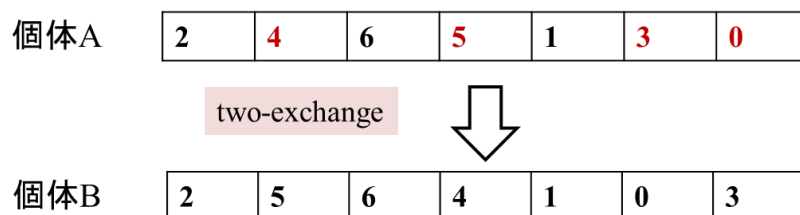


図 3.7 交換の例

### 3.1.4 アルゴリズム

本論文では、FLP の目的関数を適応度 *fitness* とする。提案手法のアルゴリズムを図 3.8 に示す。新しい配置（表現型）の生成においては、まず、各職場の縦横比をランダム（面積は保存）に設定しておいて 3.1 節に示した方法で配置を行う。このとき、施設の縦横の 20% のはみ出しは許容する。つぎに、2.1 節に示した改良モデルの制約条件を満たすように、LP（Linear Programming）を用いて各職場の位置と縦横比を再計算（最適化）する。このとき、職場が施設内に完全に収まらない場合は、その個体に式(3.2)で示すペナルティを付加する。

$$fitness\_new_{ij} = fitness_{ij} \times c \quad (3.2)$$

ただし、 $c$  は予備実験で決めた定数とする。

```
初期集団として  $\mu$  個の個体を生成する;  
for (指定世代まで) {  
  for ( $\lambda$  個の個体) {  
    集団からランダムに 1 個体選択する;  
    Levy 分布による乱数  $l$  を生成する;  
    突然変異(2-exchange)を  $m$  回に実行する;  
    配置を生成する;  
    各職場の座標と縦横比を LP で最適化する;  
    if (職場が施設からはみ出した)  
      その個体にペナルティを付加する;  
  }  
  親集団と子集団から  $\mu$  個の個体を適応度の順に選択し、次世代集団とする;  
}
```

図 3.8 提案手法のアルゴリズム

## 3.2 ベンチマーク問題の評価実験

### 3.2.1 実験方法

提案する手法の有効性を確認するため、表 3.3 に示すベンチマーク問題を対象として、次の手法と比較実験を行った。Levy 分布と比較しているのは、裾が指数関数的に減衰する正規分布 (EA-normal)、階段状に減衰する一様分布 (EA-uniform)、ならびに確率変数の一点でのみ値をもつ分布関数 (EA-fix) である。以下では提案手法を EA-levy と記す。

- A) 突然変異 (交換) の回数を固定した通常の EA (EA-fix)
- B) 提案手法のアルゴリズムの Levy 分布 (図 3.8 の下線部) を一様分布で置き換えた手法 (EA-uniform)
- C) 提案手法のアルゴリズムの Levy 分布を標準正規分布に置き換えた手法 (EA-normal)
- D) 提案手法と同じコード化を用いた GA
- E) 2.3 節に示した 6 つの従来手法 (文献の値と比較)

表 3.3 FLP のベンチマーク問題 [Komarudin 10]

問題	職場の数	施設の大きさ(m)
O7	7	8.54×13.00
O8	8	11.31×13.00
O9	9	12.00×13.00
Ba12	12	7.00×9.00
Ba14	14	6.00×10.00
AB20	20	2.00×3.00
SC30	30	15.00×12.00
SC35	35	16.00×15.00
Du62	62	100.00×137.18

## (1) 予備実験

### ① ペナルティ( $c$ )の決定

まず、通常の GA を用いてペナルティの最適値（範囲 1~1000）を決める実験を行った。表 3.4 は実験条件、表 3.5 は実験結果を示す。図 3.9~図 3.14 は各範囲の進化グラフ（縦軸は適応度、横軸は評価回数）である。実験結果から、 $c=100$  のときが一番良い。 $c=1000$ 、 $c=500$  のときは、初期解の変動が大きく（つまり進化が早い）また全体的に収束が遅い（特に後期）。 $c=10$ 、 $c=5$  のとき、掛け算をされた解（はみ出した解）と普通の解の差が小さいので良い解も淘汰された可能性がある。

表 3.4 実験条件

集団サイズ	200
問題	O9
突然変異率	0.15%
世代交代	MGG

表 3.5 ペナルティと最良解の関係  
(括弧内は標準偏差)

ペナルティ( $c$ )	最良値
5	232 (24)
10	222 (13)
50	222 (11)
100	214 (20)
500	219 (23)
1000	235 (18)

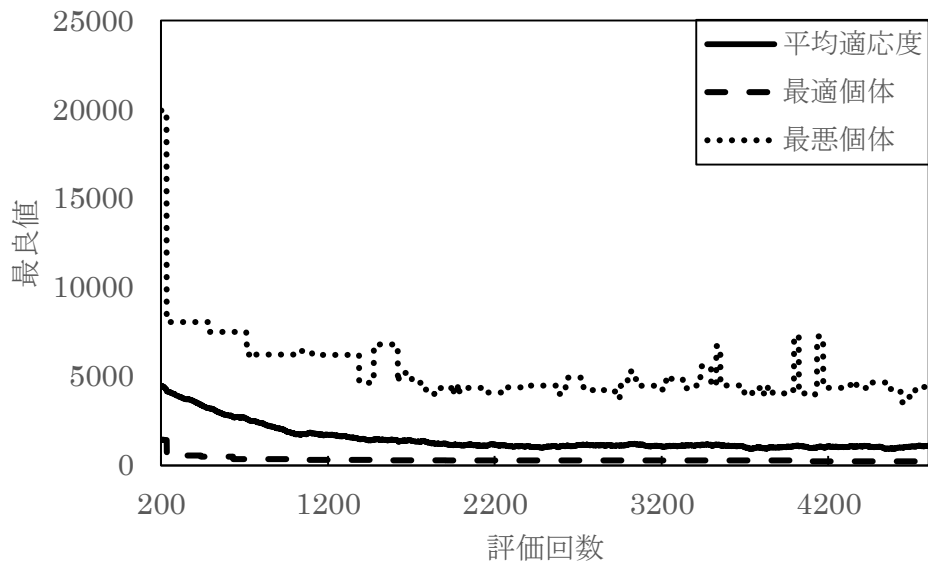


図 3.9 進化過程の例( $c=5$ )

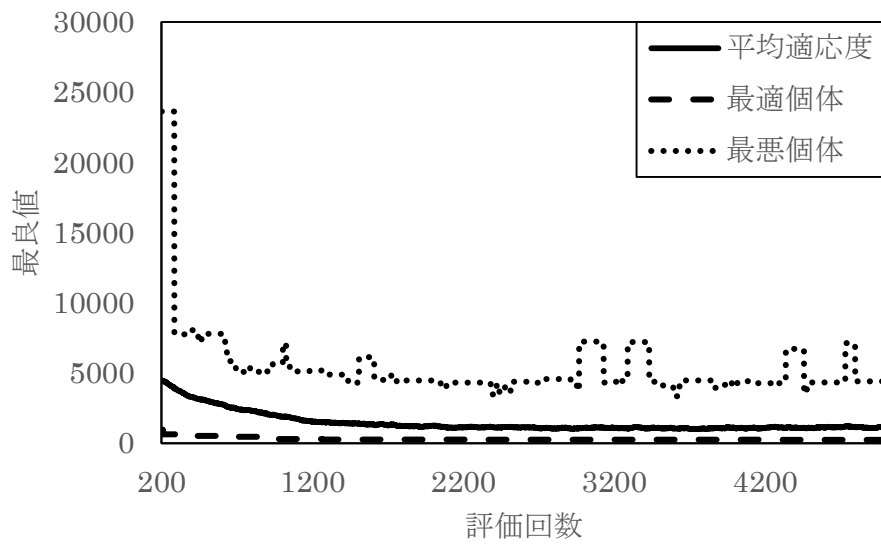


図 3.10 進化過程の例( $c=10$ )



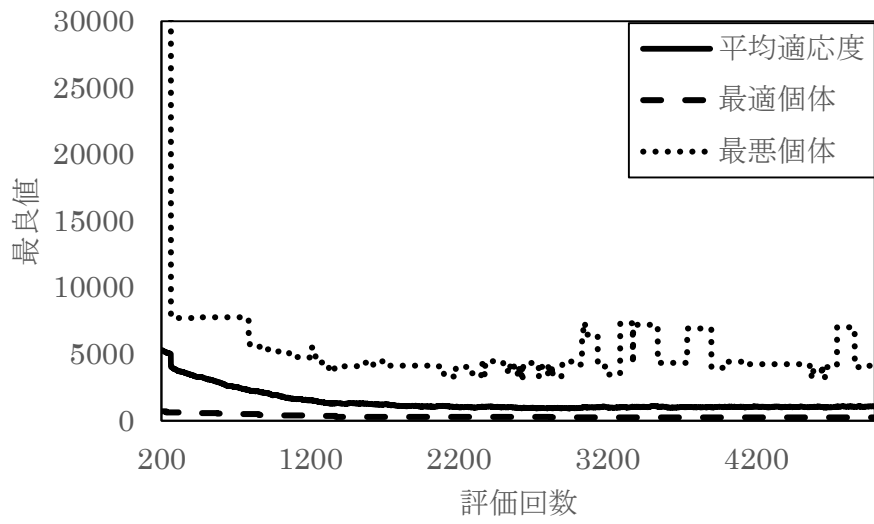


図 3.11 進化過程の例( $c=50$ )

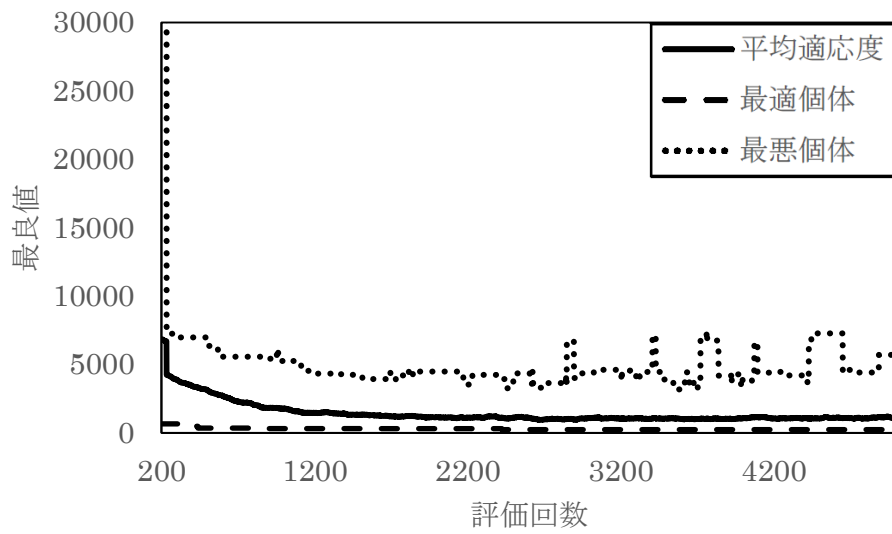


図 3.12 進化過程の例( $c=100$ )

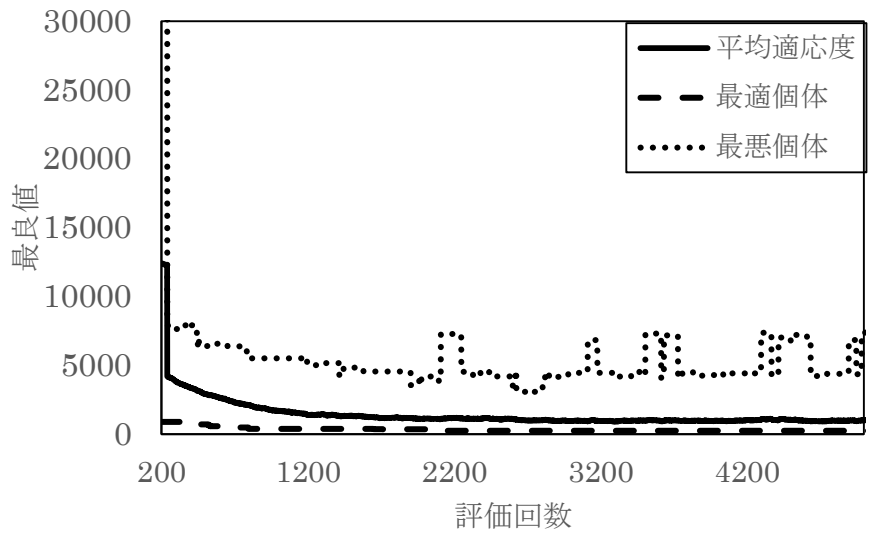


図 3.13 進化過程の例( $c=500$ )

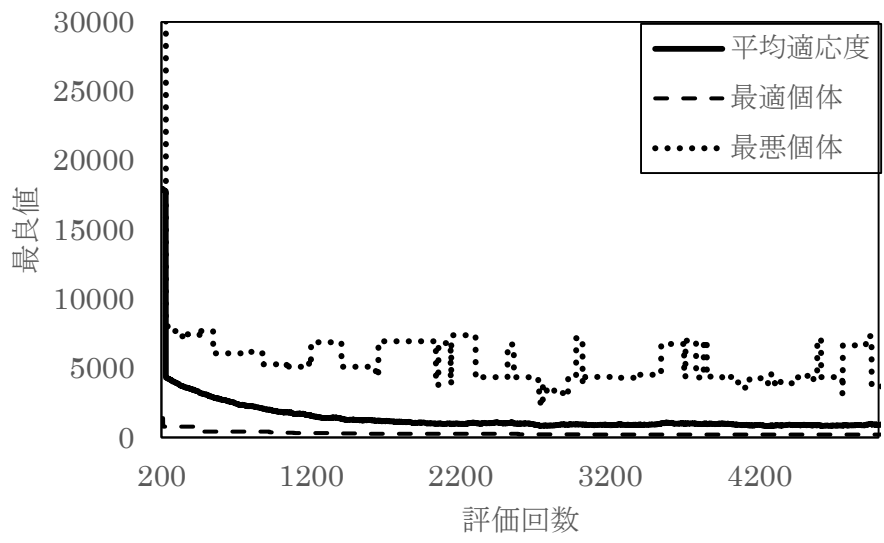


図 3.14 進化過程の例( $c=1000$ )

② EA のパラメータの決定

図 3.15～図 3.16 に、集団サイズ  $\mu=200$  のときの子集団サイズ  $\lambda$  と最良解の関係を示す。また、このときの実験条件を表 3.6 に示す。実験結果から、 $\lambda=50$  のときが良い結果となった。

表 3.6 実験条件

集団サイズ $\mu$	200
実行回数	10
突然変異方法	Two-exchange
ペナルティ	なし
対象問題	O9

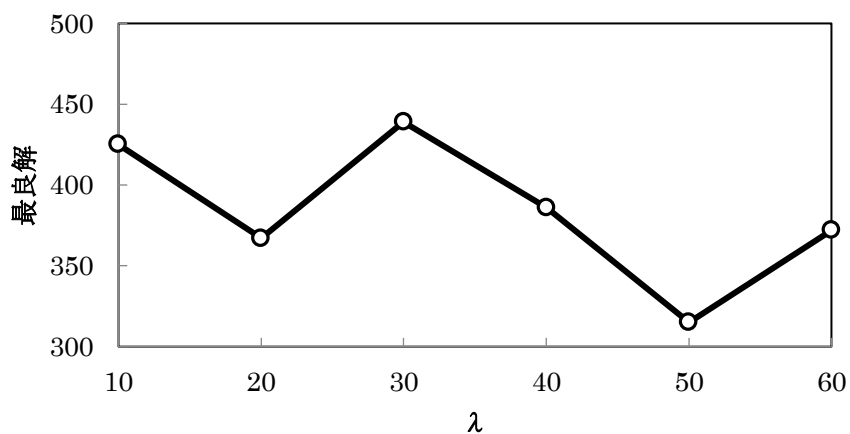


図 3.15 子集団サイズと最良解の関係 ( $\mu > \lambda$ )

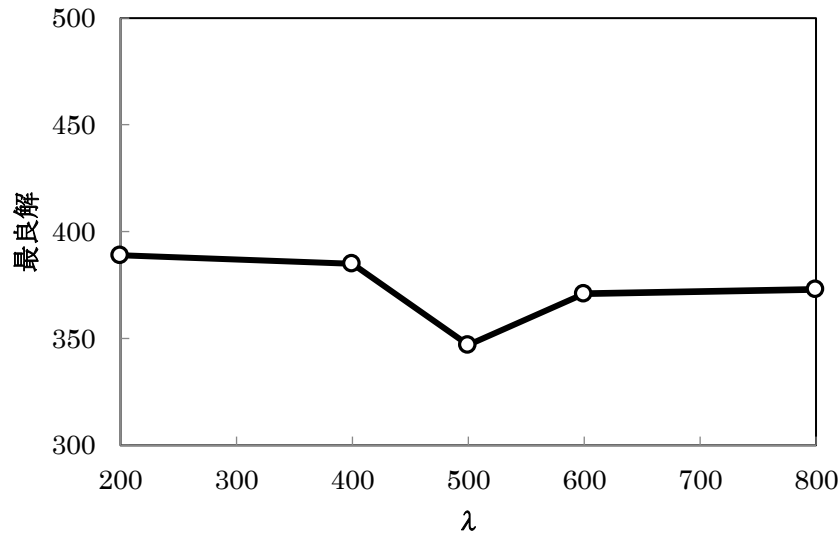


図 3.16 子集団サイズと最良解の関係 ( $\mu < \lambda$ )

また、他のパラメータについても、予備実験により表 3.7 のように設定した。EA-fix の突然変異の回数については、3.5.2 節で述べる。EA の世代交代は、すべて  $(\mu+\lambda)$ -ES と同じとした。GA の交叉としては、遺伝子が順列を表現する場合に広く用いられている OX (order crossover) を採用した。世代交代モデルとしては、多様性の維持に優れているといわれている MGG (Minimal Generation Gap) [佐藤 96]モデルを用いた。また、LP としては、単体法を用いた。EA と GA の評価回数の上限を同じにするため、GA の世代数の上限を 25000 とした。なお、EA と GA において、集団中の最良値と平均値の差が 0.001 以下になったときは、探索が収束した (集団が一様になった) と見なして探索を打ち切っている。

表 3.7 実験条件

パラメータ	EA-fix	EA-uniform	EA-normal	EA-levy (提案手法)	GA
集団サイズ ( $\mu$ )	200				
子集団サイズ ( $\lambda$ )	50				-
突然変異率	-				10%
ペナルティ倍率( $c$ )	100				
世代交代	$(\mu + \lambda)$ - ES				MGG
世代数	1000				25000
実行回数	30				
終了条件	$ \text{最良値}-\text{平均値}  < 0.001$				

## (2) 目標解の設定

表 3.8 の従来手法の計算コストについては、論文に十分な記述がないので提案手法と比較することは難しい。GA を用いた[Liu 07] [Goncalves 15] [Kazi 11,13] については、それぞれ、世代数 $\times$ 集団サイズ=10000, 70000~620000, 20000~100000 と読み取れる。これらの値は、探索が収束するのに十分な評価回数として設定しているものと思われるが、実際に収束した評価回数がわからないので、計算コストの比較はできない。また、ACO[Komarudin 10]と力学モデル[大森 13]に関しては、計算コストをどう評価するかは別の検討が必要かと思う。従って、本実験は文献の値だけを比較する。

提案手法の探索性能を評価するため、式(3.3)で定義する目標解  $S_{1.02}$  を導入した。これは、各手法で目標解を発見するまでの世代数と目標解を発見できた確率を比較検討するためのものである。ここで  $S_{best}$  は、各問題に対して本実験で得られた最良解（表 3.14 の mean の下線を施した値）とする。目標解として最良解に 1.02 を掛けた理由は、

最良解から 2%程度悪化した解ならば、全ての手法で発見できることが予備実験でわかっていたからである。図 3.17 は提案手法と ES の収束状況を示す。また、2 倍をかける時、収束が早すぎである。表 3.9 に各問題に対する  $S_{best}$  と  $S_{1.02}$  を示す。

表 3.8 従来研究の実験条件と計算コスト

論文	実験条件	実行時間記述かどうか	記述している評価回数
[Liu 07]	Pentium IV 3.2M Hz CPU and 2.0 GB of physical memory	あり	10000
[Goncalves 15]	a Intel Xeon E5-2630 @2.30GHz CPU and 16 GB of physical memory running the Linux operating system with Fedora release 18	無し	施設の数×100×100
[Komarudin 10]	an Intel Centrino Duo processor (1.7 GHz) and a Linux operating system.	あり	1500
[Kazi 11]	無し	無し	施設の数 < 15 : 20000 ; 施設の数 > 15 : 100000 ;
[Kazi 13]	無し	あり	施設の数 < 15 : 20000 ; 施設の数 > 15 : 100000 ;
[Ohmori 13]	3.4GHz Intel Core i7, 8GB physical memory, Mac OS	無し	— (ステップ数 : 5000)

$$S_{1.02} = S_{best} \times 1.02 \quad (3.3)$$

表 3.9 各問題の目標解

問題	$S_{best}$	$S_{1.02}$
O7	125	128
O8	232	237
O9	211	215
Ba12	6080	6202
Ba14	3211	3275
AB20	4429	4518
SC30	3102	3164
SC35	2980	3040
Du62	3120521	3182931

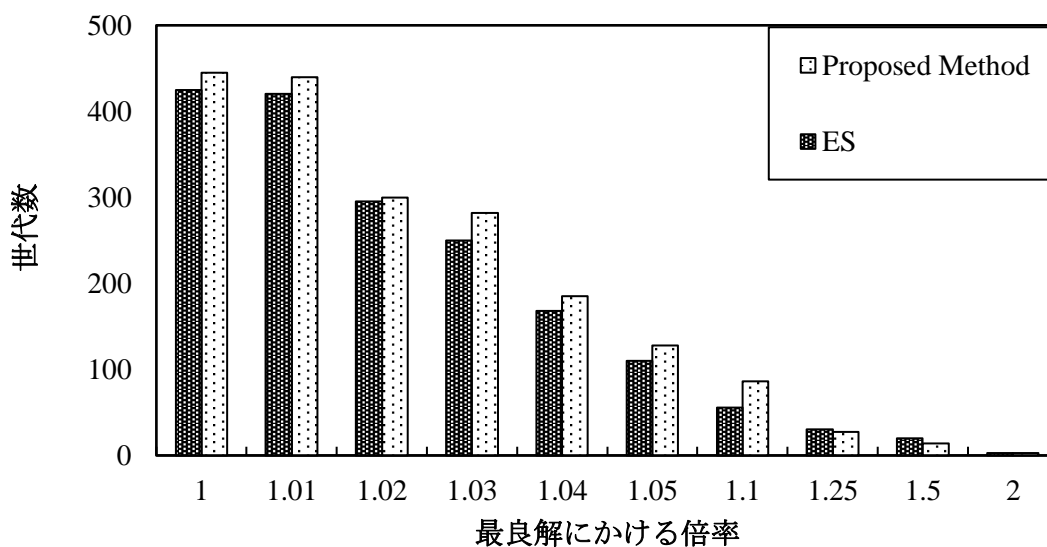


図 3.17 最良解にかける倍率による収束世代数

以下に示す「最良値」は、最終世代における集団中の最良適応度のことである。また、「平均値」「最小値」は同じ実験を乱数のシードを変えて 10~30 回行ったときの値を示している。

なお、以下の表 3.14、3.16、3.19 について得られた結果の有意検定を行った。ここで

は正規分布を仮定しないノンパラメトリック法の Wilcoxon signed rank test (95%信頼度) [Hothorn 02]を用いた。表 3.14 と表 3.16 については two-way で最良解と 2 番目に良い解の間に有意差があるかどうかを検定した。表 3.19 については提案手法が最も良かった場合に one-way で 2 番目に良い解との間に有意差があるかどうかを検定した。有意差が確認された場合には表中の平均値に\*印を付けた。

本研究では、CPU は Intel Core i5 (2.70GHz)、プログラミング言語は Visual Studio 2010 の C++を用いた。また、計算時間を表 3.10 に示す。

**表 3.10 提案手法の計算時間 (30 試行の平均)**

問題	計算時間 (h)
O7	0.13
O8	0.25
O9	0.31
Ba12	0.28
Ba14	0.36
AB20	1.1
SC30	3.4
SC35	4.3
Du62	17



### 3.2.2 EA における突然変異の回数

表 3.11 突然変異の回数を変えたときの適応度の最良値

(下線は, その問題に対する最小値を示す)

問題	1 回	2 回	3 回	4 回	5 回
O8	284	<u>253</u>	286	254	257
O9	<u>234</u>	259	283	252	263
Ba12	6523	6124	6184	<u>6031</u>	6769
AB20	4975	<u>4811</u>	5316	4988	5240
SC30	<u>3464</u>	3549	4048	3722	3781
SC35	<u>3507</u>	3904	4100	3951	4007
Du62	3472525	<u>3362111</u>	3464078	3366344	3563044
加重平均	1.039	<u>1.037</u>	1.120	1.046	1.092

EA-fix について探索性能と突然変異の回数の関係を調べた。表 3.11 は突然変異を実行する回数を変えたときの最良値の 10 回の平均を示している。下線を付けた値は, その問題に対する最小値である。最終行を単純平均ではなく加重平均とする。各問題の重みはその問題に対して、表 3.11 中の最小値の逆数とした。また、表 3.12 は交換回数 6、7 回の平均値も示す (SC35 と Du62 は 5 回、他の問題は 10 回)。6 回と 7 回もやってみたが、予想通り、5 回より悪い結果が得られた。表 3.11 から、以下のことがわかる。

- 各問題によって最適な突然変異の回数が異なる。
- 全問題の加重平均は、突然変異の回数が 2 回るとき最小となる。

従って、以下の実験では EA-fix の突然変異の回数を 2 回に固定する。

表 3.12 突然変異の回数が 6、7 の結果

問題	6 回	7 回
O8	264	279
O9	268	265
Ba12	6419	6649
AB20	5642	5954
SC30	4590	3508
SC35	3861	3649
Du62	3457825	3544281

### 3.2.3 交換回数の計算に用いた分布関数の比較

提案手法では突然変異の回数を計算するときに Levy 分布を用いている。本節では、この分布を一様分布または正規分布に変更したときの探索性能について考察する。このため、EA-fix, EA-uniform, EA-normal, EA-levy の比較実験を行った。ここで、一様分布の範囲は [1, 5]、正規分布の確率密度関数は式(3.4)とした。

$$P(l) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{l^2}{2}\right) \quad (3.4)$$

図 3.18 は問題 SC30 に各手法を適用したときの進化の過程を示している（30 試行の平均値）。横軸は世代数、縦軸は平均最良値を示す。各手法とも 650 世代以下で収束している。

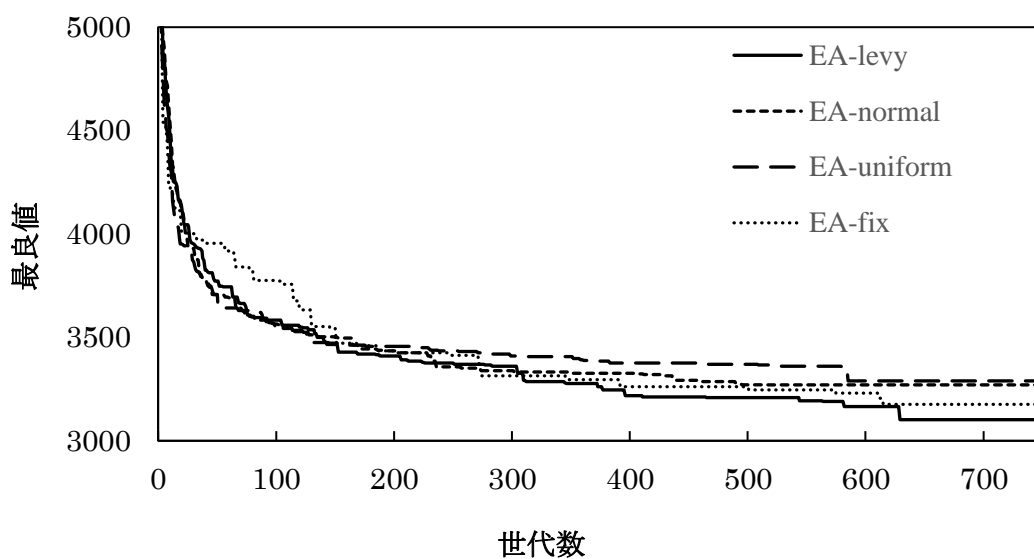


図 3.18 進化過程の例 (SC30)

表 3.13 は一様分布範囲[1-10]をやった実験結果である (SC35 と Du62 は 5 回,他の問題は 10 回)。実験結果から、[1,5]の方が良かったので、こちらを採用した。

表 3.13 一様分布範囲[1-10]の結果

問題	平均値	最小値
O7	132	125
O8	256	180
O9	219	203
Ba12	6761	5557
Ba14	3680	3389
AB20	4855	4471
SC30	3251	3050
SC35	3332	3094
Du62	3312891	3156488

表 3.14 は、各手法で得られた最良値の 30 回の平均値と最小値を示している。表 3.14 から以下のことがわかる。

- 職場数が 14 以下の小規模な問題に関しては、EA-levy と EA-normal がそれぞれ O7 ならびに O9 と Ba14 に対して平均値と最小値の両方で最も良い結果を与えている。また、EA-fix と EA-normal に関しては、それぞれ O8 と O9 で有意差が認められる。
- 職場数が 20 以上の大規模な問題に関しては、EA-Levy が平均値と最小値の両方で最も良い結果を与えている。特に、AB20、 SC35、 Du62 で有意差が認められる。

次に、30 回の実験の中で目標解  $S_{1,02}$  を発見できた回数の割合、ならびに発見できた場合の世代数の平均値をそれぞれ表 3.15 と表 3.16 に示す。これらから次のことがわかる。

- 提案手法 EA-Levy は、O9 以外のすべての問題に対して、他の手法よりも同等以上の高い割合で目標解を発見することができる (表 3.15)。
- EA-fix は、比較的小規模な問題 (O8、O9、Ba12) に対して、他の手法よりも少ない世代数で目標解を発見することができる (表 3.16)。
- 提案手法 EA-Levy は、職場数が 14 以上の大規模な問題に対して、他の手法より少ない世代数で目標解を発見することができる。特に、SC30 と SC35 で有意差が認められる (表 3.16)

表 3.14 交換回数の計算に用いた分布関数の比較結果

(mean と min は 30 試行の平均値と最小値, \*印は有意差があることを示す)

問題	EA-fix		EA-uniform		EA-normal		EA-levy (提案手法)	
	mean	min	mean	min	mean	min	mean	min
O7	126	123	128	120	126	119	<u>125</u>	<u>117</u>
O8	<u>232*</u>	226	252	<u>172</u>	256	173	254	176
O9	214	204	220	201	<u>211*</u>	<u>196</u>	223	214
Ba12	<u>6080</u>	5480	6507	5752	6289	5823	6389	<u>5357</u>
Ba14	3313	3110	3370	3221	<u>3211</u>	<u>2904</u>	3232	2976
AB20	4795	4290	4502	4321	4441	4314	<u>4429*</u>	<u>4270</u>
SC30	3176	2978	3290	3065	3270	3047	<u>3102</u>	<u>2961</u>
SC35	3849	3033	3321	3014	3599	3019	<u>2980*</u>	<u>2908</u>
Du62	3378234	3134567	3263900	3172187	3361359	3059486	<u>3120521*</u>	<u>2838576</u>

表 3.15 目標解を発見した割合 (%)

問題	EA-fix	EA-uniform	EA-normal	EA-levy
O7	100	100	100	100
O8	100	100	100	100
O9	97	97	<u>100</u>	93
Ba12	87	83	93	<u>97</u>
Ba14	70	50	70	<u>80</u>
AB20	87	80	87	<u>93</u>
SC30	<u>50</u>	40	40	<u>50</u>
SC35	37	33	37	<u>50</u>
Du62	30	30	33	<u>37</u>

表 3.16 目標解を発見するまでの世代数の平均値

問題	EA-fix	EA-uniform	EA-normal	EA-levy
O7	226	220	223	<u>219</u>
O8	<u>212</u>	223	214	219
O9	<u>220</u>	240	250	237
Ba12	<u>233</u>	244	256	241
Ba14	294	373	334	<u>280</u>
AB20	250	260	264	<u>246</u>
SC30	400	451	460	<u>343*</u>
SC35	353	318	394	<u>221*</u>
Du62	846	880	850	<u>820</u>

以上の実験結果から、Levy 分布は探索性能（表 3.14、3.15）と探索速度（表 3.16）の両面について、大規模の問題に対して特に有効であるといえる。

### 3.2.4 Levy Flight の有効性

Levy 分布は、 $l=1$  付近の乱数を多数生成しつつ、他の分布に比べて  $l$  が大きい乱数をより多く生成することができるので、大域的探索（一個体当たりの突然変異の回数が多いこと）が重要となる大規模問題に対して良い性能が得られたと考える。低次元問題と高次元問題の代表として、O8 と AB20 二つ問題を選び、スケールパラメータを 1 の半分と 2 倍を設定して検証実験を行った。結果は下の表 3.17 で示す。結果から見ると 5 回の実験でスケールパラメータは 0.5 の時、1 と 2 の時より悪い解を得られた。標準偏差も考え入れると、O8 の問題に対して、1 と 2 の時、範囲も近いである。スケールパラメータを変換しても改善があまりなかったことを確認した。

表 3.17 スケールパラメータを変換した結果

問題	O8			AB20		
	0.5	1	2	0.5	1	2
スケールパラメータの値	0.5	1	2	0.5	1	2
5 回平均	256	243	246	4427	4306	4409
標準偏差	9	11	12	41	23	61

前節に示したとおり、提案手法は職場数 20 以上の大規模問題に対しては、すべての実験結果で他の手法より優れた性能を示したが、14 以下の小規模問題では他の手法の方が良い場合も見受けられた。この理由を考察するために、大規模問題として AB20、小規模問題として O9 を取り上げて以下の実験を実施した。これらの問題は、表 3.14 においてそれぞれ EA-Levy と EA-normal が平均値と最小値の両方で最も良い性能を示しており、また 2 番目に良い解との差に統計的有意差が認められている。

表 3.18 は、提案手法において、探索の各世代で生成された子個体の中で次世代に生き残った個体の数をその個体が受けた交換の回数 ( $m$ ) ごとに示している。表 9 から以下のことがわかる。

- 両問題とも、 $m$  が小さい個体の方が次世代に生き残る割合が高い傾向にある。特に  $m=1, 2$  の個体数が多い。
- AB20 では、 $m > 2$  の個体数が世代とともに増加している。また、O9 と比較すると 200 世代では 4 倍となっている。
- O9 では、 $m > 2$  の個体数が世代とともに増加する傾向は見られない。また、表中の全世代において、AB20 以下となっている。

以上のことは、大規模な AB20 においては、探索の序盤では小さな  $m$  で近傍の良い解を生成し、局所解に陥ると大きな  $m$  で大域的に探索して良い解を得ていると解釈できる。これより、2.4 節で述べた LF 採餌仮説が実現できていると考える。また、小規模な O9 では、探索の終盤まで小さな  $m$  が良い解を生成しており、大域的な探索の有効性が AB20 に比べて少なかったことがわかる。この結果、裾の重い分布を用いた EA-Levy よりも裾が指数関数的に減少する分布を用いた EA-normal の方が、良い結果を示したと思われる。

表 3.18 各世代で生成された個体の中で次世代に生き残った個体数

(1) 問題 AB20

世代		1	10	50	100	200
$m$	1	35	34	35	35	27
	2	3	6	6	3	9
	3	1	1	0	3	5
	4	1	1	0	1	3
	5	1	0	1	1	1
	6	0	1	1	0	1
	7	0	1	0	0	0
	8	0	0	1	2	0
	9	0	0	2	1	1
	10	1	1	1	0	1
$m>2$ の個数		4	5	6	8	12

(2) 問題 O9

世代		1	10	50	100	200
$m$	1	31	29	40	39	36
	2	5	3	6	9	9
	3	3	2	1	1	2
	4	0	2	1	1	1
	5	0	1	0	0	0
$m>2$ の個数		3	5	2	2	3

### 3.2.5 従来手法との比較

職場数が 7~62 の問題に対して、従来手法の文献の値、提案手法と同じコード化を用いた GA、ならびに提案手法で得られた最良値の平均値と最小値（実験回数 30 回）を比較した。表 3.19 に比較結果を示す。図 3.19 は問題 AB20 と O9 の実行例である。



表 7 から次のことがわかる。

- 提案手法は、職場数が 9 以上のすべての問題に対して、他の手法よりも最良値の平均値が小さい。また、その中で SC30 以外のすべての問題で有意差が認められる。
- 提案手法は、O7、O9、AB20 を除くすべての問題に対して、他の手法より最良値の最小値が小さい。

従って、提案手法は大規模な問題に対して従来手法より有効である可能性が高いといえる。

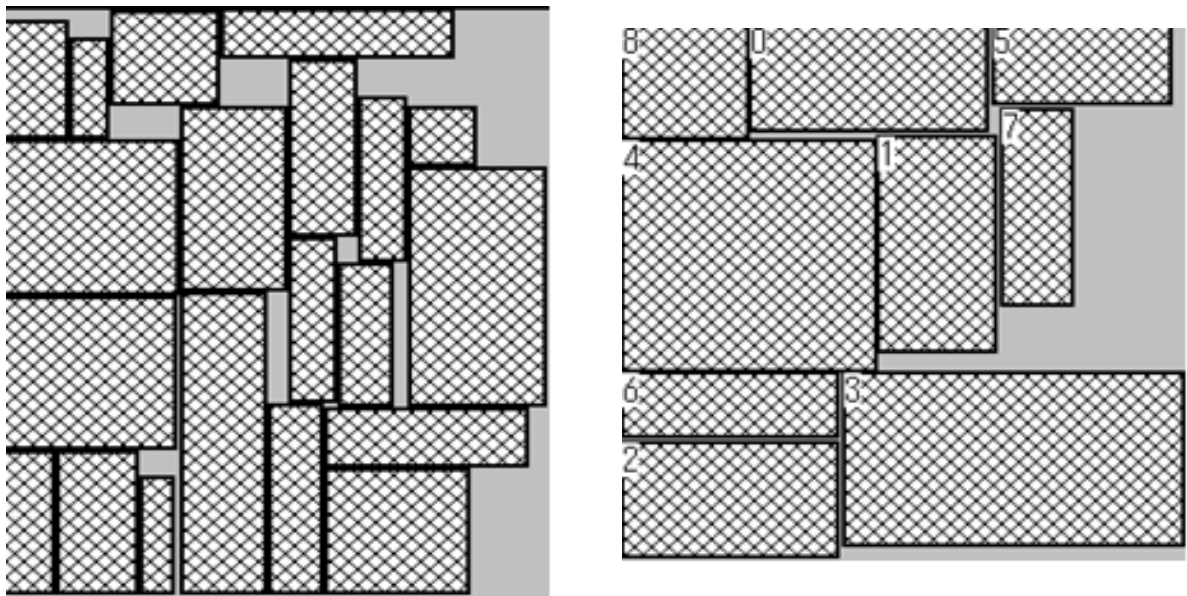


図 3.19 AB20 と O9 の実行例

### 3.3 まとめ

この部分は FLP に対する新たな解法を提案した。提案手法は、EA における探索空間上の解の移動に Levy Flight を用いたものである。これにより、探索点の近傍だけではなく、遠方に移動することができるので、局所最適解から脱出することができると考えている。職場数が 62 までのベンチマーク問題を用いた実験結果は、Levy 分布に基づく探索は、一様分布や正規分布に基づく探索よりも大規模な問題に対して性能が高いこと、ならびに、提案手法は GA や ACO などの従来手法より大規模な問題に適している可能性が高いことを示唆している。

Levy Flight は、連続値最適化問題に対する有効性は報告されているものの、組合せ最適化問題への応用はあまり検討されていない。今後は、GA などの他のメタヒューリスティックと組み合わせて、有効な応用分野を明確にすることが重要であると考えられる。また、本論文では LF 採餌仮説を満たす分布関数として標準 Levy 分布を用いたが、提案手法を応用問題に適用する場合は、他の分布関数や一般の Levy 分布について比較検討することも重要であると考えられる。

表 3.19 提案手法と従来手法との比較

問題	GA	力学モデル	GA		提案手法	
	[Liu 07]	[大森 13]	(提案手法と同じコード化)			
	mean	mean	mean	min	mean	min
O7	-	227	135	130	125	117
O8	-	420	257	213	254	<u>176</u>
O9	-	372	236	205	<u>223*</u>	214
Ba12	8702	-	7379	6320	<u>6389*</u>	<u>5357</u>
Ba14	4852	-	4755	4322	<u>3232*</u>	<u>2976</u>
AB20	5668	5189	4555	4094	<u>4429*</u>	4270
SC30	3707	3106	3856	3295	<u>3102</u>	<u>2961</u>
SC35	3604	3045	4199	3223	<u>2980*</u>	<u>2908</u>
Du62	-	-	3401429	3145645	<u>3120521*</u>	<u>2838576</u>

問題	GA	ACO		多目的 GA		多目的 GA	
	[Goncalves 15]	[Komarudin 10]		[ Kazi 11]		[ Kazi 13]	
	mean	mean	min	mean	min	mean	min
O7	132	132	132	111	99	<u>102</u>	<u>99</u>
O8	<u>243</u>	<u>243</u>	243	293	203	270	203
O9	237	236	236	445	<u>202</u>	294	202
Ba12	8021	8275	8252	8712	8104	8591	7982
Ba14	4687	4729	4725	5500	4791	5397	4712
AB20	5021	5182	4973	6131	<u>4015</u>	5856	3978
SC30	3368	3985	3868	4444	3740	4146	3707
SC35	3385	4231	4132	4306	3836	4190	3637
Du62	-	3728791	3720521	3220771	2977513	3218854	2975604

## 第四章 室内レイアウト問題への適用

### 4.1 提案手法

#### 4.1.1 コード化

図 4.1 に染色体の例を示す。遺伝子座（変数  $X$ ）を配置順とする。また、遺伝子（領域  $D$ ）を部屋名または家具番号と家具の向きとのペアとする。この例では、家具 1、6、5、3 をこの順番で部屋 A に配置し、次に家具 2、4 をこの順番で部屋 B に配置することを意味している。また、配置するときは、家具番号の下に記述された向きに配置する。家具の向きは、家具本体(白い部分)にワークスペース(掛け線の部分)を含めて図 4.2 のように設定している。

配置順	1	2	3	4	5	6	7	8
部屋と家具	A	1	6	5	3	B	2	4
家具の向き	-	左	右	上	下	-	左	上

図 4.1 染色体の例(1~6 は家具番号、A と B は部屋名)

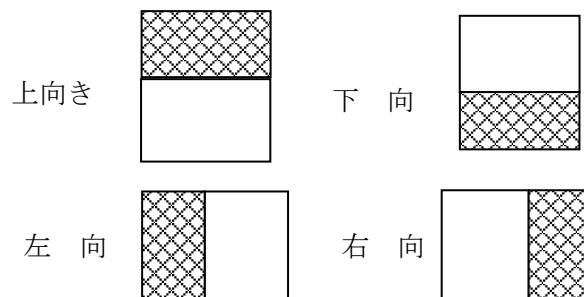


図 4.2 家具の向き

実際のレイアウトを設計するときは、何も置いていないスペースの配置が重要である[秋川 13]。本論文では部屋のサイズと家具の合計面積からスペースの全面積を計算し、小さく分割して単位スペースとして家具の一種と見なし、遺伝子に加えることとする。ただし、スペースの向きは考慮しない。単位スペースの種類と大きさは、文献[秋川 13]を参考に図 4.3 のように設定した。

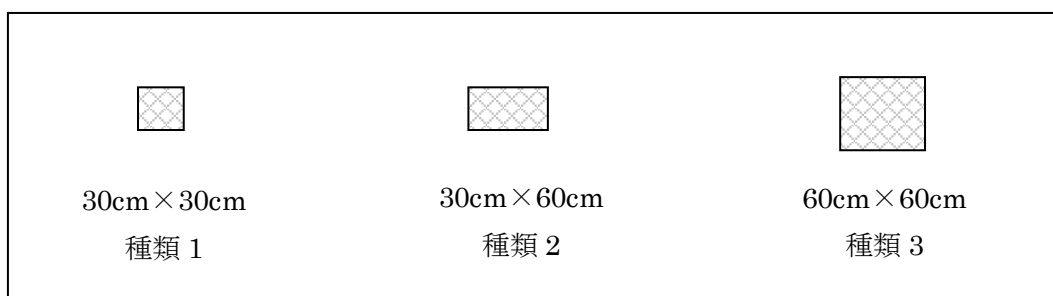


図 4.3 単位スペースの種類と大きさ

#### 4.1.2 デコード化

本論文では個体の集団を部分集団（島）に分割し、各島の染色体に対して島ごとに異なるデコード化法を適用することにより多様なレイアウトを生成する。ここでは一つの染色体に対して、家具を配置する方向を変えることにより、異なるレイアウトを生成することを考える。

遺伝的アルゴリズム（GA）において集団の多様性を維持する方法の一つとして、個体集団をいくつかの独立した部分集団（島）に分ける島モデル GA が提案されている。島ごとに通常の GA を用いて進化を進めていき、数世代置きに各島の間で個体を交換する「移住」という方法で遺伝情報の交換が行われる。ある島が一様になっても、他の島の中に別の形質を持った個体が存在すれば、その個体と遺伝情報をやりとりすることで、多様性が維持される可能性は残っている。移住操作は、適応度の評価や遺伝的演算に比べ計算量が小さいので、大きな効果を期待することができる。また、集団を複数に分け

ることで、島ごとに計算機を割り当てる並列化が容易に実現できる。しかし、移住操作を行う間隔を短くすると集団全体での多様性が維持しづらく、集団を一つにした場合と差が無くなってしまう。逆に、移住の間隔を長くし過ぎると進化に遅延が生じるという問題点がある。

デコード化方法を決定するため、大森らの論文[大森 11]に示されているベンチマーク問題を用いて、島ごとに異なるデコード化法 GA を用いて論文で乗った従来手法との比較実験を行った。こちらの問題は家具の向きなどを考慮しないかつ各家具の長さとは幅は定数である。表 4.1 と表 4.2 はベンチマーク問題の属性と実験のパラメータを示す。

表 4.1 問題の属性

問題	部屋の大きさ	職場数
Y10	4m×4m	10
Y15	5m×5m	15

表 4.2 実験用のパラメータ

パラメータ	設定値	単位
突然変異率	25	%
島ごとの個体数	500	個
評価回数	30000	回
試行回数	20	回

代表例として、各島における配置する順番を示しています。島 1 は左上から右下、島 2 は右上から左下、島 3 は左下から右上、島 4 は右下から左上の順番に配置した。

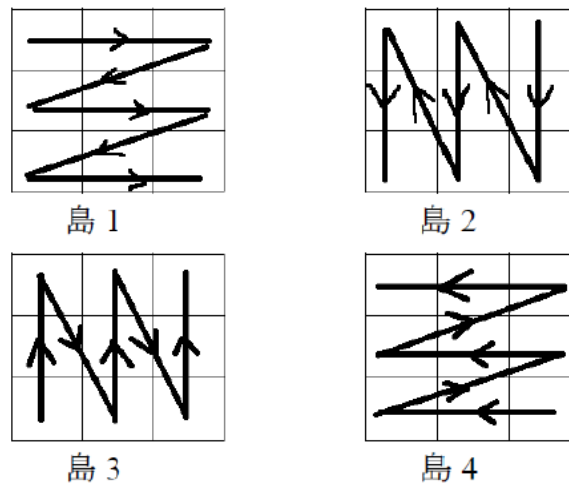


図 4.4 配置例

問題 Y10 と Y15 の最終世代における mean、best を表 4.3 に示している。RCGA は大森らの実数値 GA、SA は Tam のやきなまし手法である。括弧内は標準偏差である。島ごとに異なるデコード化法 GA と RCGA を比較すると、標準偏差を考えると、問題 Y10 はほぼ同じですが、Y15 が優れている。次に本手法と SA を比較すると、すべての場合で本手法は優れていることがわかった。

表 4.3 実験結果

問題	結果	本手法				従来手法	
		島 1	島 2	島 3	島 4	RCGA [大森 11]	SA [Tam 92]
Y10	Mean	35173 (2137)	35004 (1874)	35145 (2463)	34828 (1026)	40801	45728
	Best	33037 (235)	33091 (241)	33045 (252)	33083 (238)	33172	37792
Y15	Mean	127476 (8934)	126884 (9287)	127184 (9062)	124057 (9147)	140792	156832
	Best	103471 (3296)	103369 (3048)	103589 (2875)	103169 (3524)	106661	109672

図 4.5 は問題 Y10 の、各島における最良解のレイアウトの一例である。括弧内は各島の染色体（配置順）を示す。島 1、4 は同じ順番、島 2、3 は同じ順番となった。つまり、島 1 と島 4、島 2 と島 3 は同じレイアウトになっていた。同じレイアウトになっている理由は、施設が正方形であることと問題サイズ小さいことが考えられる。



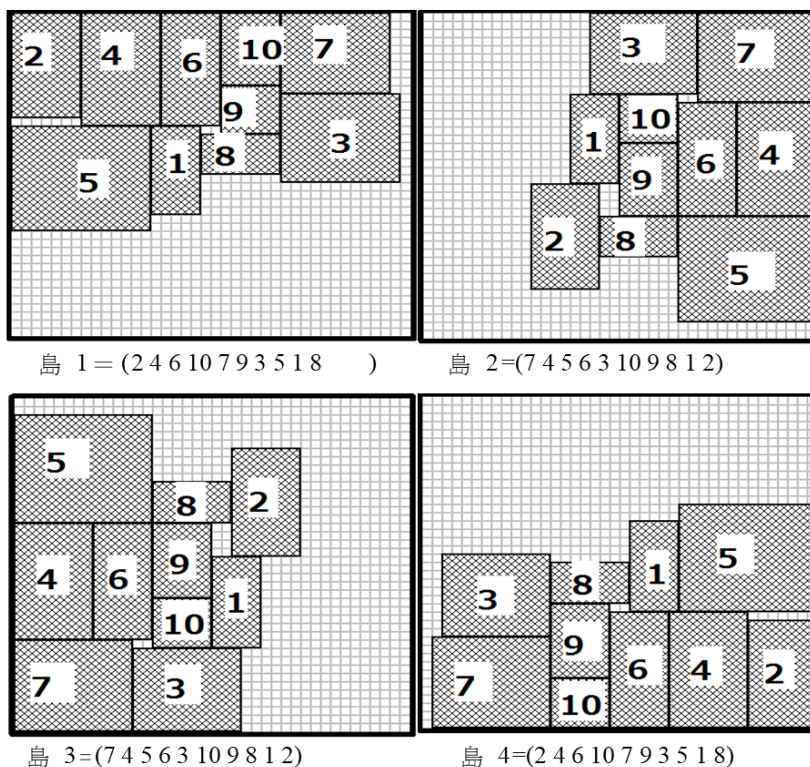


図 4.5 問題 Y10 の、各島における最良解のレイアウト

配置する方向としては、右方向 ( $r$ )、左方向 ( $l$ )、上方向 ( $u$ )、下方向 ( $d$ ) の 4 方向及びこれらの組合せとする (斜め方向への配置は考えない)。図 4.6 は、同じ大きさの家具を正方形の部屋に、染色体 9534162087 に従って配置する例を示している。図 4.6 の (a)配置方向  $rrr$  は、部屋の左上から始めて右方向に家具を配置し、はみ出す場合は左端に戻って上詰めで配置することを表している。この例では、 $r$  を 3 行に渡って実行しているので、 $rrr$  と記述することにする。

このように定義すると、例えば 4 行 (または列) 配置する場合は、全部で  $4^4=256$  通りの組合せが存在することになる。しかし、以下に示すように、組合せによっては類似のレイアウトが生成されるケースが存在する。このようなケースは本論文の目的に合わないので、除外して考えることとする。

まず、図 4.6 の  $rrr$  と  $lll$ 、ならびに  $uu$  と  $dd$  を比較すると、レイアウトが対象形とな

っており、実質的に同じレイアウトを生成している。次に、染色体 9534162087 と 9534026187 のように、下線部が逆位となっている2つの染色体に対して、それぞれ *rrr* と *rlr* を適用した場合、類似したレイアウトが生成されることが予想できる (図 4.6(a)と 図 4.7 を参照)。

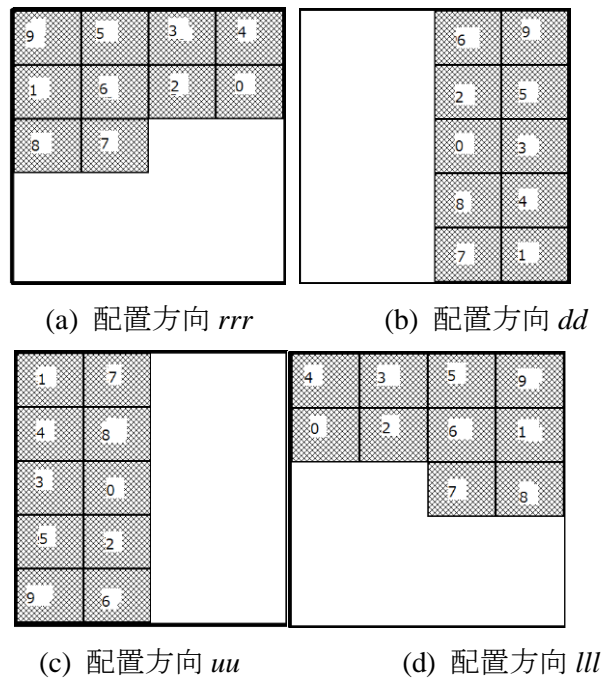


図 4.6 配置方向が異なる場合の配置図の例  
(染色体 : 9 5 3 4 1 6 2 0 8 7)

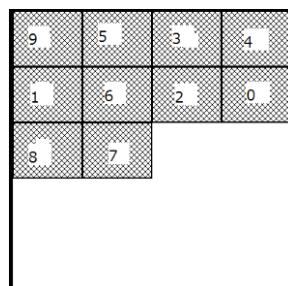


図 4.7 染色体: 9 5 3 4 0 2 6 1 8 7 に対して配置方向 *rlr* を適用した場合の配置図

回転対称や鏡対象を考慮して、本論文では、水平方向 ( $h$ ) と垂直方向 ( $v$ ) の2方向のみを考えることとする。さらに、 $hhhv$  と  $hhvh$  のように、同じ部分構造を持つ配置方向も類似のレイアウトを生成する可能性がある。そこで本論文では、 $h$  と  $v$  の出現確率によって場合分けすることにより、効率的に多様なレイアウトを生成することを目指す (表 4.4 参照)。

表 4.4 出現確率による配置方法の分類 (4 行/列の場合)

分類	$h$ の出現確率	$v$ の出現確率	代表例
1	1	0	$hhhh$
2	0.75	0.25	$hhhv$
3	0.5	0.5	$hhvv$
4	0.25	0.75	$hvvv$
5	0	1	$vvvv$

#### 4.1.3 適応度関数

得られたレイアウトに対して、違反している制約の重み (違反点数) の合計を適応度関数とする。表 4.5 に示した通り、各制約は家具に関する制約と部屋に関する制約に分類されている。前者については、家具ごとに違反点数を計算してその合計を求める。後者については、部屋ごとに違反点数を計算してその合計を求める。そして両者の合計をそのレイアウト (個体) の適応度とする。

#### [定義]

(1)ある家具  $F_i$  (またはスペース)の配置が制約に違反しているとき、その制約の違反点数をその家具に加算する。これを次のように行う。

$$conf(i, C_k) = \begin{cases} w_k & F_i \text{ の配置が制約に違反} \\ 0 & \text{違反なし} \end{cases}$$

$k$  は制約条件の番号である。

## (2) $C_k$ の違反点数

$C_k$  反している違反点数は次のように計算する。 $i$  は制約条件の和である。

$$W_i = \sum_{k=1}^i conf(i, C_k)$$

## (3) 総合違反点数

各個体  $i$  (染色体) の制約違反の程度を表す。 $j$  は制約の個数である。この総合違反点数を最少化することが本システムの目的である。

$$W_{total} = \sum_{i=1}^j W_i$$

## [計算方法]

- [C<sub>1</sub>]: 家具は室内に配置 (20)

すべての  $F_i$  に対して、配置図上に配置しなければ、 $w_i=20$  になる。

- [C<sub>2</sub>]: 窓をふさぐ配置を行なわない (1~12)

① 窓の前 (90cm 以内) に  $F_i$  が配置された場合に制約違反。

$w_2 = F_i$  が窓の前に配置されている面積 / 窓の前の総面積  $\times$  cost

② 窓の高さの判断

if ( $F_i$  の高さ  $\geq$  部屋  $i$  の窓の高さ)

cost +=  $w_2$ ;

- [C<sub>3</sub>]: 入り口付近にスペースがある (10)

ドアのまえ 90cm の範囲に家具があったら(スペースを除く)違反する。

- [C<sub>4</sub>]: 高さの高い家具は壁際に配置 (12)

高さの高い家具は壁際に対する距離  $Q$  を計算する。

$Q=0$  の時、制約を違反しない。

- [C<sub>5</sub>]: 家具のある場所と向きがあっている (10)

家具は部屋の中心に向かわないと、違反する。

- [C<sub>6</sub>]: 配置する部屋が希望とあっている (15)

問題設計リストのデータを参照して、家具が配置する部屋に配置しないと、違反する。

- [C<sub>7</sub>]: 電気器具はコンセントに近いところに配置する(1~8)

左上の座標から近い方のコンセントまでのマンハッタン距離を測る、270cm 以上なら、制約を違反する。

- [C<sub>8</sub>]: 高さの近い家具は近くに配置する (5)

if ( $|$ 家具  $i$  の高さ - 家具  $j$  の高さ  $| > 10$ cm)

スペースを埋めて行く。

- [C<sub>9</sub>]一つ部屋の広さに対する家具の占有率は3分の1を超えない(5)

$if(\text{部屋の面積} \times 1/3 - \text{この部屋の家具の総面積} < 0)$

制約を違反する。

- [C<sub>10</sub>]出入口と各家具とを結ぶ通路が存在する(15)

もし出入口と各家具の距離は0となる時、スペースを埋めて行く。

表 4.5 制約条件

分類	制約	制約の内容	違反点数
家具	C <sub>1</sub>	家具は室内に配置する	20
	C <sub>2</sub>	窓をふさぐ配置を行わない	1~12
	C <sub>3</sub>	入り口付近にスペースがある	10
	C <sub>4</sub>	高さの高い家具は壁際に配置する	12
	C <sub>5</sub>	家具の位置と向きが合っている	10
	C <sub>6</sub>	配置する部屋が希望と合っている	15
	C <sub>7</sub>	電気器具はコンセントに近いところに配置する	1~8
	C <sub>8</sub>	高さの近い家具は近くに配置する	5
部屋	C <sub>9</sub>	一つ部屋の広さに対する家具の占有率は3分の1を超えない	5
	C <sub>10</sub>	出入口と各家具とを結ぶ通路が存在する	15

#### 4.1.4 遺伝の操作

交叉としては、遺伝子が順列を表現する場合に広く用いられている OX(order crossover)を採用する。突然変異は、遺伝子の交換とする。すなわち、ランダムに選択した2つの遺伝子座の遺伝子を交換する(図 4.9)。本手法では、この交換の回数を LF によって決定する。また、世代交代モデルとしては、多様性の維持に優れているといわれている MGG(Minimal Generation Gap)モデル[佐藤 96]を用いる。

本研究では OX 交叉方法を導入する。実行の流れを図 4.8 に示す。

step1 親 1(親 2)の交叉ポイント前の遺伝子を子個体にコピーする。

step2 親 2(親 1)の交叉ポイント後の遺伝子で子 1(子 2)に使われていない遺伝子を親 2(親 1)と同じ遺伝子座にコピーする。

step3 子 1(子 2)に格納されていない遺伝子を親 2(親 1)の遺伝子座の小さいほうから順に探し格納する。

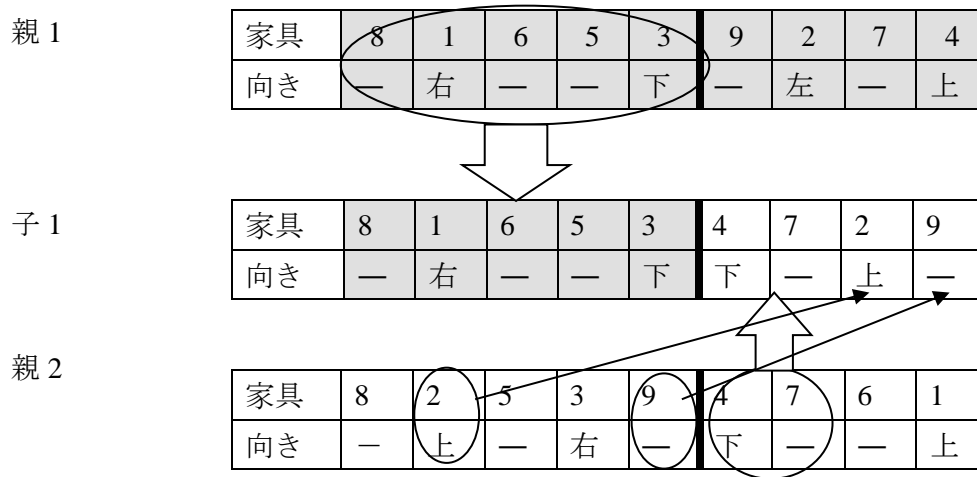


図 4.8 交叉方法の例

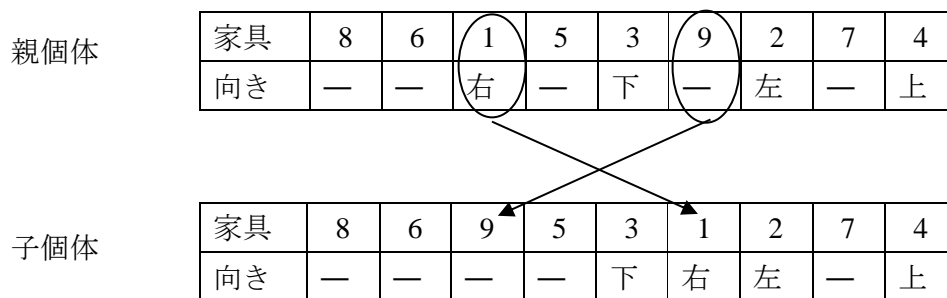


図 4.9 突然変異の例

#### 4.1.5 解の更新方法

LF に基づく探索手法は、一般には、変数が連続値をとる最適化問題に適用されている。しかし、室内レイアウト問題の遺伝子は離散値であるため LF を直接適用することは難しい。本論文では、著者らが EA に対して提案した以下の方法を用いる。

まず、確率密度関数として式(4.1)の Levy 分布[四辻 10]を用い、移動距離  $l$  をこれに従う Levy 乱数として生成する。

$$P(l) = \frac{1}{\sqrt{2\pi}} l^{-\frac{3}{2}} e^{-\frac{1}{2l}} \quad (4.1)$$

次に、個体間の距離を以下のように定義する。一つの個体 A に交換を  $M$  回施して生成した個体を B とし、 $M$  を A-B 間の距離と見なす。すなわち、交換の回数を探査空間上の移動距離とする。以上より、突然変異による解の更新を式(4.2)の  $M$  を用いて行う(交換を  $M$  回実行する)。ただし、 $n$  は染色体の長さ、 $[l]$  は  $l$  の整数部分を表わす。

$$M = \begin{cases} [l] + 1 & \text{if } l < \frac{n}{2} \\ 0 & \text{otherwise} \end{cases} \quad (4.2)$$



#### 4.1.6 アルゴリズム

本手法では、複数の島からなる島モデル（移住なし）を用いる。各島では、染色体に対して遺伝的演算を行う。適応度を計算するときは、以下の手順でレイアウトを生成する。

- Step1. 家具を配置する順番を染色体から読み取る。
- Step2. 図 2 に示した矢印の順番（島毎に異なる）に従って空いているメッシュを探す。
- Step3. 家具の重複と施設からのみ出しが生じなければ、そのメッシュに配置する。
- Step4. 各家具の座標を記録する(ファイルに書き込む)。
- Step5. 各家具の距離を計算する。
- Step6. 家具間のコストファイルを読み込む。
- Step7. 家具間の総運搬コストを計算する。

個体の適応度を計算するときは、表 2 の分類に従って、島ごとに異なるデコード化を適用する。各島におけるアルゴリズムを図 4.10 に示す。

```
初期集団をランダムに生成して適応度を求める;  
while (指定世代まで)  
{  
    集団からランダムに 2 個体選択する;  
    OX により交叉を行う;  
    Levy 乱数  $l$  を生成する;  
    突然変異 (交換) を  $M$  回に実行する;  
    子個体のレイアウトを生成し適応度を求める;  
    生存選択により次世代の個体を選択する;  
}
```

図 4.10 提案手法の各島のアルゴリズム

## 4.2 室内レイアウト問題の評価実験

### 4.2.1 実験方法

提案する手法の有効性を確認するため、次の手法との比較実験を行った。

- **ES** : 突然変異を交換とした通常の $(\mu+\lambda)$ ES。
- **GA** : 図 8 のアルゴリズムにおいて、LF は用いないで突然変異の回数を固定した 1 集団の GA。
- **GA+LF** : 図 8 のアルゴリズムを用いた 1 集団の GA。

各種パラメータの値は、予備実験により表 4.6 ように設定した。ES と GA の集団サイズ = 1000、提案手法の各島の集団サイズ = 200 (島が 5 つあるので、全個体数は 1000)、ES の子集団サイズ  $\lambda = 3000$ 、GA の突然変異率 = 10%、また、ES と GA で、1 回の突然変異として交換をそれぞれ 2 回と 1 回適用した。以下の表に示すデータは、乱数の初期値を変えて 30 回実験を行ったときの最終世代における最良適応度の平均と標準偏差、ならびに 30 回中の最良値である。なお、混乱を避けるため、以下では適応度のことを違反点数と称す。

表 4.6 パラメータ

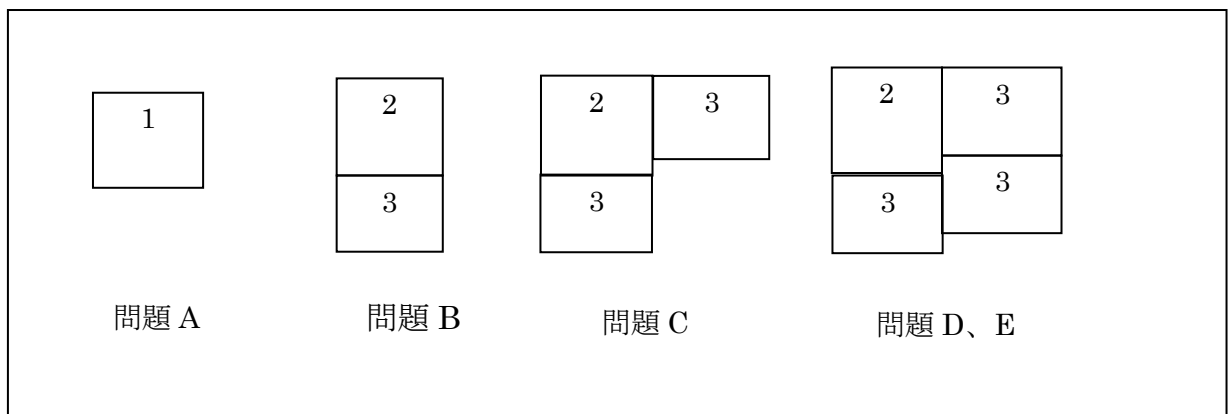
パラメータ	ES	GA	GA+LF
集団サイズ ( $\mu$ )	1000		
子集団サイズ ( $\lambda$ )	3000	-	
子集団の各個体突然変異の操作回数( $M$ )	2	1	Levy 分布に従う生成した乱数の値
世代数	400		
突然変異率	-	0.1	
実行回数	30		

#### 4.2.2 室内レイアウト問題の設計

対象問題の属性を表 4.7 に示す。部屋のサイズと構造、および使用した家具の属性は図 4.11 と表 4.7 に示す。これらの値は、市販されている家具と住宅のカタログを参考に設定した。

表 4.7 対象問題の属性と染色体長

問題	部屋数	家具数	単位スペース数	染色体長
A	1	8	20	28
B	2	13	37	50
C	3	17	48	65
D	4	25	51	76
E	4	30	54	84



\* (部屋 1 のサイズ : 350cm×350cm(約 8 畳) ; 部屋 2 のサイズ : 340cm×340cm ;  
部屋 3 のサイズ : 240cm×340cm ; )

図 4.11 部屋の構造

表 4.8 家具の属性

家具番号	家具名前	長さ	横幅
1	ソファ	173	85
2	センターテーブル	80	55
3	ローボード	79	35
4	アッパーライト	21	21
5	チェスト1	81	35
6	電話	20	20
7	テレビ	60	33
8	ベッド (S)	197	93
9	ナイトテーブル	42	30
10	洋服タンス	80	52
11	ドレッサー	80	40
12	ベッド (SD)	196	120
13	ナイトテーブル	42	30
14	テーブルランプ	24	24
15	洋服タンス	80	52
16	チェスト2	70	40
17	デスク	100	50
18	チェア	58	58
19	本棚	89	29
20	パソコン	37	26
21	カラーボックス	42	30

A) 1 部屋問題

部屋サイズ :  $350 * 350\text{cm}^2$

家具数 : 8

表 4.9 問題 A の家具

家具番号	名前
1	ソファ
2	センターテーブル
3	テレビ
4	ベッド (S)
5	ナイトテーブル
6	チェスト
7	電話
8	パソコン

B) 2 部屋問題

部屋サイズ :  $340 * 340\text{cm}^2$ 、 $240 * 340\text{cm}^2$

家具数 : 13

表 4.10 問題 B の家具

家具番号	名前
1	ソファ
2	センターテーブル
3	テレビ
4	ベッド (S)
5	ナイトテーブル
6	チェスト
7	電話
8	パソコン
9	ナイトテーブル
10	デスク
11	ワークチェア
12	本棚
13	デスクライト

C) 3 部屋問題

部屋サイズ :  $340 * 340 \text{ cm}^2$ 、 $240 * 340 \text{ cm}^2$ 、 $240 * 340 \text{ cm}^2$

家具数 : 17

表 4.11 問題 C の家具

家具番号	名前
1	ソファ
2	センターテーブル
3	テレビ
4	ベッド (S)
5	ナイトテーブル
6	チェスト
7	電話
8	パソコン
9	ナイトテーブル
10	デスク
11	ワークチェア
12	本棚
13	ベッド (SD)
14	ナイトテーブル
15	テーブルランプ
16	ワードローブ
17	チェスト

D) 4 部屋問題

部屋サイズ :  $340 * 340 \text{ cm}^2$ 、 $240 * 340 \text{ cm}^2$ 、 $240 * 340 \text{ cm}^2$ 、 $240 * 340 \text{ cm}^2$

家具数 : 25 (C 問題全ての家具+8 個カラーボックス)

E) 4 部屋問題

部屋サイズ :  $340 * 340 \text{ cm}^2$ 、 $240 * 340 \text{ cm}^2$ 、 $240 * 340 \text{ cm}^2$ 、 $240 * 340 \text{ cm}^2$

家具数 : 30 (D 問題全ての家具+5 個カラーボックス)

### 4.2.3 適応度の評価

提案手法で用いた LF の効果を評価するため、ES、GA、GA+LF の比較実験を行った。図 4.12 は、問題 E に対する各手法の進化過程の一例である。縦軸は集団中の最良個体の違反点数を表している。この図から、最も規模の大きい問題 E に対して、3 手法とも評価回数 60000 回以下で収束していることがわかる。

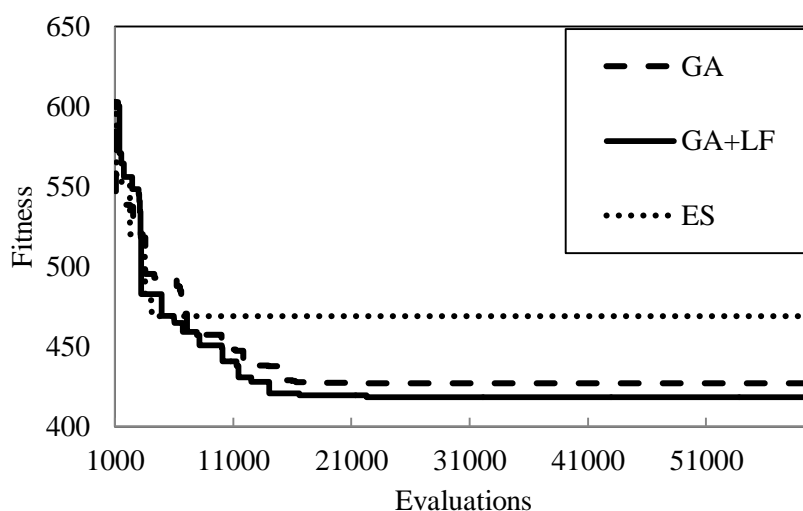


図 4.12 問題 E に対する各手法の進化過程

次に、問題 A～E に対する実験結果を表 4.12 に示す。表中の下線で示した値は、各問題における最良値である。この表から次のことがわかる。

- ES と GA を比較すると、全ての問題に対して GA の方が優れている。
- GA と GA+LF を比較すると、全ての問題に対して違反点数の平均値は GA+LF の方が小さい。また、違反点数の最小値も小さいまたは等しい。

以上のことから、本問題に対する探索手法としては、ES よりも GA が適しているといえる。また、GA に LF を導入することにより、適応度の向上が期待できるといえる。

表 4.12 各手法の違反点数の比較（括弧内は標準偏差）

問題	ES		GA		GA+LF	
	min	mean	min	mean	min	mean
A	5	9(4)	2	7(4)	<u>1</u>	<u>5(3)</u>
B	23	47(11)	14	14(1)	<u>11</u>	<u>13(1)</u>
C	182	214(13)	169	189(11)	<u>158</u>	<u>174(9)</u>
D	387	406(18)	<u>334</u>	352(5)	<u>334</u>	<u>340(7)</u>
E	459	487(10)	415	426(10)	<u>407</u>	<u>420(9)</u>

表 4.13 は通常の GA も 5 つの島を分けて提案手法との比較結果を示す。結果から見ると、各島で提案手法は GA より同等また良い結果を得られた。

表 4.13 通常の GA と提案手法の各島の結果の比較

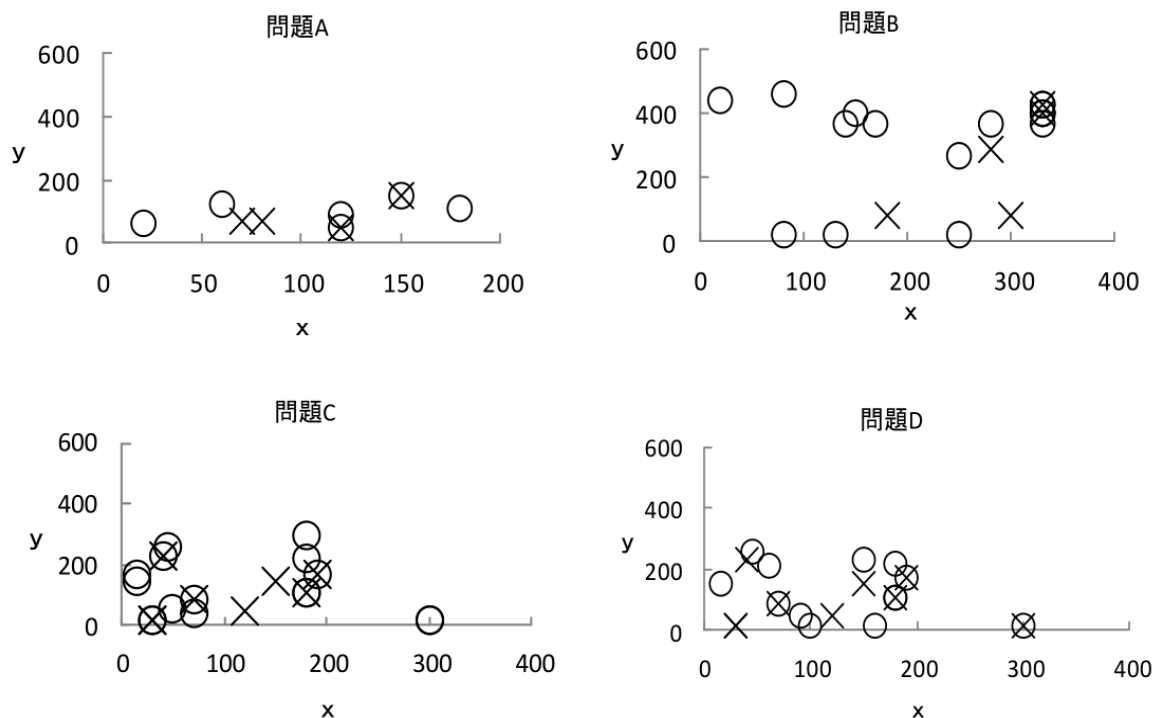
問題	手法	島									
		1		2		3		4		5	
		min	mean	min	mean	min	mean	min	mean	min	mean
A	GA	3	9(6)	2	8(5)	4	7(3)	2	9(4)	4	<u>6(2)</u>
	提案手法	3	7(4)	3	6(3)	1	8(2)	1	7(4)	<u>1</u>	<u>6(3)</u>
B	GA	13	22(10)	12	21(7)	13	22(8)	13	22(8)	12	22(10)
	提案手法	12	20(7)	12	20(8)	12	19(7)	<u>12</u>	<u>18(8)</u>	12	21(8)
C	GA	162	204(17)	177	207(15)	182	208(13)	187	208(12)	180	206(13)
	提案手法	177	190(9)	<u>160</u>	191(14)	170	<u>188(9)</u>	167	192(14)	168	193(13)
D	GA	353	366(9)	355	369(12)	354	369(10)	352	370(8)	356	371(11)
	提案手法	362	374(6)	<u>335</u>	<u>360(10)</u>	354	370(9)	352	367(10)	362	378(12)
E	GA	421	460(18)	437	460(10)	434	458(13)	427	461(16)	425	457(14)
	提案手法	<u>420</u>	<u>454 (11)</u>	432	456(12)	429	459(14)	425	457(14)	429	459(18)



#### 4.2.4 多様性の評価

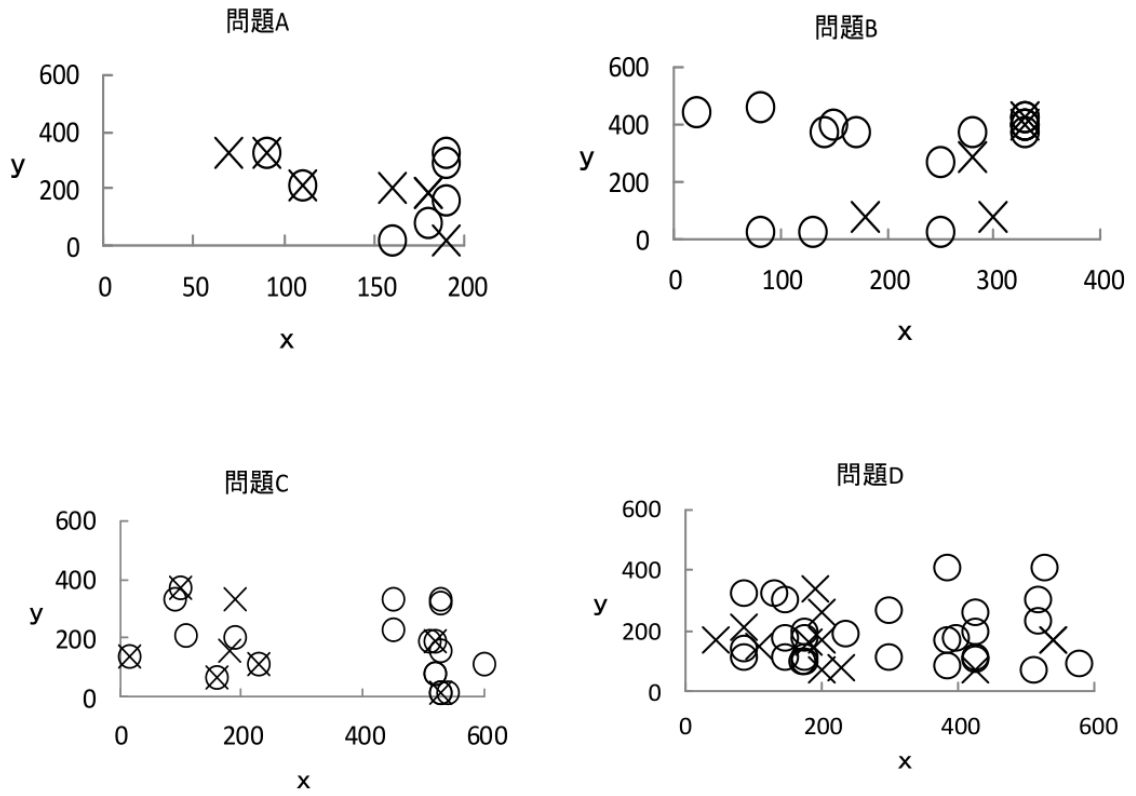
次に提案手法の多様性を評価するため、提案手法と GA+LF によって生成された個体（レイアウト）における家具の位置座標の分布を比較した。図 4.13 と図 4.14 は、問題 A~D の部屋 1 または部屋 2 の左上端を原点としたときのテレビとパソコンの位置座標をプロットしたものである。提案手法について、各島の上位 20 個体（合計 100 個体）、GA+LF については、集団中の上位 100 個体を選択した。複数の家具の座標が完全に一致している場合は、一つの家具としてプロットした。図 4.13 と 4.14 中のすべてのグラフにおいて、提案手法の方がプロットの数が多く、分布の範囲も広いことがわかる。これより、提案手法は GA+LF よりも多様性の維持に優れているといえる。

図 4.15 は、問題 B に提案手法を適用して得られた各島の最良解のレイアウトの例を示している。この図から、各島で異なるレイアウトが得られたことがわかる。



○提案手法 ×GA+LF

図 4.13 各問題におけるテレビの座標分布



○提案手法 ×GA+LF

図 4.14 各問題におけるパソコンの座標分布

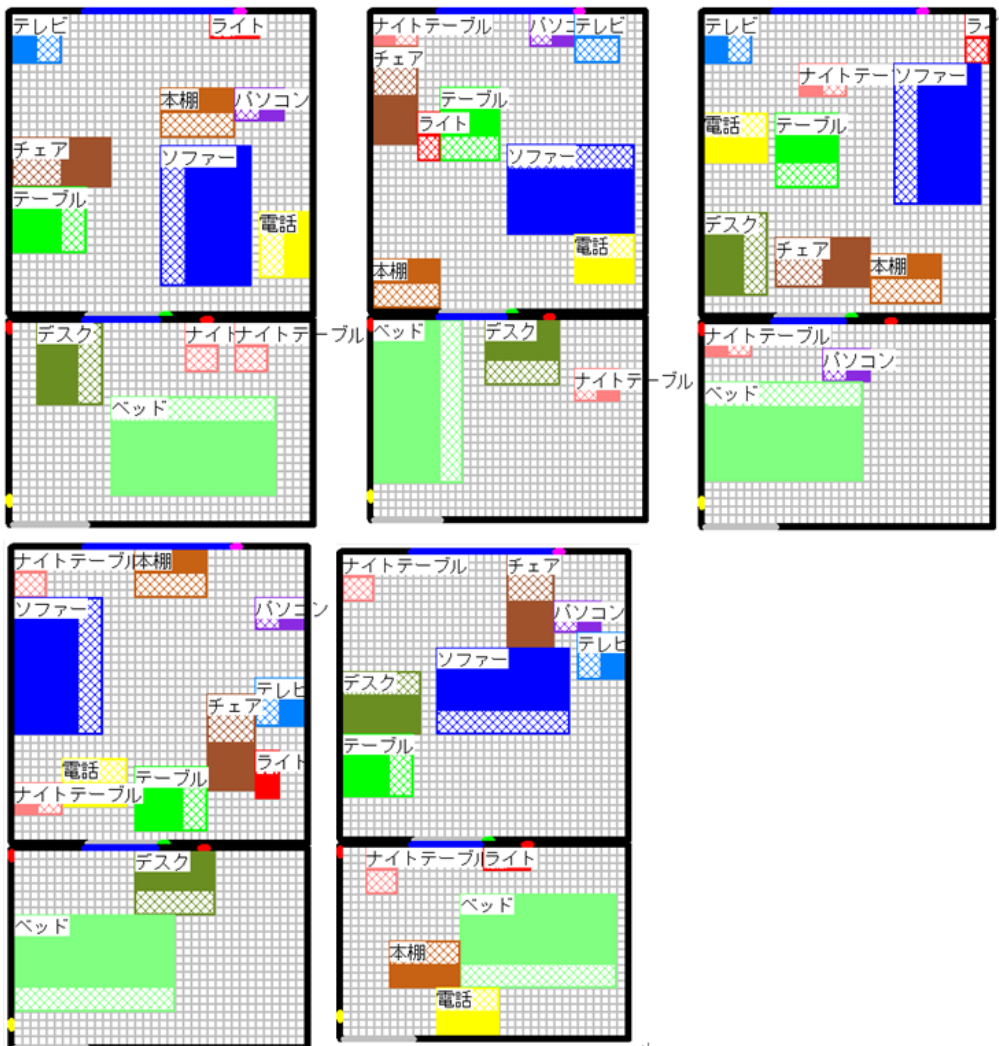


図 4.15 問題 B のレイアウト例

### 4.3 まとめ

この部分は室内レイアウト問題に対する新たな解法を提案した。家具数が 8~30 で部屋数が 1~4 の問題を対象とした実験から、本手法は通常の進化計算より適応度と多様性の両面で優れていることを確認した。

本論文では、多様性の評価として家具の座標分布を利用したが、これだけでは十分とはいえない。今後は、多様性の定量的な評価方法を開発することが必要と考える。また、提案手法は、室内レイアウト問題を対象としているが、今後、別の探索問題への適用検討を通して、本手法をより一般化したいと考えている。

## 第五章 結論

本研究は、Levy Flight を用いた EA による施設レイアウト問題の解法を提案した。また、室内レイアウト問題を対象として、島毎に異なるデコード化方法を用いた GA による対話型システムを提案した。多種多様な制約を含む室内レイアウト問題の実用的な解を出力することができた。

本章では、本研究から得られた結論を示し、最後に今後の課題について述べる。

### 5.1 研究の結論

#### 5.1.1 第三章「ベンチマーク問題への適用」の結論

職場数が 62 までのベンチマーク問題を用いた評価実験を実施し以下の結論を得た。

- Levy 分布に基づく探索は、一様分布や正規分布に基づく探索よりも大規模な問題に対して性能が高い。
- 提案手法は GA や ACO などの従来手法より大規模な問題に適している可能性が高い。

#### 5.1.2 第四章「室内レイアウト問題への適用」の結論

(1) 室内レイアウト問題に対する複数のデコード化と Levy Flight に基づく方法を提案した。家具の数が 8~30、部屋サイズ 1~4 の問題に対して、提案手法は通常の進化計算より適応度が優れていることを確認した。また、多様なレイアウトを生成する可能であることを確認した。

- (a) 室内レイアウト問題には多種多様な制約がある。

- (b) 個人の趣味、希望、ユーザの主観に関わる制約がある。
- (c) ドア、窓、コンセントの位置などユーザの利用しやすさに関わる制約がある。

(2) 以上の問題点を解決するために次のような方針を採用した。

- (a) 室内レイアウト問題を制約充足問題としてとらえ、定式化した。
- (b) 多種多様な制約を整理・分類して表現し、制約の重要度を違反点数で表した。
- (c) 染色体の表現方法を検討し、より形質が遺伝しやすい交叉方法を考察した。
- (d) 染色体に家具の向きの情報を持たせることで家具の向きまで考慮した配置が探索をできるようにした。
- (e) 島毎に異なるデコード化方法を用いて多様性の維持を図る。

## 5.2 本研究の工学的意義

(1) 組み合わせ最適問題への適用：

- ① Levy Flight に基づく進化計算手法が施設レイアウト問題に対して有効であることを示した。
- ② 組合せ問題の解探索において、Levy 分布は正規分布や一様分布よりも有効である分野が存在することを確認した。

(2) 実レイアウト設計問題への適用：

- ① ユーザの嗜好を反映しつつ、多様なレイアウトを生成する一手法を示した。
- ② 実用的なレイアウトシステムを開発した。

### 5.3 今後の課題

今後の課題としては、以下が重要であると考えます：

- 提案手法の有効な応用分野を明確にすること。
- 他の分布関数や一般の Levy 分布について比較検討すること。
- 多様性の定量的な評価方法を開発すること。
- アンケートによる主観評価実験を実施すること、ならびに、システムを拡張して、任意の部屋に適用できるようにすることが挙げられる。

## 謝辞

本研究を行うにあたり、多くの方々にご指導、ご協力をいただきました。

指導教官である本学システム情報系情報工学域狩野均教授には、研究の方針，論文の執筆など有益なご教授を賜るとともに、留学生活方面もいろいろご指導をいただきました。狩野均教授に深く感謝いたします。本論文をまとめるにあたり、本学システム情報系安永守利教授、山本幹雄教授、古川宏准教授、アランニャ・クラウス助教には貴重なご意見を頂きました。心からお礼申し上げます。

また、本研究を進めるにあたっては、本学知識システム研究室の皆様にも、多くのご協力と助言をいただきました。本研究室のOB 落合さん、杉山さんは技術方面には数々の御協力を頂きまして心から御礼申し上げます。最後に、支えてくださった家族、友人に感謝します。

## 参考文献

- [Ali 15] Ahmed Fouad Ali: A Hybrid Gravitational Search with Levy Flight for Global Numerical Optimization, *Information Sciences Letters Inf. Sci. Lett.* 4, No. 2, pp.71-83 (2015)
- [Armour 63] Armour G.C., Buffa E.S., A Heuristic Algorithm and Simulation Approach to Relative Location of Facilities, *Management Science* vol.12, pp. 450-468 (1963)
- [Asl 16] Ali Derakhshan Asl : Unequal-area stochastic facility layout problems: solutions using improved covariance matrix adaptation evolution strategy, particle swarm optimisation, and genetic algorithm, *International Journal of Production Research* Volume 54, Issue 3, pp 799-823 (2016)
- [Bartumeus 05] Bartumeus, F.: Animal search strategies: a quantitative random-walk analysis. *Ecology* 86, pp.3078-3087(2005).
- [Beyer 02] Hans georg Beyer: Evolution strategies A comprehensive introduction, *Natural Computing* 1, pp 3-52 (2002).
- [Buldyrev 93]S.V. Buldyrev, Generalized Levy-walk model for DNA nucleotide sequences. *Phys.Rev.E.* 47 4514(1993)
- [Dorigo 91] Dorigo, M., Maniezzo, V. and Colorni, A.: The Ant System: An autocatalytic optimizing process, Technical report 91-016, Dipartimento di Elettronica e Informazione, Politecnico di Milano (1991).
- [Drira 07] Amine Drira, Henri Pierreval, Sonia Hajri Gabouj: Facility layout problems: A survey, *Annual Reviews in Control*, Vol.31, 255–267(2007)
- [遠藤 71]遠藤,吉田,武岡,設備配置と流れ作業, 日刊工業新聞社 (1971)
- [Francis 74] Francis, R. L. and White, J. A. Facility layout and location an analytical approach, Prentice Hall (1974).
- [Gonçalvesa 15] José Fernando Gonçalvesa, Mauricio G.C.Resende:A biased random-key genetical gorithm for the unequal area facility layout problem, *European Journal of Operational Research*,pp 86-107(2015)
- [Guan 16] Guan, J. and Lin. G. Hybridizing variable neighborhood search with ant colony optimization for solving the single row facility layout problem. *European Journal of Operational Research* 248, pp. 899-909(2016).
- [Hassan 87] Hassan, M. and Hogg, G: A review of graph theory applications to the facilities layout problem. *Omega*, Volume 15, Issue 4, pp. 291-300(1987).
- [Hakli 14] Hakli, H. and Uguz, H. A novel particle swarm optimization algorithm with Levy flight, *Applied Soft Computing*, Vol. 23, pp. 333-345(2014).



- [Hernández 15]Hernández, L. G. and Romero, J. M. P. et al. A novel hybrid evolutionary approach for capturing decision maker knowledge into the unequal area facility layout problem, *European Journal of Operational Research*, pp. 4697-4708(2015).
- [Kamaruzaman 13] Kamaruzaman, G. M. and Zain, A. M. et al.. Levy Flight Algorithm for Optimization Problems – A Literature Review, *Applied Mechanics and Materials*, Vol. 421, pp. 496-501(2013).
- [Hothorn 02]Hothorn T and Hornik K, Exact nonparametric inference in R, *Compstat*, pp.355-360, (2002).
- [Kazi 11] Kazi Shah Nawaz Ripon: Using Pareto-Optimality for Solving Multi-Objective Unequal Area Facility Layout Problem, *GECCO'11*, July 12–16, pp.681-688(2011).
- [Kazi 13]Kazi Shah Nawaz Ripon: Adaptive variable neighborhood search for solving multi-objective facility layout problems with unequal area facilities, *Swarm and Evolutionary Computation*, pp.1-12(2013).
- [Kim 00]Kim, H.-S., Cho, S.-B. Application of interactive genetic algorithm to fashion design, *Engineering Applications of Artificial Intelligence* (13) pp.635-644(2000).
- [北野 93]北野,遺伝的アルゴリズム, 産業図書(1993).
- [是永 00] 是永基樹, 萩原将文. 対話型進化計算法によるインテリアレイアウト支援システム,情報処理学会論誌,vol.41,No.1, pp. 3152-3160(2000).
- [Komarudin 10] Komarudin, Kuan Yew Wong: Applying Ant System for solving Unequal Area Facility Layout Problems, *European Journal of Operational Research* 202 730–746(2010).
- [Konaka 11] Kulturel-Konaka, S. and Konak, A. A new relaxed flexible bay structure representation and particle swarm optimization for the unequal area facility layout problem, *Engineering Optimization*, Vol.43, No. 12, pp. 1263–1287(2011).
- [Kusiak 87]Kusiak, A. and Heragu, S. S.:The facility layout problem. *European Journal of Operational Research* 29, pp. 229-251(1987).
- [Lee 04] Evolutionary Programming Using Mutations Based on the Levy Probability Distribution, *IEEE Transactions on Evolutionary Computation*, 8-1, pp.1--13 (2004).
- [Liu 07] Qi Liu, Russell D.Meller: A sequence-pair representation and MIP-model-based heuristic for the facility layout problem with rectangular departments, *IIE Transactions*,Vol 39, 377–394(2007)
- [Meller 99]Meller, R.D., Narayanan, V. and Vance, P.H., Optimal facility layout design. *Operations Research Letters*, 23, pp.117–127(1999).
- [Montreuil 90] Montreuil, B.: A modeling framework for integrating layout design and flow network design, *Proceedings of the material handling research colloquium*, pp. 43-58(1990).

- [長尾 00]長尾,最適化アルゴリズム, 昭晃堂(2000).
- [並木 08]並木,線形計画法 (応用最適化シリーズ),朝倉書店(2008).
- [大森 11]大森,三好,吉本,実数値 Genetic Algorithm を用いた施設レイアウト問題の解法, J Jpn Ind Manage Assoc 62, 182-189 (2011)
- [大森 13] 大森,吉本:力学的モデルを用いた施設レイアウト問題の解法,日本経営工学会論文誌 64, pp.145-156 (2013).
- [栞川 13] 栞川寿美礼,住まいのインテリア設計, pp. 40-43(2013).
- [佐藤 96] 佐藤浩, 小野功, 小林重信. 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,人工知能学誌, Vol .12,No.5, pp.734-748(1996).
- [Scholz 09]Scholz, D. and Petrick, A. et al.. Slicing tree and tabu search based heuristic for the unequal area facility layout problem. European Journal of Operational Research 197 (1), pp.166–178(2009).
- [Singh 06]Singh, S. P. and Sharma, R. R. K., “A review of different approaches to the facility layout problems”, International Journal of Advanced Manufacturing Technology 30, pp. 425-433(2006).
- [Sherali 03] Sherali, H.D., Fraticelli, B.M.P. and Meller, R.D. Enhanced model formulations for optimal facility layout. Operations Research, 51,629–644(2003)
- [清水 01]清水, 遺伝的アルゴリズムを用いた室内レイアウトシステム, 情報処理学会第 62 回 全国大会 8K-02 , 2001.
- [Tam 92] Tam,K,Y:A Simulated Annealing Algorithm for Allocation Space To Manufacturing Cells,Int.J.Prod,Res.,Vol.30,pp.63-87(1992).
- [Yang 09]Yang, X. S. and Deb, S.. Cuckoo Search via L´evy Flights, World Congress on Nature & Biologically Inspired Computing, pp. 210-214(2009).
- [四辻 10] 四辻哲章, 計算機シミュレーションのための確率分布乱数生成法, プレアデス出版, pp. 175-180, 2010.
- [Viswanathan 08] Viswanathan: Levy flights and super diffusion in the context of biological encounters and random searches, Physics of Life Reviews 5 133–150 (2008) .
- [Viswanathan 11] G. M. Viswanathan, M. G. E. da Luz, H. E. Stanley, The Physics of Foraging, Cambridge University Press (2011)
- [脇下 96]脇下, 西森, 狩野, 西原: 制約に基づく対話型室内レイアウトシステム, 情報処理学会第 52 回 全国大会 6C-5(1996).