

遺伝的ネットワークプログラミングによる
エレベータ群管理システムの最適化に関する研究

Study on Optimization of
Elevator Group Supervisory Control System
using Genetic Network Programming

2006年2月

早稲田大学大学院 情報生産システム研究科
情報生産システム工学専攻 進化型計算システム研究

江口 徹

目 次

第 1 章	序論	1
1.1	研究の背景	1
1.2	従来研究と本研究の位置付け	2
1.2.1	エレベータ群管理研究の歴史的変遷	3
1.2.2	AI 技術の応用による従来研究の事例	3
1.2.3	本研究の目的および位置付け	5
1.3	本研究の内容	6
1.4	本論文の構成	8
第 2 章	GNP によるエレベータ群管理システムの基礎検討	9
2.1	序	9
2.2	GNP によるエレベータ群管理システムの基本モデル	10
2.2.1	基本構成	11
2.2.2	ノード関数	11
2.2.3	適合度関数	12
2.3	シミュレーション	14
2.3.1	実行条件	14
2.3.2	進化の様子	14
2.3.3	従来手法との性能比較	17
2.3.4	ノード使用頻度の検討	21
2.4	まとめ	25

第 3 章	ランキング処理とノード関数最適化を考慮した GNP による群管理システム	27
3.1	序	27
3.2	導入アルゴリズム	28
3.2.1	ランキング処理	28
3.2.2	ノード関数最適化	28
3.3	ランキング処理とノード関数最適化による拡張モデル	29
3.3.1	基本構成	30
3.3.2	ノード関数	32
3.3.3	適合度関数	33
3.4	進化による最適化のシミュレーション	33
3.4.1	実行条件	33
3.4.2	進化の様子	35
3.4.3	ノード関数最適化の効果の検討	35
3.5	テストシミュレーションによる性能評価	35
3.5.1	実行条件	37
3.5.2	従来手法との性能比較	37
3.5.3	獲得した群管理ルールの検討	41
3.6	まとめ	45
第 4 章	重要度指標付き GNP による切替え制御型群管理システム	47
4.1	序	47
4.2	基本構成	48
4.3	割当て GNP	48
4.4	切替え GNP	48
4.4.1	重要度指標による切替えアルゴリズム	49
4.4.2	ノード関数	49
4.4.3	適合度関数	51

4.5	切替え GNP の進化のシミュレーション	51
4.5.1	実行条件	51
4.5.2	進化の様子	54
4.6	テストシミュレーションによる性能評価	54
4.6.1	実行条件	54
4.6.2	従来手法との性能比較	56
4.6.3	切替えの様子	56
4.7	まとめ	59
第 5 章	GNP によるダブルデッキエレベータ群管理システムの基礎検討	61
5.1	序	61
5.2	GNP によるダブルデッキエレベータ群管理システム	62
5.2.1	基本構成	62
5.2.2	評価項目	62
5.2.3	呼び割当てのメカニズム	66
5.2.4	ノード関数	69
5.2.5	適合度関数	70
5.3	シミュレーション	71
5.3.1	実行条件	71
5.3.2	進化の様子	73
5.3.3	従来手法との性能比較	73
5.3.4	ノード使用頻度の検討	77
5.3.5	従来システムに対する有効性の検討	81
5.4	まとめ	85
第 6 章	結論	87
	参考文献	89

付 録 A 遺伝的ネットワークプログラミング (GNP)	97
A.1 基本概念	97
A.2 基本構成	99
A.2.1 構造および実行手順	99
A.2.2 プログラムの表現	99
A.2.3 遅れ時間による実時間処理	101
A.2.4 適合度	101
A.3 進化のアルゴリズム	103
A.4 遺伝的オペレータ	103
A.4.1 選択	103
A.4.2 交叉	104
A.4.3 突然変異	104
付 録 B エレベータ群管理システム (EGSCS)	107
B.1 エレベータ群管理の目的	107
B.2 システムの概要	107
B.2.1 基本構成	108
B.2.2 ビル階床および交通流	108
B.2.3 群管理の性能評価の指標	110
B.3 エレベータシミュレータによる評価	110
付 録 C 実数値 GA によるノードパラメータの最適化	113
付 録 D 時間帯における GNP 選択の重要性	115
付 録 E 重要度指標付き GNP による機能切替え	117
E.1 基本概念	117
E.2 重要度指標による切替えアルゴリズム	118

付 録 F 行先階登録方式エレベータシステム (DFGES)	121
F.1 システムの概要	121
F.2 行先階登録方式エレベータの群管理方式	122
付 録 G ダブルデッキエレベータシステム (DDES)	125
G.1 システムの概要	125
G.2 運行モード	126
G.3 ダブルデッキエレベータ特有の挙動	126
G.4 ダブルデッキエレベータの群管理方式	128
G.5 行先階登録方式の導入の可能性	128
謝辞	133
研究業績	135

第1章

序論

1.1 研究の背景

飛躍的な発展を遂げた科学技術によって、近年様々なシステムの効率化や最適化が実現されてきた [1]。その背景には、システムに内在する複雑な問題を工学的、数理的アプローチを用いて効率的に解決する手法が開発されたことが挙げられる。大規模かつ複雑なシステムの最適制御は、その探索空間の膨大さや強い非線形性により解決が困難とされる [2]。したがってそれらを柔軟かつ効率的に扱うには既存の知識の適用よりむしろ、経験や創発といったプロセスを経て獲得された知識を用いるほうが好ましい。この知見に基づき、近年人間や生命体の持つ高度な情報処理能力を模したシステム制御のアプローチである、人工知能 (Artificial Intelligence: AI) 研究 [3, 4] が盛んに行われている。

AI 研究のひとつに、生物の進化に関わる知見に着想を得た計算モデルである進化論的計算手法 [4, 5] がある。進化論的計算手法には遺伝的アルゴリズム (GA) [6, 7, 8, 9]、遺伝的プログラミング (GP) [10, 11, 12]、進化的プログラミング (EP) [13]、進化的戦略 (ES) [14] 等が提案されており、それらは選択、淘汰という自然選択のプロセスにより世代を重ね最適解を求める手法である。問題の解を表現する個体は遺伝子情報によって規定され、個体群 (集団) に対し交叉や突然変異といった遺伝的オペレータを施すことで問題に適合する遺伝子を生成する。このような遺伝子集団による解の大域的、発見的探索により、進化論的計算手法は大規模複雑システムに対する有効な手法として期待できる。実際、進化論的計算手法によるシステムの最適制御に関する研究例は数多く報告されており [8, 11]、その有効性が確認されている。

2 第1章 序論

一方，身近な大規模複雑システムの例として，エレベータや鉄道，自動車に代表される交通システムが挙げられる．これらは機械や装置，人間の挙動の相互作用および確率的要素による強い非線形性を持つため，その最適制御は非常に困難となる．また特にエレベータシステムの最適制御に関して，ビル内の複数のエレベータの運行管理を行うエレベータ群管理システム（Elevator Group Supervisory Control System: EGSCS）[15, 16, 17]として，古くからその検討が行われている．近年ではこれに AI 技術を導入する研究が盛んに進められており [18]，その特徴を活かした従来の群管理方式に対する性能の向上を達成し実用化に至っている．さらにここ数年では，エレベータシステムに対するニーズの多様化やビルの高層化による既存システムの輸送能力の限界等の理由から，次世代型エレベータシステム[18, 19]と呼ばれる新しいエレベータの開発が進められている．次世代型エレベータシステムの導入によってその群管理が難易度を増すことは容易に予想され，既存の知識や経験則に基づく群管理方式では一層対応が困難となる．今後，AI 技術は次世代型エレベータの群管理のコア技術となることが期待され，これに既存の枠組みにはない新しい技術を導入することは非常に興味深く，挑戦的なアプローチである．

1.2 従来研究と本研究の位置付け

本研究は上記研究背景の下で，近年提案された新しい進化論的計算手法である遺伝的ネットワークプログラミング（Genetic Network Programming: GNP）[20, 21, 22]を，エレベータ群管理システムへ導入する研究を行う．GNP はグラフ構造のプログラムを進化のプロセスにより最適化する手法で，GP や EP 等との比較検討によりその有効性が確認されている [20, 21, 23]．これまでに GNP を EGSCS へ導入した例はなく，本研究は GNP の特徴を活かした，有効かつ新規性の高い群管理方式や知見の確立を期待できる．さらには，それらを活用した GNP による次世代型エレベータ群管理システムの検討によって，GNP をその代表的群管理技術として確立することも期待できる．以下，従来のエレベータ群管理研究の流れとそれに対する本研究の位置付けについて述べ，本研究の目的を明確にする．

1.2.1 エレベータ群管理研究の歴史的変遷

エレベータ群管理研究の流れは大別してマイクロコンピュータの登場以前と以後に分けられる。マイクロコンピュータの登場以前は、エレベータの停止階や割当ての管理はリレー回路やIC回路により計算していた。群管理制御技術としては基準階での出発管理や運行距離の制御、停止階のゾーン分割を行いゾーン毎に割当てを行う方式が採用された。その後の半導体技術の発展によりマイクロコンピュータが登場すると、エレベータ群管理研究は飛躍的に発展を遂げ、また同時期に現在の群管理の根幹を成す、ホール呼び単位でエレベータを割当てる方式が確立された。呼びの割当ては基本的に呼び発生と同時に決定される（即時割当て方式）が、当初は割当て決定の際の演算処理時間が問題となったため、単純な制御理論や数理モデル、経験則に基づく割当て制御で実現されていた[24]。近年のコンピュータの性能向上によりその演算時間が問題とならなくなるに及び、群管理システムにおいて複雑な演算処理を実行することが可能となり、群管理の性能はさらに向上を遂げた[25, 26]。そしてなお一層の群管理性能の向上を図る試みとして様々なAI技術が導入され、現在ではエレベータ群管理研究の主流となっている。

1.2.2 AI技術の応用による従来研究の事例

現在、EGSCSに導入されている主たるAI技術として、GA、ニューラルネットワーク（NNs）[27]、ファジィ理論（FL）[28]が挙げられる。これらはそれぞれの持つ特徴を活かした呼びの割当ておよび交通流の検出を行う機能を構成しており、1990年代初頭より研究開発が進められ、現在稼働中の多くのエレベータシステムへ導入されている。以下、各方式の特徴およびそのアルゴリズムについて簡潔に述べる。

【GAによる方式 [29, 30, 31, 32, 33]】

GA、NNsを始め多くの呼びの割当て方式では、エレベータ評価関数を使用して割当てエレベータを決定する方式がとられる。エレベータ評価関数 $e(i)$ ($i \in I$) は呼び発生時の各エレベータの呼びに対するサービス能力を待ち時間等の評価指標を用いて数値化したものであり、一般にEq. (1.1)によって表される。

4 第1章 序論

$$e(i) = \sum_{X \in \Gamma} w_X \cdot X(i) \quad (1.1)$$

ここで，

- I : エレベータ ID の集合
- Γ : エレベータ評価指標の集合
- w_X : 評価指標 X に関する重み付けパラメータ
- $X(i)$: エレベータ i の評価指標 X に関する評価値

であり，各評価関数値の中で最良値を持つエレベータへ割当てが行われる．この方式では定義した指標およびパラメータ設定に基づく最適なエレベータが常に選択され，群管理を実行する．また指標の追加によってより効率的な群管理が実現可能である．しかしそれに伴う重み付けパラメータの増加によって，手動によるパラメータ調整は困難となる．

GA による方式では，この重みパラメータベクトル w ($w_X \in w$) を GA の遺伝子として規定し，交叉や突然変異の遺伝的オペレータにより最適化する．GA の適用によって指標数の増加が可能となり，文献 [29, 30] では階床単位にまで指標を設定することにより，きめ細かな群管理が可能となっている．次世代型エレベータシステムへの GA の導入も行われており，文献 [32] ではマルチカーエレベータシステム (MCES) [34, 35] の群管理システムの最適化を GA により行い，パラメータの手動設定による方式よりも効率的な群管理を達成している．また上記評価式によらない GA による割当て方式として，呼び割当て後のエレベータ変更が可能な割当て変更方式において割当ての組合わせを遺伝子とし，最適化する研究例も報告されている [33] ．

【NNs による方式 [36, 37, 38, 39]】

NNs による方式も同様に評価式 $e(i)$ の最適化を NNs の学習機能により行う．具体的には入力層においてエレベータ位置情報，方向，呼び発生状況等のデータを入力し，フィードフォワード演算により各エレベータの評価値を出力し，最適なエレベータへ呼びを割当てる [36] ．したがって評価式は厳密には Eq. (1.1) の形はとらないが，NNs を各指標を考慮したエレベータ評価式として見ることができる．この他にも，NNs のパターン識別

能力に着目し交通流検出に利用した研究例 [37] や、呼び発生時に求める到着予想時間の計算への適用例 [38] 等の多岐にわたる応用例が報告されている。

【FL による方式 [40, 41, 42]】

上述のエレベータ評価関数による割当て方式はその簡潔性、拡張性の高さから GA、NNs 等への導入を始め広く用いられてきた。しかし評価関数によって呼び割当ての決定論理が固定されるため、多様な状況に追従する柔軟な群管理の実現は難しい。FL による方式では上記問題点の解決のため、評価関数ではなく IF-THEN ルールベースによる群管理制御を行う。IF-THEN ルールベースによる群管理では、呼び発生時の状況に応じたルールをルールベースに記述することによりケース・バイ・ケースの柔軟な呼びの割当てを実現する。この方式では、呼び発生時の状況に最適なルールを選択するメカニズムに FL を導入し、その状況と各ルール条件との適合度をファジィ論理演算により計算しルールを選択する [40]。また文献 [41] では EGSCS の交通流検出機能に FL を導入し、FL による割当て機能と併用する群管理システムが提案されており、交通流の変化を検知し適応的に割当て機能を変更する群管理を実現している。

また上記方式の他にも NNs による割当て方式の学習に強化学習 (RL) [65] を導入する方式 [43]、エージェントによる自律分散制御技術の枠組みを導入する方式 [44]、リアルタイムシミュレーションによる待ち時間等の指標の予測を行う方式 [45, 46]、進化的戦略 (ES) を導入する方式 [47] 等の各種 AI 技術による研究例が報告されている。

1.2.3 本研究の目的および位置付け

本研究で提案する GNP による EGSCS は、FL による方式と同じく IF-THEN ルールベースによる群管理方式である。ただし FL では設計者が全ルールベースを予め作成する必要があるが、また設計者の経験に基づくルールによる群管理しか実現し得ない。これに対し提案方式では、GNP のグラフ構造プログラムから生成されるルールを進化により最適化するため、設計者に依存する部分は GNP の構造の初期設定のみであり、またルールの自動生成により設計者の経験の範疇を超えるルールの発見を期待できる。さらに提案方式

は群管理ルール生成のプロセスに対し，GA，強化学習等の他の AI 技術を導入した拡張や，交通流検出機能を付加した切替え制御型群管理システムの構築も可能である．

このように高い適応性と拡張性を備えた AI 技術ベースの群管理方式は他に例がなく，本研究は最新の AI 技術を駆使したエレベータ群管理方式の最先端研究と位置付けられる．また本研究は次世代型エレベータシステムであるダブルデッキエレベータシステム (DDES) [19, 48, 49, 50] への GNP の導入についても検討しており，現在開発途上のダブルデッキエレベータ群管理システムの GNP による構築という点から見ても非常に興味深い研究といえる．

1.3 本研究の内容

本研究では GNP による EGSCS の構築に関して，その導入可能性と基本特性の検討，ランキング処理とノード関数最適化の有効なアルゴリズムの導入による拡張，交通流検出・切替え機能の付加，DDES への導入に至る各群管理モデルを提案し，計算機シミュレーションによってその性能評価を行っている．以下に本研究の概略を示す．

本研究は GNP を EGSCS へ導入する初めての例であり，その導入可能性および基本特性については未知数である．第1段階として，EGSCS への導入に対する GNP の利点を明確にし，GNP の基本的枠組みを導入した EGSCS の基本モデルを提案しその基礎検討を行う [51, 52, 53]．基本モデルは GNP の基本的枠組みによるモデルであるが，EGSCS への導入に際しいくつかの新規枠組みの導入も行っている．GNP による群管理（呼びの割当て）は GNP のグラフ構造プログラムを実行することで行われ，群管理に関する各種の先験知識に基づくノード関数上の遷移により割当てルールが生成される．シミュレーションでは現在最も一般的な UP/DOWN ボタン方式のエレベータシステム [15] に対して進化による最適化を行い，単純な割当てルールの従来手法と比較しその性能評価を行った．その結果，提案モデルは従来手法に対して性能が改善し，また各種の先験知識を考慮した効果的な群管理ルールを生成することを明らかにした．しかし同時に，いくつかの問題点も浮彫りとなった．

次に、基本モデルの問題点解決のためのランキング処理とノード関数最適化を考慮した拡張モデルの提案と検討を行う[54, 55, 56]。基本モデルでは GNP の構造および生成される群管理ルールが一般性に欠けるという課題を残したため、新たにランキング情報を利用した判定ノードの引数処理と実数値 GA[57] によるノードパラメータの最適化の 2 種類のアルゴリズムを導入し、GNP の柔軟な進化と群管理ルールの一般化を図っている。また各種先験知識をエレベータ評価項目として定義し、それらを活用する GNP を構成している。シミュレーションでは複数の交通条件に対する GNP の進化を実行し、動的に変化する交通データに対してそれらを使い分けることで柔軟かつ効率的な群管理制御が行われることを示した。また GNP が獲得した群管理ルールの詳細な検討を行い、一般性の高いルール生成が行われることを確認した。

上記拡張モデルの検討では、複数の交通条件に対して最適化した GNP の使い分けにより効率的な群管理制御を実現したが、これは EGSCS の交通流検出機能によりシステム自身が実行できることが望ましい。次段階として、交通流の特定と最適な割当て機能への切替えを司るシステムを GNP により構成し、拡張モデルの割当て機能と連動させた切替え制御型群管理システムの構築を試みる[58, 59]。切替えのメカニズムは、脳の機能局在モデルを規範とする、重要度指標付き GNP[60] により構成する。シミュレーションによる検討の結果、提案する群管理システムは交通状況の変化に追従した適切な割当て機能の切替えを行い、効率的な群管理を行うことを明らかにした。

上記検討により GNP による EGSCS の基本特性および有効性を明らかにし、ひとつの群管理の枠組みとして確立した。本研究の最後に、その枠組みや得られた知見を活かした、ダブルデッキエレベータシステム (DDES) への導入とその群管理モデルの基礎検討を行う。DDES には特有の挙動が存在するため、それらを効率的に扱う群管理が必要となる。したがって提案モデルはこれまでの先験知識に加え、DDES 特有の挙動を考慮した合計 12 種類の評価項目を活用し GNP を構成している。またルール生成のプロセスに GA による評価式の最適化のアプローチを導入し、より効率的な群管理ルールの生成を可能にしている。シミュレーションでは GNP と共通の評価項目を考慮する従来手法との性能比較を行い、提案モデルの有効性を確認した。また従来型エレベータシステムに対する提案モデルのコスト削減効果についても明らかにした。

1.4 本論文の構成

本論文の構成は以下のとおりである．

第2章のGNPによるエレベータ群管理システムの基礎検討では，GNPによるEGSCSの基本モデルの提案とその基礎検討を行う．シミュレーションでは進化による最適化の様子，従来手法との性能比較およびGNPのノード使用頻度の検討を行っている．

第3章のランキング処理とノード関数最適化を考慮したGNPによる群管理システムでは，基本モデルの問題点を解決する2種類のアルゴリズムを導入した拡張モデルの提案とその検討を行う．シミュレーションではビルの1日の交通データを用いた性能評価を行い，またGNPが獲得した群管理ルールの検討も行っている．

第4章の重要度指標付きGNPによる切替え制御型群管理システムでは，重要度指標付きGNPによる割当て機能の切替えシステムを提案し，第3章と同じ交通データを用いたシミュレーションを実行しその性能評価を行う．

第5章のGNPによるダブルデッキエレベータ群管理システムの基礎検討では，第4章までの検討結果をふまえ，GNPをDDESへ導入した群管理モデルを提案しその基礎検討を行う．シミュレーションでは5種類の従来手法との性能比較，ノード使用頻度の検討および従来型エレベータシステムに対するコスト削減効果を検討している．

第6章は，第2章から第5章までの本研究の内容についての結論とする．

第2章

GNPによるエレベータ群管理システムの基礎検討

2.1 序

近年，遺伝子が有向グラフ構造を持つ新しい進化論的計算手法，遺伝的ネットワークプログラミング（Genetic Network Programming: GNP）[20, 21, 22] が提案され，その有効性が明らかにされている．一方，第1章でも述べているように，近年大規模複雑システムの最適制御に対し進化論的計算手法を始めとする AI 技術を応用する試みがなされている．従来の GNP の研究例は基本特性やアルゴリズムの検証を目的とし，仮想世界を扱うベンチマーク問題への適用を主として進められてきた．それゆえ，GNP による大規模複雑システムの最適制御に関する研究はこれまで行なわれていない．本研究では大規模複雑システムのひとつであるエレベータ群管理システム（Elevator Group Supervisory Control System: EGSCS）[15, 16, 17] に対し GNP を導入した，新しい群管理方式の構築を試みる．

本章ではその第1段階として，GNP の基本的な枠組みによる EGSCS の基本モデルを提案し，GNP の導入可能性および基本特性を検討する．提案モデルは GNP の基本的枠組みに基づくが，EGSCS への導入に際し先験知識の判定ノードへの導入，処理ノードの引数処理，適合度によるループ遺伝子の淘汰の新規枠組みを導入している．EGSCS では環境の情報量が膨大になるため，群管理の手法としては有限オートマトン（FSA）等の完全情報を前提とした手法よりはむしろ，環境からの入力情報が部分的な部分観測マルコフ環境下でも適用可能な手法が望ましい．したがって，環境に依存した適切なノード関数の設定により部分的な入力情報のみで適切な処理を実行できる GNP は，EGSCS の

最適化制御に適した手法である．また GNP は IF-THEN ルールベースによる群管理の構成を基本とするため群管理アルゴリズムの解析が容易となり，有益な知見の発見を期待できる．

以下，提案モデルの構成について説明し，シミュレーションでは GNP の進化による最適化，従来手法との性能比較による提案モデルの有効性および GNP のノード使用頻度の検討を行う．

2.2 GNP によるエレベータ群管理システムの基本モデル

本節では提案する GNP による EGSCS の基本モデルの構成について述べる．基本モデルの提案にあたり，以下の3点の新規枠組みを導入している．

【先験知識の判定ノードへの導入】 従来の GNP では判定ノードは環境の単純な観測情報を入力とした．しかし，観測可能な情報が膨大な EGSCS では単純な情報のみでは効率的な群管理ルールを実行できない．そこで先験知識に基づく入力情報を前処理で計算・管理し，それを判定ノードへの入力とする枠組みを導入している．

【処理ノードの引数処理】 EGSCS において，GNP は呼びの割当てエレベータを決定する．そのため処理ノードで特定のエレベータ ID を扱える GNP の構成が必要となるが，特定のエレベータ毎に処理ノードを用意すると GNP のプログラムサイズの増加を招き，またエレベータ数の変更に対応できない等の問題点が生じる．そこで特定のエレベータ ID に依存しない割当て決定が可能で，コンパクトな構成の GNP を実現するため，処理ノードがエレベータ ID を引数とするメカニズムを導入している．

【適合度によるループ遺伝子の淘汰】 これまでは GNP のループ遺伝子は進化の過程で自然に淘汰されるため特別な考慮は必要なかった．しかし提案モデルではループ時の外部処理による群管理の影響によって，適切な GNP 個体の評価が困難となる．したがってループの回数をペナルティとして適合度計算に考慮することによって，ループ遺伝子の効率的な淘汰を可能としている．

以上をふまえ，以下に基本構成，ノード関数および適合度関数について述べる．

2.2.1 基本構成

提案モデルの基本構成を Fig. 2.1 に示す．システムはエレベータシステム，GNP コントローラ，情報管理部から構成される．情報管理部ではエレベータシステムの情報を先験知識を利用しやすい値に変換する．すなわちこの情報管理部は PADO[70] の外部記憶のように処理の過程での計算結果の一時記憶ではなく，現在の環境の情報を入力情報へ変換し記憶する部分である．

システムの開始後，最初の乗客がエレベータホールに出現しホール呼びボタンを押すと GNP が起動しノード遷移を開始する．判定ノードではホール呼び発生時の入力情報を判定し，処理ノードでホール呼びを割当てるエレベータ ID を決定する．処理ノードでの割当て後，次のホール呼びが発生するまでノード遷移はその処理ノードに留まる．ノード遷移の際，累積遅れ時間が予め設定した閾値を超えた場合は GNP の実行を停止し，外部処理として呼び発生階へ最も早く到着可能なエレベータを割当てる．上記の処理を与えられた評価時間が経過するまで実行する．

2.2.2 ノード関数

提案モデルの GNP のノード関数として，以下の 10 種類のノードを用意した．

処理ノード（1 種類）

- エレベータ $d (\in I)$ に発生呼びを割当てる (I : エレベータ ID の集合)

判定ノード（9 種類）

- (1) 発生呼びへ最も早く到着可能なエレベータを判定
- (2) 発生呼びへ 2 番目に早く到着可能なエレベータを判定
- (3) 乗車中の乗客数が 5 人以下で，最も早く到着可能なエレベータを判定
- (4) 乗車中の乗客数が 10 人以下で，最も早く到着可能なエレベータを判定
- (5) 登録ホール呼びの数が 1 個以下で，最も早く到着可能なエレベータを判定
- (6) 登録ホール呼びの数が 3 個以下で，最も早く到着可能なエレベータを判定
- (7) 発生呼び階と既登録かご呼び階が一致し，最も早く到着可能なエレベータを判定
- (8) 発生呼びへ最も早く到着可能な上位 2 台のエレベータの時間差を判定
- (9) 発生呼び階床および呼び方向の判定

12 第2章 GNPによるエレベータ群管理システムの基礎検討

上記判定ノード群において，(1) および (2) の判定結果はエレベータ ID に相当し有向枝の数は $|I|$ 本となる．(3)～(7) までの判定結果は，各エレベータ ID に“該当なし”を加えた $|I| + 1$ 種類となる．以上の判定ノード (1)～(7) から処理ノードへ接続する場合，判定結果のエレベータ ID を引数 d として渡す（“該当なし”の場合は処理ノードに接続できない）．(1)～(7) から (8)，(9) へは接続しない．(8) は呼び発生階へ最も早く到着するエレベータと 2 番目に早く到着するエレベータの到着予想時間の差を比較し，エレベータ間の接近度を判定する．本シミュレーションでは，時間差が 22.5 [s] 未満ならば“接近している”，以上なら“接近していない”と 2 分岐判定する．(9) は発生呼びが { 基準階，一般階 / 上方向，一般階 / 下方向 } のいずれに該当するかの 3 分岐判定を行う．判定ノード (8)，(9) から，処理ノードへは接続しない．処理ノードは判定ノード (1)～(7) で判定したエレベータ ID を引数として受けとり，そのエレベータへ発生呼びを割当てて役割を持ち，1 本の有向枝を有する．また，処理ノードからは判定ノード (8)，(9) へ接続する．引数設定により処理ノードを一般的な表現にし，GNP の構成の簡略化および進化の効率化が期待できる．以上の構成から，“呼び情報やエレベータ情報の判定 → 呼びの割当て”という一連の群管理ルールを進化により獲得できる．以上をふまえた，提案モデルの GNP の構成例を Fig. 2.2 に示す．

2.2.3 適合度関数

EGSCS の群管理目標は乗客がエレベータホールに出現してから行先階へ到着するまでの所要時間，特にエレベータホールでの待ち時間を最小化することである．したがって GNP の適合度関数は，待ち時間の最小化を考慮して定義する．

EGSCS の適合度の一般的な指標に“平均 2 乗待ち時間”[32, 43] がある．平均 2 乗待ち時間の最小化は乗客の平均待ち時間の最小化に寄与する．提案モデルではこれに加えて，2 種類の指標を適合度関数に考慮する．ひとつは長待ちを特に考慮する指標である“2 乗最大待ち時間”である．もうひとつは GNP のループ遺伝子の淘汰を考慮する指標である“ループ回数の 2 乗”である．適合度関数 *Fitness* は以上の重み付け和として Eq. (2.1) のように定義した．

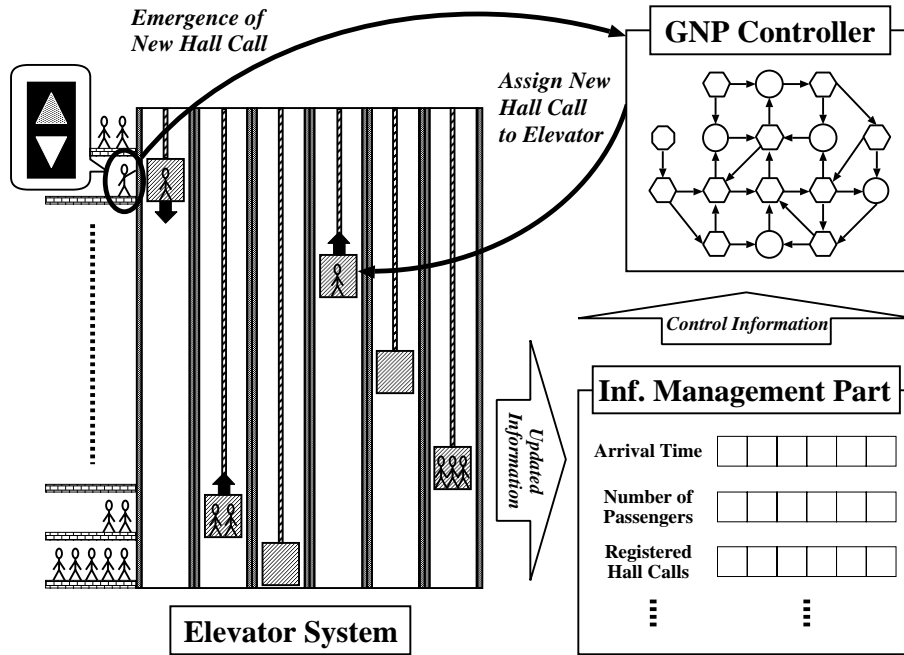


Fig. 2.1 Basic Structure of Proposed Model

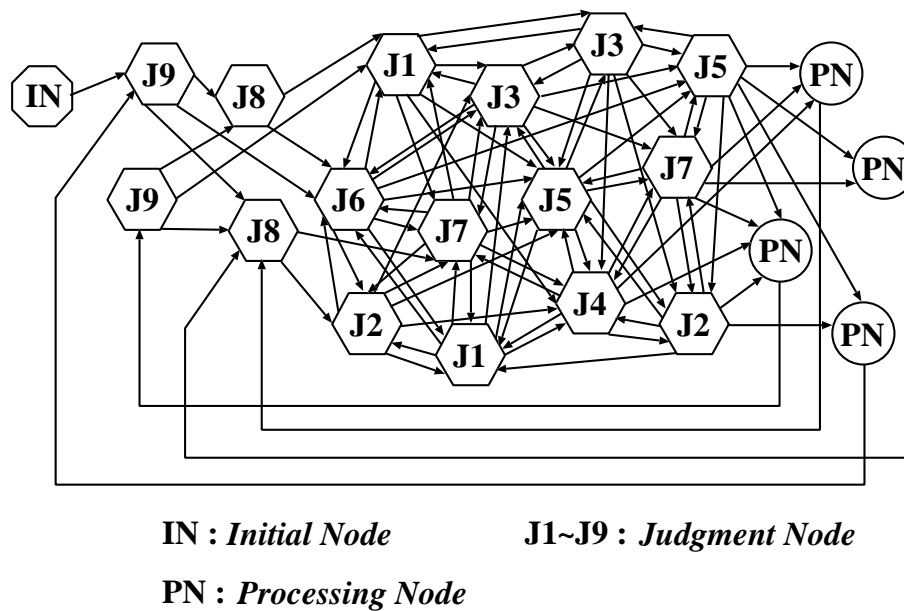


Fig. 2.2 An Example of GNP by Proposed Model

$$Fitness = \frac{1}{N} \sum_{n=1}^N (t_n)^2 + w_t \cdot (t_{max})^2 + w_l \cdot (l)^2 \quad (2.1)$$

ここで，

- N : 全乗客数
- t_n : n 番目の乗客の待ち時間
- t_{max} : 乗客の最大待ち時間
- l : 1 時間当たりのループ回数
- w_t, w_l : 各項の重み係数．

2.3 シミュレーション

本節では，提案モデルの検討をエレベータシミュレータを使用したシミュレーションにより行う．シミュレーションは GNP の進化による最適化，テストシミュレーションによる従来手法との性能比較，そしてノード使用頻度の検討の順に行う．

2.3.1 実行条件

Table 2.1，Table 2.2 に使用するエレベータシミュレータの仕様および GNP の進化の実行条件を示す．これらの条件の下，3 種類の交通流（平常時，アップピーク時，ダウンピーク時）に対し GNP の進化を実行する．GNP の評価はシミュレーション開始から 5 分後より開始し，最初の 5 分間の過渡状態は外部処理による群管理手法（発生呼びに最も早く到着可能なエレベータへ割当てて）を用いて群管理を行う．シミュレーションは EGSCS の確率的要素を考慮し，異なる 5 種類の乱数系列に対して実行した．

2.3.2 進化の様子

Fig. 2.3 (a)～(c) に各交通流における GNP の最良個体の適合度曲線を示す．結果はそれぞれ 5 種類の乱数系列の実行結果の平均を示している．

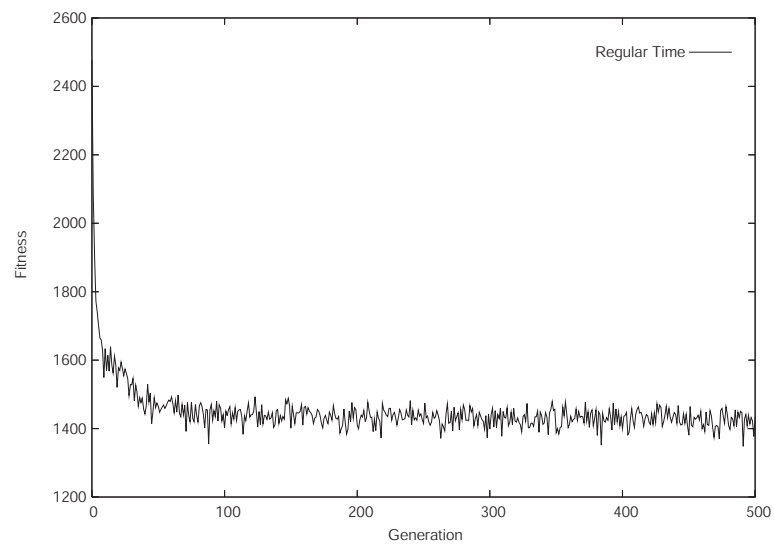
Fig. 2.3 を見ると，全体の傾向として適合度曲線は EGSCS の確率的要素により微小な振動が見られる．しかしいずれのケースも世代の経過につれて改善しており，GNP の EGSCS に対する適応性が示されている．また各適合度曲線は異なる形状を見せており，GNP は交通流毎に適切な群管理の最適化を実行していると考えられる．

Table 2.1 Specification of Elevator Simulator

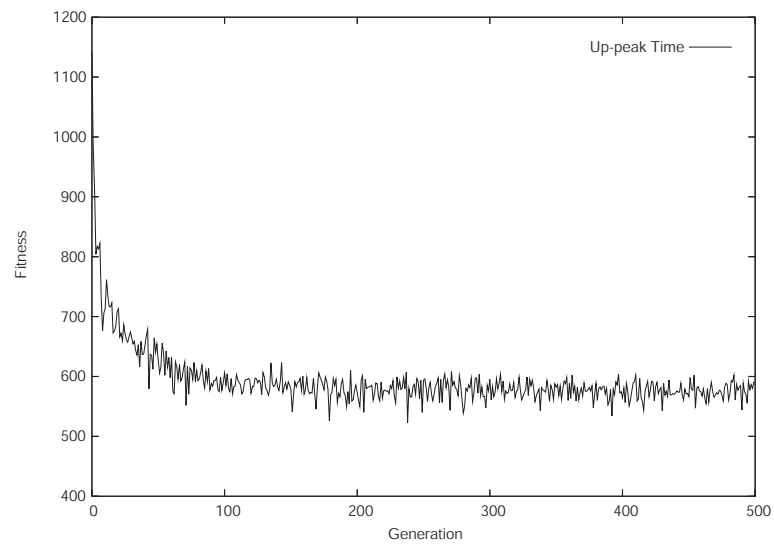
Items	Value
Number of Floors	16
Number of Elevators	6
Floor Distance [m]	4.5
Max. Velocity [m/s]	2.5
Max. Acceleration [m/s^2]	0.7
Jerk [m/s^3]	0.7
Cage Capacity [$person$]	20
Time for Opening Door [s]	2.0
Time for Closing Door [s]	2.3
Time for Riding [$s/person$]	1.0
Passenger Density [$person/h$]	
—Regular Time , Down-peak Time	2000
—Up-peak Time	1800

Table 2.2 Evolutional Conditions of GNP

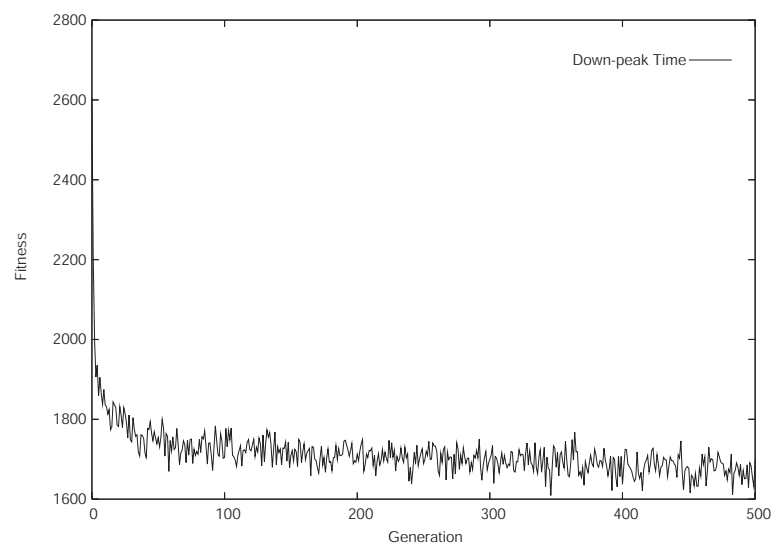
Items	Value
Generation	500
Population Size	300
—Crossover	116
—Mutation	174
—Elite	10
Node Size	42+Initial Node
—Processing Node	15
—Judgment Node	27 (3/seed)
Time Delay	
—Processing Node	5
—Judgment Node	2
—Branch	1
Time Delay Threshold	20
Crossover Probability P_c	0.1
Mutation Probability P_m	0.01
Evaluation Time [h]	2
w_t, w_l	0.007, 0.6



(a) Regular Time



(b) Up-peak Time



(c) Down-peak Time

Fig. 2.3 Fitness Curves in Each Traffic Flow

2.3.3 従来手法との性能比較

提案モデルの性能評価を行うために、進化で得られた GNP の最良個体（5 種類の乱数系列中の最良ケース）と固有の割当てルールによる従来手法との性能比較を行う。従来手法として、発生呼びへの到着予想時間 AT を考慮する AT 法[17] とエレベータ間の接近度を考慮した EC 法[24, 25] を用いる。これらは Eq. (1.1) のエレベータ評価式 $e(i)$ による群管理方式であり、それぞれ以下のルールに従って割当てエレベータ $d (\in I)$ を決定する。

【AT 法】 Eq. (1.1) で $X = AT$ とし、Eq. (2.2) により割当てエレベータを決定。

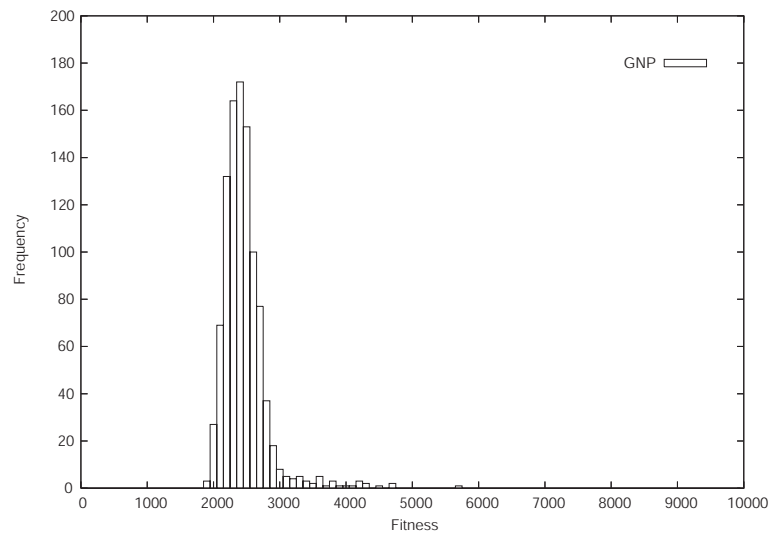
$$d = \arg \min_{i \in I} e(i) \quad (2.2)$$

【EC 法】 Eq. (1.1) で $X = AT$ とし、Eq. (2.3) により割当てエレベータを決定。

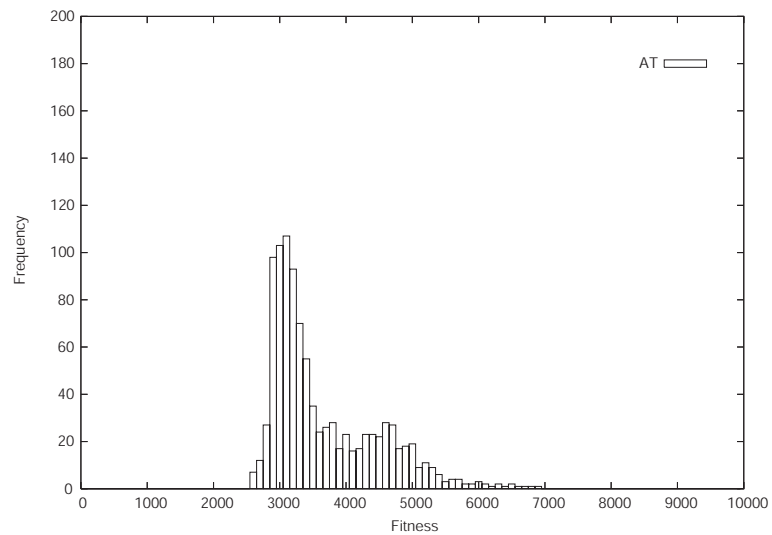
$$\begin{aligned} & \text{if } AT(AT_2) - AT(AT_1) > T \text{ then } d = AT_1, \\ & \text{else } d = AT_2 \end{aligned} \quad (2.3)$$

ここで、 $AT(i)$ はエレベータ i の AT の評価値、 $AT_1, AT_2 (\in I)$ はそれぞれ AT に関する上位 2 台のエレベータ ID、 T は接近度を判定する閾値時間（本検討では 22.5 [s]）である。

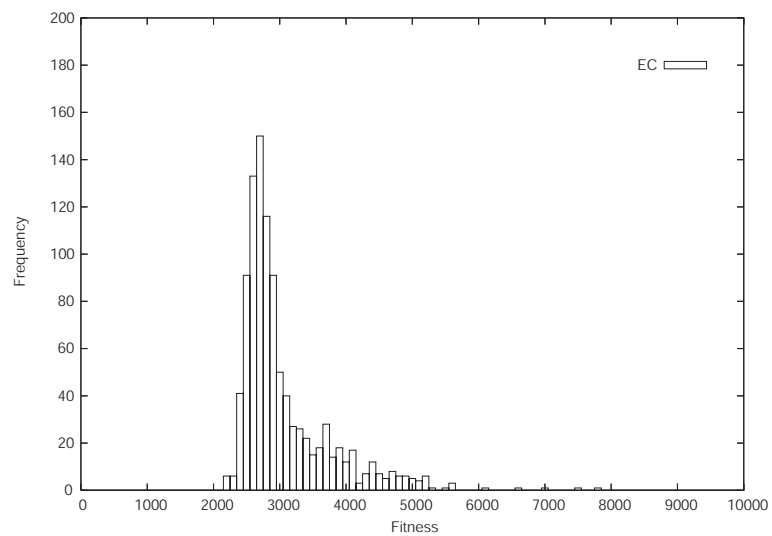
GNP と上記 2 手法を用いて、24 [h] \times 1000 試行のシミュレーションを実行した。Fig. 2.4~Fig. 2.6 は各手法の評価値として Eq. (2.1) により求めた適合度¹のヒストグラムを交通流毎に示したものであり、分布がばらつかず全体的に左側にあるほど良い性能を示す。3 手法を比較すると各交通流において GNP は従来手法よりも明らかに適合度がばらつかず左側に分布している。これは到着時間や接近度といった単純な判断基準のみで割当てエレベータを決定する従来手法に比べ、GNP は呼び発生時の状況に応じて到着時間、乗客数、登録呼び数等の情報を考慮した割当てを行うためである。また GNP は適合度関数によって平均待ち時間に加え最大待ち時間も最適化するため、その最適化を考慮しない従来手法に対する群管理の効率化も達成しているといえる。



(a) Proposed GNP

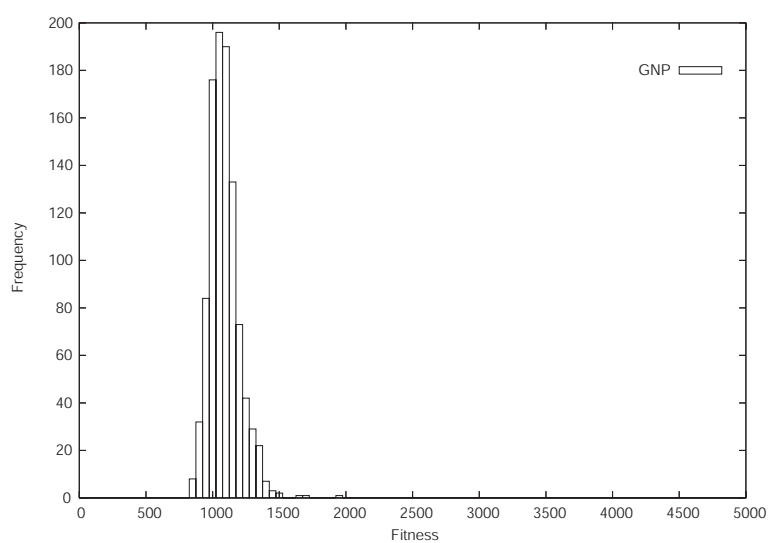


(b) AT Method

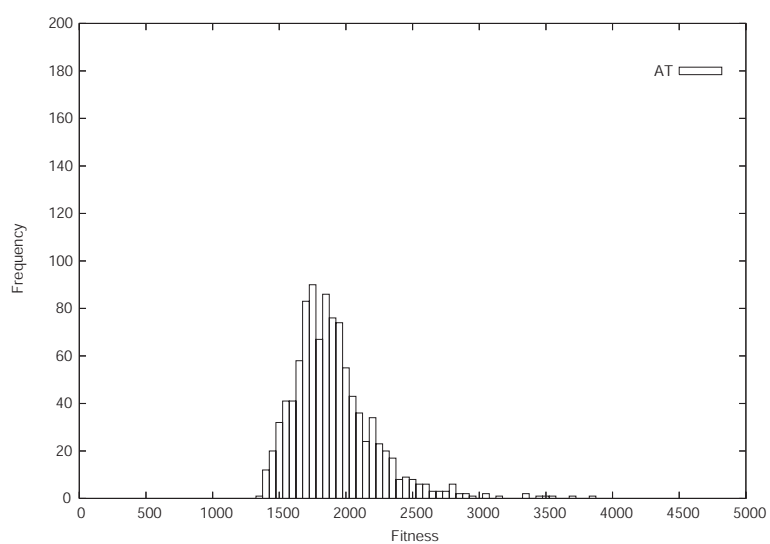


(c) EC Method

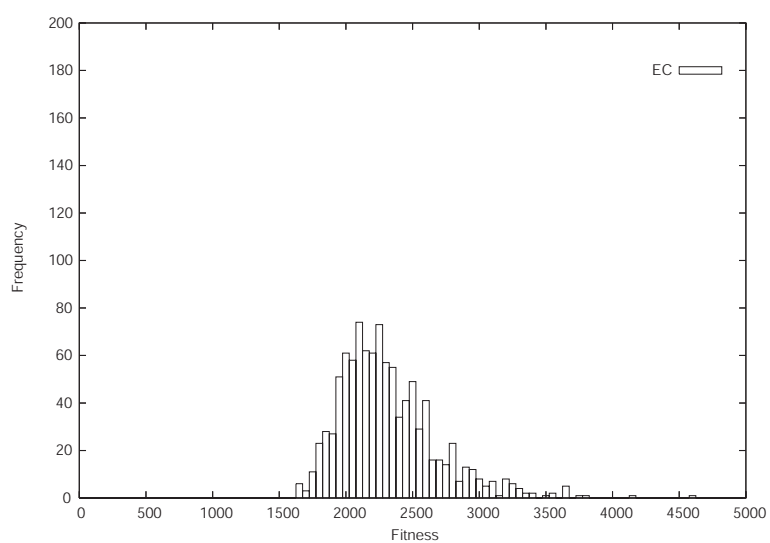
Fig. 2.4 Histogram of Fitness of Each Method in Regular Time



(a) Proposed GNP

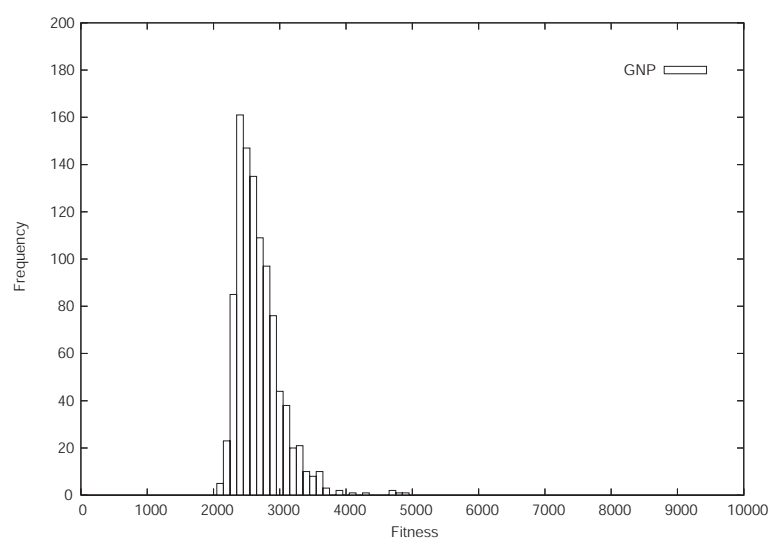


(b) AT Method

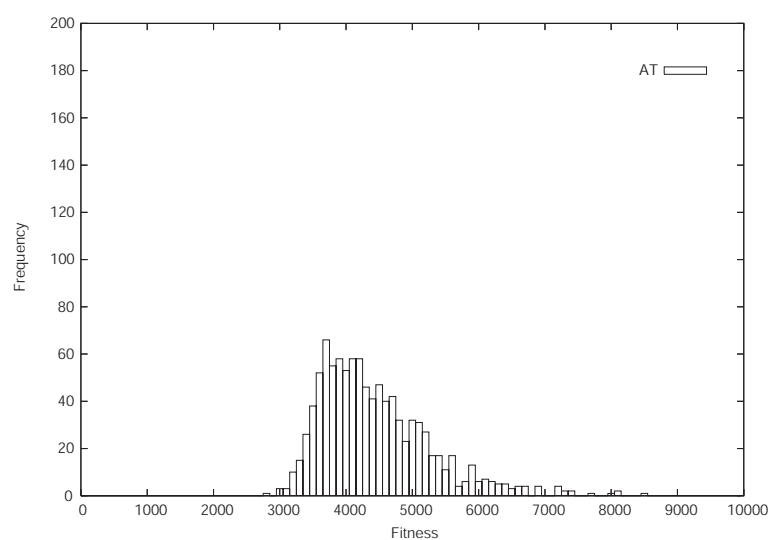


(c) EC Method

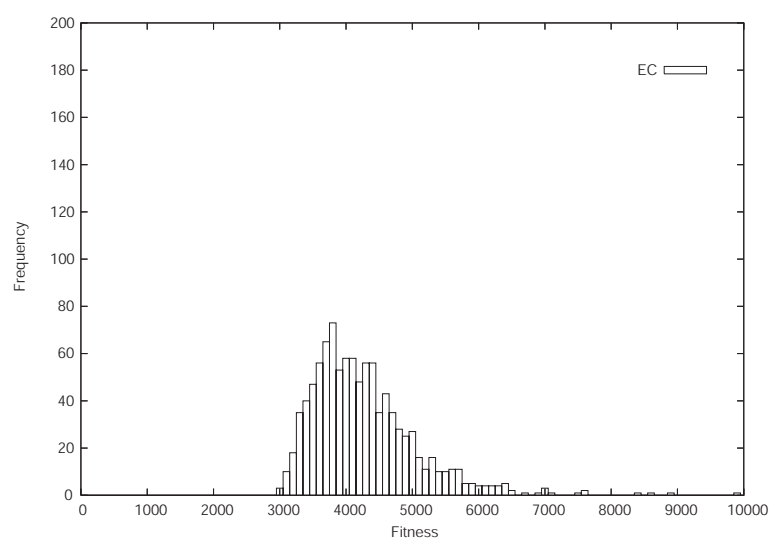
Fig. 2.5 Histogram of Fitness of Each Method in Up-peak Time



(a) Proposed GNP



(b) AT Method



(c) EC Method

Fig. 2.6 Histogram of Fitness of Each Method in Down-peak Time

次に、Table 2.3 に以下の 5 種類の指標 { 平均待ち時間 (AWT [s]) , 平均乗車時間 (ATT [s]) , 最大待ち時間 (MWT [s]) , 60 [s] 以上の長待ち乗客率 (LWP [%]) , 待ち時間の分散 (VWT [s^2]) } に関する各手法の平均および標準偏差を示す。Table 2.3 中の太字の数値は GNP が従来手法に対して性能が改善したケースである。Table 2.3 を比較すると、各交通流において GNP はほぼ全ての指標で従来手法に対して性能が改善している。特に注目すべき指標として、平常時：MWT、アップピーク時：AWT、LWP、ダウンピーク時：MWT 等が挙げられる。このような性能向上の理由として、平常時やダウンピーク時では一般階から基準階へ移動する乗客の積み残しによる長待ちが発生しやすいが、GNP は最大待ち時間の最適化により積み残しの解消を考慮するため、それを考慮しない従来手法に比べて性能が改善したと考えられる。またアップピーク時では基準階から一般階へ移動する乗客が大部分を占めるため、基準階へいかに早くエレベータをサービスするかが性能の向上に影響する。従来手法が用いる到着予想時間 AT のみでは乗客の行先階による不確定要素の影響から適切なエレベータを決定できない。これに対し GNP は各種情報を考慮し、より早く基準階へサービス可能なエレベータを適切に決定できるため、従来手法よりも性能が改善していると考えられる。以上の考察より、提案モデルの従来手法に対する有効性を明らかにした。

2.3.4 ノード使用頻度の検討

本節では、進化で得られた GNP の最良個体のノード使用頻度を調べ、GNP の最適な遺伝子構造および獲得した群管理ルールについて検討する。Fig. 2.7 (a)~(c) に、各交通流における GNP のノード使用頻度を示す。Fig. 2.7 中の JN1~JN9 はそれぞれ各判定ノードのラベルを示している。

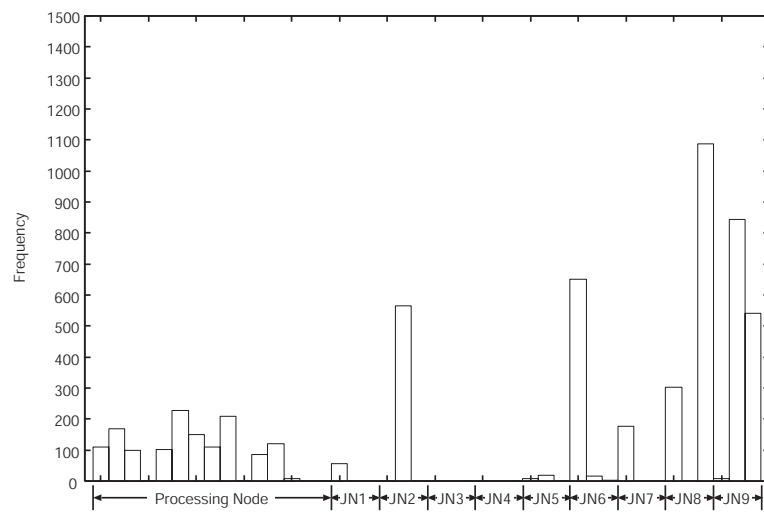
全体的なノード使用頻度の傾向としては、処理ノードおよび JN8、JN9 はいずれのケースにおいても頻繁に使用されている。これは Fig. 2.2 に示すように、処理ノード実行後は必ず JN8、JN9 へ遷移するためである。すなわち、GNP のノード遷移は“新規呼び階、エレベータの接近度の判定 (JN8、JN9)” → “エレベータに関する情報の判定

¹ループ項は GNP に特有の項であるため、従来手法の場合は 0 とする。

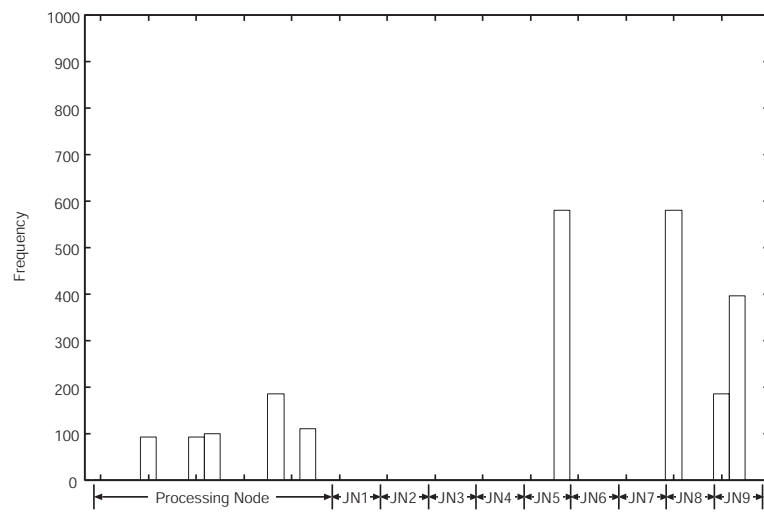
Table 2.3 Performance of Comparison of Each Method in Test Simulations

Method	AWT [s]		ATT [s]		MWT [s]		LWP [%]		VWT [s^2]	
	Ave.	S.D.	Ave.	S.D.	Ave.	S.D.	Ave.	S.D.	Ave.	S.D.
Regular Time										
GNP	25.64	0.291	70.98	0.325	381.8	53.22	11.40	0.296	813.7	29.14
AT	28.30	0.400	74.10	0.339	508.7	98.86	13.20	0.392	1023	58.77
EC	26.24	0.309	72.05	0.349	459.7	88.42	11.73	0.291	887.8	44.96
Up-peak Time										
GNP	21.05	0.789	86.55	0.317	194.2	30.10	4.821	0.746	400.2	31.64
AT	29.34	2.103	88.28	0.335	213.2	31.81	13.08	2.190	741.4	158.2
EC	33.18	2.307	88.12	0.329	218.0	30.67	17.02	2.349	894.5	172.9
Down-peak Time										
GNP	31.81	0.309	58.41	0.214	355.5	57.41	13.88	0.345	806.4	29.69
AT	32.51	0.480	62.34	0.233	549.4	93.40	13.70	0.365	1260	103.5
EC	32.91	0.449	61.38	0.214	532.5	89.06	13.99	0.337	1194	93.89

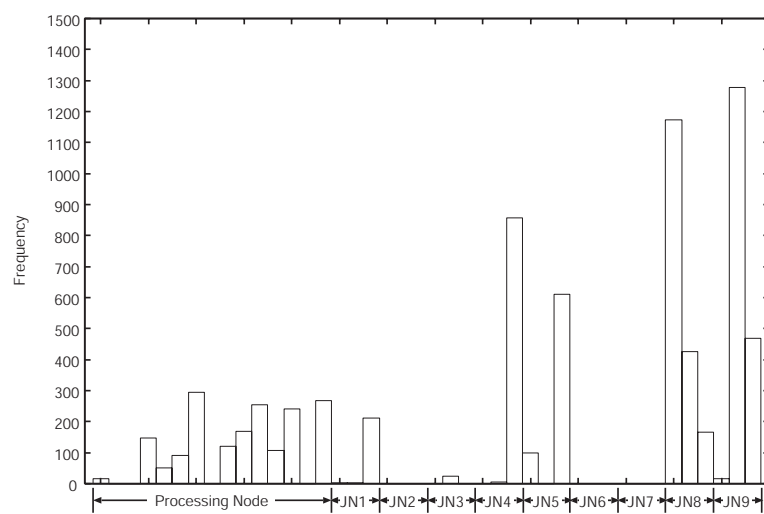
(JN1~JN7)”→“呼びの割当て (Processing Node)”という基本的な流れに沿って実行され、交通流毎に必要な情報を JN1~JN7 から入力して群管理ルールを生成している。具体的には、平常時では JN1, JN2, JN6, JN7 が主に使用されていることから、エレベータの到着時間や接近度、登録呼び数等を考慮した群管理ルールを実行している。アップピーク時では JN5 のみを使用しており、登録呼び数が少なく早く到着可能なエレベータへ割当てするルールを獲得している。ダウンピーク時は JN1, JN4, JN5 を主に使用し、到着時間に加え乗客数、登録呼び数を考慮した群管理ルールを実行している。しかしながら、これら一連のノード遷移から群管理ルールを記述した場合、処理ノードへの引数はその直前の判定結果によって決定されるため、それ以前の判定結果のエレベータ ID が引数と必ずしも一致しないケースが考えられ、この点で提案モデルは問題点を残していることがいえる。



(a) Regular Time



(b) Up-peak Time



(c) Down-peak Time

Fig. 2.7 Node Usage Frequency of GNP in Each Traffic Flow

2.4 まとめ

本章では GNP による EGSCS の初期検討として、GNP の基本的枠組みに基づく基本モデルを提案し、シミュレーションによる性能評価を行った。シミュレーション結果から、GNP の EGSCS に対する導入可能性を確認し、テストシミュレーションでは各交通流において従来手法より優れた性能を示す結果が得られた。またノード使用頻度の検討により、GNP の最適な遺伝子構造および獲得した群管理ルールについても様々な知見が得られた。

しかしながら群管理ルールの検討で示したように、提案モデルの GNP は以下の問題点を残していることが明らかとなった(1) 処理ノードのみに引数を導入しているため、実際の割当ては処理ノードの直前の判定結果に大きく依存することとなる。すなわち、それ以前の判定ノードの判定結果(エレベータ ID)が実際の割当てエレベータ ID と一致しないケースが考えられ、ルールの解釈の点で矛盾が生じる(2) 固有のノード関数を使用しているため、それらに依存した群管理ルールが生成され、柔軟な群管理の最適化が難しくなる。上記2点から、提案モデルによる一般性が高く効率的な群管理ルールの生成は困難となる。また、基本モデルの性能評価は固定の交通条件下で行っているが、実際のエレベータシステムのように時間経過と共に変化する交通条件下での性能評価も必要と考えられる。次章第3章では上記問題点の解決を目指したモデルの拡張について検討を行う。

第3章

ランキング処理とノード関数最適化を考慮した GNP による群管理システム

3.1 序

第2章では、GNP による EGSCS の基本モデルの提案、検討を行い、その導入可能性および従来手法に対する有効性を確認した。しかし一方で、以下の問題点が浮彫りとなっている。

- (1) 引数処理が処理ノードのみであり、一連の判定ノードの判定結果と割当てエレベータが必ずしも一致せず、一般性の高い群管理ルールを生成できない。
- (2) 設計者が定義した固有のノード関数に依存した GNP の進化を行うため、多様な交通条件に対する柔軟な群管理の最適化が困難となる。
- (3) 動的に変化する交通条件下での性能評価を行っていない。

これらが示しているように、EGSCS はビルやエレベータの仕様、交通流といった条件の変更によって制御対象が異なる。したがって、多様な条件に適応しうる柔軟かつ一般性の高い群管理システムの構築は重要である。本章では上記問題点 (1)、(2) の解決のため、各種の先験知識に基づくエレベータ評価項目によるランキング処理およびノード関数最適化を考慮した、GNP による EGSCS の拡張モデルを提案する。

以下、ランキング処理とノード関数最適化、その導入による提案モデルの概要について述べ、シミュレーションでは上記問題点 (3) に対し、動的に変化する交通条件下での提案モデルの性能評価を行う。

3.2 導入アルゴリズム

本節では新規導入アルゴリズムの、評価項目によるエレベータのランキング処理とノード関数最適化について述べる。

3.2.1 ランキング処理

基本モデルの GNP では判定ノードで判定結果に該当するエレベータを選定し分岐する。しかしこれは判定結果であるエレベータ ID がその後の判定ノードの遷移において有効に活用されず（割当てエレベータとの不一致）、また判定結果がエレベータ数に依存し一般性に乏しい点で課題を残した。その解決のため、新たに先験知識に基づく各種の評価項目によるエレベータのランク付けを行い、GNP がランキング情報により適切なエレベータ ID を選定し、それを引数として判定ノード処理を実行する方式を提案する。

評価項目 X ($\in \Gamma$: 評価項目の集合) のランク n のエレベータ ID を X_n とする。Fig. 3.1 に示すように、エレベータ選定処理ノード（ノードラベル $S(X_n)$ ）ではエレベータ ID: X_n を計算し、これを引数として以降のノード遷移へ渡す。 $S(X_n)$ から接続するエレベータ情報判定ノード（ノードラベル $J^Y(X_n)$, X_n : 引数）ではエレベータ X_n の評価項目 Y に関する判定を行う。そして $J^Y(X_n)$ から割当て処理ノード（ノードラベル $A(X_n)$, X_n : 引数）へ遷移すると、発生呼びのエレベータ X_n への割当てを行う。以上のノード遷移により、判定ノードで引数処理を行わない基本モデルの GNP と比較して、各種評価項目を考慮した一般性の高い群管理ルールの構成が可能となる。

3.2.2 ノード関数最適化

従来の GNP では、ノードの内容は設計者によって予め定義され、進化を通して不変であった。しかし EGSCS のように扱う問題が複雑になると、適切なノード設計は困難となる。したがってそのような問題に対しては、GNP 自身が適応的にノードを獲得することが望ましい。このノード獲得のアプローチとして、GNP のエレベータ情報判定ノード $J^Y(X_n)$ にノードパラメータを設定し、これを実数値 GA (Real-coded Genetic Algorithm) [57] によって最適化するアルゴリズムを新たに提案する。

今, エレベータ i ($\in I$: エレベータ ID の集合) の評価項目 Y ($\in \Gamma$) の評価値を $Y(i)$ として, この正規化変数 $y(i)$ を Eq. (3.1) で求める.

$$y(i) = \frac{Y(i)}{\max_{i \in I} Y(i)} \quad (3.1)$$

Eq. (3.1) を利用し, 評価項目 X のランク n のエレベータ X_n を引数とするエレベータ情報判定ノード j ($\in J$: エレベータ情報判定ノードの添字の集合) では,

$$if \ y(X_n) \leq r_j^Y \quad (3.2)$$

の判定を行うことになる. すなわち $J^Y(X_n)$ では, “評価項目 X で選定したエレベータの評価項目 Y に関する評価値がノードパラメータ r_j^Y 以下かを判定する” 内容の処理が実行される. 提案モデルでは r_j^Y ($\in [0, 1]$) を実数値 GA により GNP の進化と併せて最適化する. なお, 実数値 GA の遺伝的オペレータ (交叉, 突然変異) のアルゴリズムに関しては, 付録 C を参照されたい. 以上の結果, 提案モデルでは判定ノードに含まれるパラメータの最適化をも考慮した GNP の進化が可能になり, 基本モデルより柔軟で優れた特性を持つ GNP を構成できる.

3.3 ランキング処理とノード関数最適化による拡張モデル

本節では, 前節の 2 種類のアプローチを導入した拡張モデルについて述べる. 評価項目として以下の 5 種類を考慮する.

- AT : 呼び発生階へのエレベータの到着予想時間
- PN : エレベータ内の乗客数
- CN : エレベータの登録ホール呼び数
- SI : エレベータの現在階と呼び発生階間の登録呼びを考慮したサービス指標
- MS : エレベータの全登録ホール呼びに対する予想待ち時間の最大値

AT はエレベータの登録呼び数, 移動方向等を考慮して計算する. PN , CN は実測値を使用する. SI はエレベータの現在階 f_i , 呼び発生階 f_{nc} , 呼び発生階の直前の停止階 f_{pc} とその呼びの種類 $call$ による指標で, エレベータが呼び発生階へ向かう場合の $call$ による場合分けを行い Eq. (3.3) のように計算される.

$$\begin{aligned}
& \text{if } \quad \quad \text{call} = CAGE \quad \text{then } SI(i) = |f_{pc} - f_{nc}|, \\
& \text{else if } \quad \text{call} = HALL \quad \text{then } SI(i) = |2f_i - f_{pc} - f_{nc}|, \\
& \text{else if } \quad \text{call} = NONE \quad \text{then } SI(i) = |f_i - f_{nc}|
\end{aligned} \tag{3.3}$$

ここで, $CAGE$, $HALL$, $NONE$ はそれぞれかご呼び, ホール呼び, 呼びなしを意味し, エレベータが呼び発生階から遠ざかっている場合, SI には大きい値を代入する. すなわち, SI は呼び発生階への到着時の乗車人数の程度を直前の登録呼びを考慮して見積もった指標である. MS はエレベータの全登録ホール呼びに対する (到着予想時間 + 登録後の経過時間) の最大値であり, Eq. (3.4) により求める.

$$MS(i) = \max_{h \in H} (apt_h(i) + et_h(i)) \tag{3.4}$$

ここで H は登録ホール呼びの添字の集合, $apt_h(i)$, $et_h(i)$ はそれぞれエレベータ i の登録ホール呼び h に関する到着予想時間および登録後の経過時間である. また上記評価項目は, AT は乗客の待ち時間の短縮に, PN , SI は長待ちを引き起こす乗客の積み残しの解消に, そして CN , MS はシステム全体の運行バランスの最適化にそれぞれ寄与する. 以下, これらを考慮した提案モデルの概要を述べる.

3.3.1 基本構成

提案モデルの基本構成を Fig. 3.2 に示す. システムはエレベータシステム, GNP コントローラおよびランキング処理部 (情報管理部) からなる.

GNP による発生呼びの割当てメカニズムを以下に示す. 乗客がフロアに出現しホール呼びボタンを押したとき, GNP は初期ノードより起動しノード遷移を開始する. 始めにシステム情報判定ノード (ノードラベル JEC , JFD) でエレベータ間の接近度や発生呼びの種類を判定する. そして $S(X_n)$ で X_n を計算後, $J^Y(X_n)$ でエレベータ情報の判定を行う. $A(X_n)$ で発生呼びのエレベータ X_n への割当てを行うが, その後の次ホール呼び発生までノード処理は $A(X_n)$ に留まる. 提案モデルはノード間の接続に制約を設け, 進化の効率化を図っている. ノード遷移の際, 累積遅れ時間が予め設定した閾値を超えた場合は GNP の実行を停止し, 代わりにエレベータ AT_1 へ呼びを割当てる.

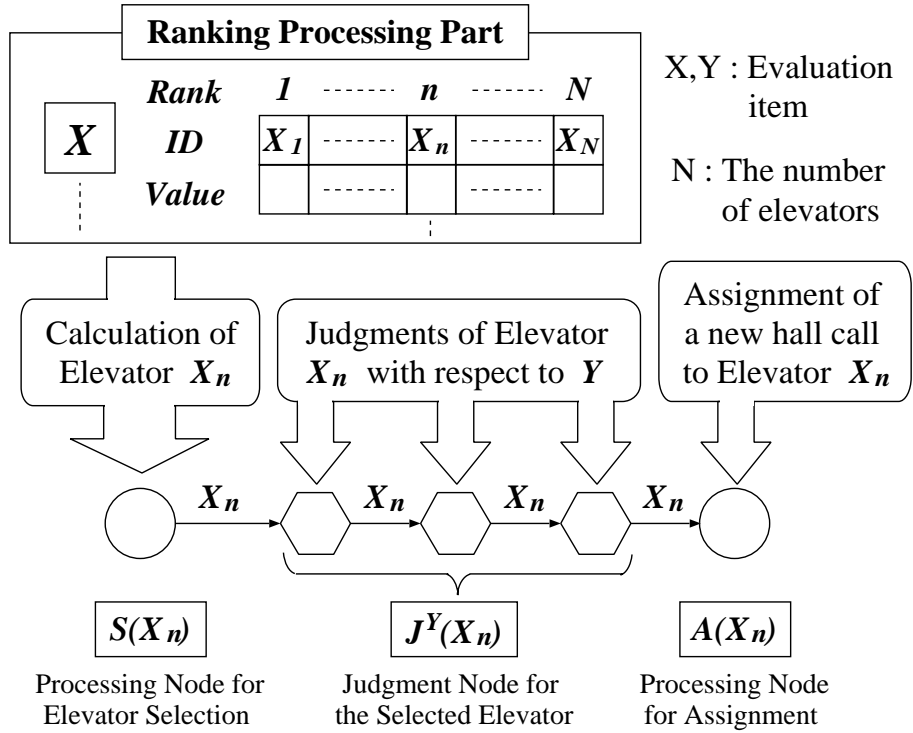


Fig. 3.1 Ranking Processing Part and Nodes with Argument

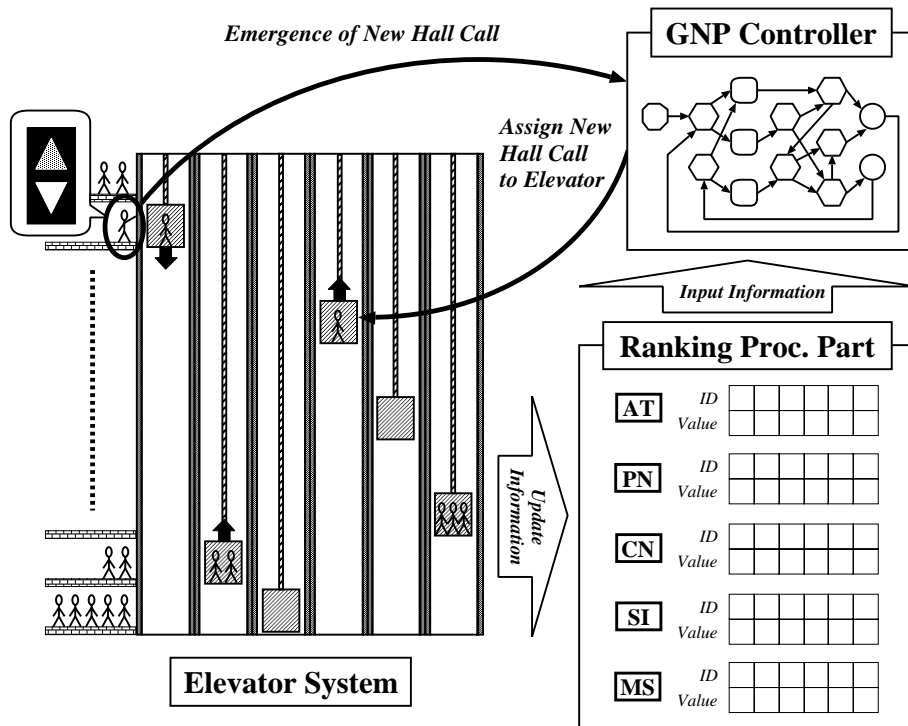


Fig. 3.2 Basic Structure of Proposed Model

3.3.2 ノード関数

提案モデルのGNPのノード関数として、以下の18種類を用意した。

処理ノード（11種類）

- 割当て処理ノード（1種類）
 - $A(X_n)$ ：エレベータ X_n への発生呼びの割当て（ X_n ：引数）
- エレベータ選定処理ノード（2種類×5項目、計10種類）
 - $S(X_1)$ ：エレベータ X_1 を計算
 - $S(X_2)$ ：エレベータ X_2 を計算

$$\begin{cases} X \in \Gamma \\ \Gamma = \{AT, PN, CN, SI, MS\} \end{cases}$$

判定ノード（7種類）

- エレベータ情報判定ノード（1種類×5項目、計5種類）
 - $J^Y(X_n)$ ： $y(X_n) \leq r_j^Y$ かを判定（ X_n ：引数）
- システム情報判定ノード（2種類）
 - JEC ：エレベータ AT_1 と AT_2 の接近度を判定
 - JFD ：呼び発生階と呼び方向を判定

$$\begin{cases} Y \in \Gamma, j \in J \\ \Gamma = \{AT, PN, CN, SI, MS\} \end{cases}$$

$A(X_n)$ は引数としてとるエレベータ X_n に発生呼びを割当てる（有向枝：1本）。 $S(X_1)$, $S(X_2)$ はランキング情報を利用し、ノードラベルに記載された評価項目のランクが1および2のエレベータをそれぞれ計算する（有向枝：1本）。 $J^Y(X_n)$ は引数としてとるエレベータ X_n の評価項目 Y の正規化変数 $y(X_n)$ がノードパラメータ以下かどうかを判定する（有向枝：2本）。 JEC は AT のランクが1と2のエレベータの時間差が閾値時間 T 以下かを判定する（有向枝：2本）。 JFD は呼び発生階と呼びの方向の組合せが{基準階，一般階低階層/下方向，一般階低階層/上方向，一般階高階層/下方向，一般階高階層/上方向}のいずれに該当するかを判定する（有向枝：5本）。 JEC , JFD はエ

エレベータ選定を行う前に予め呼び発生時のシステムの状況を分類し，その後のノード遷移によって各状況に応じた割当てルールを生成するために導入している．

3.3.3 適合度関数

提案モデルの適合度関数 *Fitness* は基本モデルと同じく，待ち時間の最小化とループ遺伝子の淘汰を考慮する Eq. (3.5) によって計算する．

$$Fitness = \frac{1}{N} \sum_{n=1}^N (t_n)^2 + w_t \cdot (t_{max})^2 + w_l \cdot (l)^2 \quad (3.5)$$

ここで，

- N : 全乗客数
- t_n : n 番目の乗客の待ち時間
- t_{max} : 乗客の最大待ち時間
- l : 1 時間当たりのループ回数
- w_t, w_l : 各項の重み係数．

3.4 進化による最適化のシミュレーション

ここでは，提案モデルの GNP を多様な交通条件の下で進化によって最適化するシミュレーションについて検討する．

3.4.1 実行条件

Table 3.1，Table 3.2 にシミュレーションで使したエレベータシステムの仕様および GNP の進化の実行条件を示す．Table 3.2 ではエレベータ情報判定ノードを 25 個に設定しているため，ノードパラメータ数も 25 個となる．ノードサイズ，交叉／突然変異率等の GNP の進化条件は全て経験的に設定している．また交叉および突然変異個体数の（ ）内の数値はノードパラメータの交叉，突然変異を併せて実行する個体数を示す．上記条件の下，Table 3.3 に示す 6 種類の交通条件（*Case* 1 ～ 6）に対する GNP の進化を実行した．GNP の評価は過渡状態を避け，シミュレータ起動後 5 分後より開始する．またシミュレーションは EGSCS の確率的要素を考慮し，異なる 5 種類の乱数系列に対して実行した．

Table 3.1 Specification of Elevator Simulator

Items	Value
Number of Floors	16
Number of Elevators	6
Floor Distance [m]	4.5
Max. Velocity [m/s]	2.5
Max. Acceleration [m/s^2]	0.7
Jerk [m/s^3]	0.7
Cage Capacity [$person$]	20
Time for Opening Door [s]	2.0
Time for Closing Door [s]	2.3
Time for Riding [$s/person$]	1.0

Table 3.2 Evolutional Conditions of GNP

Items	Value
Generation	400
Population Size	300
—Crossover (with Node Para. Cross.)	110 (100)
—Mutation (with Node Para. Mut.)	160 (150)
—Node Parameter Crossover, Mutation	10, 10
—Elite	10
Node Size	74+Initial Node
— $\mathbf{A}(X_n)$	15 (15/seed)
— $\mathbf{S}(X_1), \mathbf{S}(X_2)$	30 (3/seed)
— $\mathbf{J}^Y(X_n)$	25 (5/seed)
— $\mathbf{JEC}, \mathbf{JFD}$	4 (2/seed)
Time Delay	
— $\mathbf{A}(X_n)$	5
— $\mathbf{S}(X_1), \mathbf{S}(X_2)$	2
— $\mathbf{J}^Y(X_n), \mathbf{JEC}, \mathbf{JFD}$	1
—Branch	0
Time Delay Threshold	20
Crossover Probability P_c, P_c^{np}	0.1, 0.1
Mutation Probability P_m, P_m^{np}	0.01, 0.01
Evaluation Time [h]	3
w_t, w_l	0.007, 0.7

3.4.2 進化の様子

Fig. 3.3 に各 *Case* における GNP の最良個体の適合度曲線を示す．結果はそれぞれ 5 種類の乱数系列の実行結果の平均である．

Fig. 3.3 では，全体の傾向として最良個体の適合度の推移には EGSCS の確率的要素により多少の振動が見られるが，世代の経過により概ね改善されており，各交通条件下で GNP の進化による最適化が行われていることがわかる．また EGSCS では発生する乗客数の増加に伴い問題が複雑となるため，混雑時 (*Case* 2, 4, 6) が閑散時 (*Case* 1, 3, 5) よりも適合度の収束値が大きく，収束へ多くの世代を要している．

3.4.3 ノード関数最適化の効果の検討

ここでは，ノード関数最適化の効果について検討する．Table 3.3 の *Case* 2 の交通条件下で， $NPOA_{cont}$ ， $NPOA_{0.05}$ ， $NPOA_{0.1}$ ， \overline{NPOA} の 4 種類のケースの進化を実行し，性能比較を行った．ここで $NPOA_{cont}$ はノードパラメータ r_j^Y が連続値をとるケース， $NPOA_{0.05}$ ， $NPOA_{0.1}$ は r_j^Y がそれぞれ 0.05，0.1 刻みの離散値をとるケース，そして \overline{NPOA} は最適化を行わないケースである．それぞれの進化のシミュレーションは異なる乱数系列 5 種類に対して実行した．

各ケースの進化後の最良個体 5 個（乱数系列毎）に対し，進化と同条件で $24 [h] \times 100$ 試行のシミュレーションをそれぞれ実行した結果の適合度平均および標準偏差を Table 3.4 に示す．結果の比較より，ノードパラメータ最適化の効果が見てとれる．またノードパラメータ最適化を使用する 3 ケースの中では $NPOA_{0.05}$ が最も性能が良く，これはパラメータの探索幅が適切に設定されていることを示す．

3.5 テストシミュレーションによる性能評価

本節では進化で得られた 6 種類の GNP の最良個体（5 種類の乱数系列中の最良ケース）に対するテストシミュレーションを実行し，基本モデルおよび固有の群管理ルールによる従来手法との性能比較，獲得した群管理ルールの検討を行う．

Table 3.3 Simulation Cases

Case	Traffic Flow	Passenger Density [<i>person/h</i>]	T [<i>s</i>]
1	Regular Time	1500	15.0
2		2400	25.0
3	Up-peak Time	1200	5.0
4		2000	15.0
5	Down-peak Time	1200	10.0
6		2000	20.0

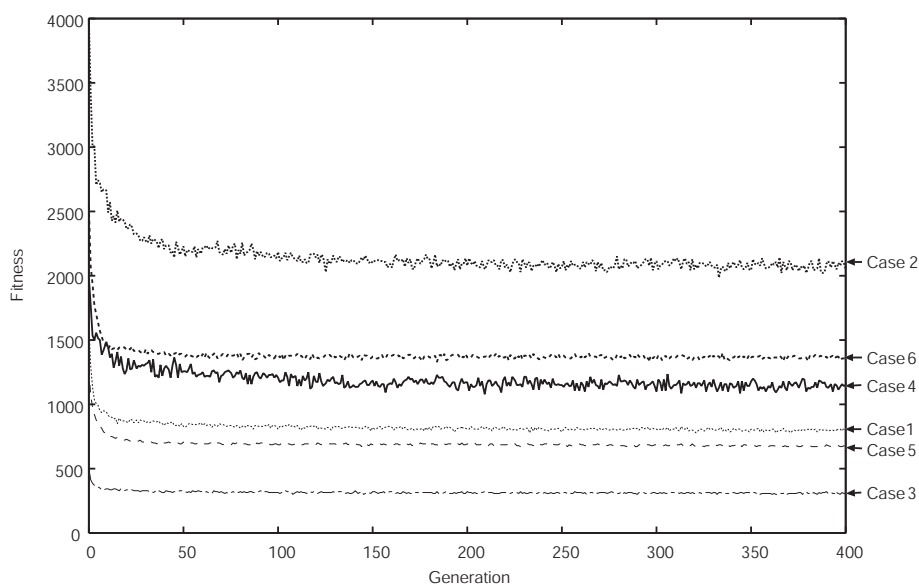


Fig. 3.3 Fitness Curves in Each Case

Table 3.4 Comparisons of Fitness with Various Implementation of Node Parameter Optimization Algorithm

Methods	Fitness	
	Average	Std. Dev.
$NPOA_{cont}$	4205	622.1
$NPOA_{dis0.05}$	4171	626.8
$NPOA_{dis0.1}$	4299	650.5
\overline{NPOA}	4463	623.4

3.5.1 実行条件

テストシミュレーションは Table 3.1 と同じ仕様のビルの 1 日の時間帯交通データを使用して実行する．Fig. 3.4 に交通データの概要を示す．これは時間経過に伴い上 / 下方向の乗客密度が動的に変化する交通データである．ビルの稼動時間帯は 8:00~19:00 の 11[h] であり，これは Table 3.5 に示すように出勤時（Incoming Time），勤務時（Business Time），昼食時（Lunch Time），退勤時（Outgoing Time）の 4 種類の時間帯に分類できる．Case 1 ~ Case 6 の GNP を，Table 3.5 のように予め設定した時間帯に適用し群管理性能を評価する．Table 3.5 の各時間帯に適用する GNP 個体は，Table 3.3 の交通条件と Fig. 3.4 の上 / 下方向の乗客数比および乗客密度を考慮して最適と思われるものを選択している．なお，この時間帯における GNP 選択の重要性の議論に関しては，付録 D. を参照されたい．またシミュレーションは乱数系列の異なる 1000 試行を実行した．

3.5.2 従来手法との性能比較

始めに，提案モデルと従来手法との性能比較を行う．従来手法は第 2 章の基本モデルの GNP および，固有の割当てルールに基づく 3 手法（AT 法，EC 法，THV 法[15]）を使用する．基本モデルの GNP は提案モデルと同様，6 種類の交通条件に対して進化させたものを Table 3.5 と同様に適用する．他の 3 手法のうち AT 法，EC 法は第 2 章と同様であり，THV 法は以下のルールによって割当てエレベータ d を決定する．

【THV 法】 ビル階床数を F ，エレベータ i の現在階から呼び発生階への移動距離を d_i ，エレベータの移動方向を dir_i ，発生呼び方向を dir_c として，Eq. (3.6) により割当てエレベータを決定．

$$\begin{aligned}
 & \text{if} \quad \quad \quad dir_i = dir_c \quad \quad \text{then} \quad e(i) = F + 1 - d_i, \\
 & \text{else if} \quad dir_i \neq dir_c \quad \quad \text{then} \quad e(i) = F - d_i, \\
 & \text{else if} \quad \textbf{Departure} \quad \quad \text{then} \quad e(i) = 1, \\
 & \text{else} \quad \quad \quad \quad \quad \quad \quad \quad e(i) = F - d_i,
 \end{aligned} \tag{3.6}$$

$$d = \arg \max_{i \in I} e(i)$$

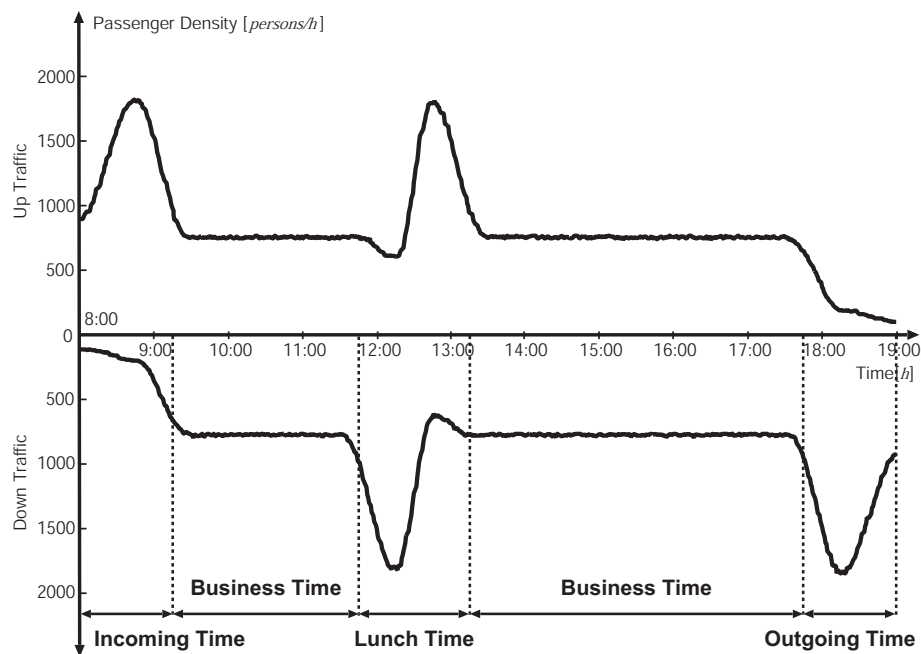


Fig. 3.4 Traffic Data of Test Simulations

Table 3.5 Time Table and Applied GNPs in Test Simulations

Classified Time	Time	Applied GNP
Incoming Time	8 : 00 ~ 8 : 30	$\langle Case 3 \rangle$
	8 : 30 ~ 9 : 15	$\langle Case 4 \rangle$
Business Time	9 : 15 ~ 11 : 45	$\langle Case 1 \rangle$
Lunch Time	11 : 45 ~ 12 : 00	$Case 1$
	12 : 00 ~ 12 : 15	$Case 6$
	12 : 15 ~ 12 : 45	$\langle Case 2 \rangle$
	12 : 45 ~ 13 : 00	$Case 4$
	13 : 00 ~ 13 : 15	$Case 1$
Business Time	13 : 15 ~ 17 : 45	$Case 1$
Outgoing Time	17 : 45 ~ 18 : 30	$\langle Case 6 \rangle$
	18 : 30 ~ 19 : 00	$\langle Case 5 \rangle$

ここで Eq. (3.6) の *Departure* はエレベータが呼び発生階から出発した（通過した）直後の状態を指す．なお，上式中の記号は全て提案モデルの表記に従っている．

AT 法，EC 法および THV 法は固有の群管理ルールにより呼びの割当てを行うため，適合度関数による評価や割当てルールの変更等は考慮しない．また出勤時のみ，呼びの割当てによる群管理とは別に，各手法とも一般階へのサービス終了後に自動的に基準階への配車を行う配車制御 [26] を採用している．

Table 3.6 に 4 種類の時間帯および全時間帯の上記 5 手法の性能比較を平均待ち時間 (AWT [s])，平均乗車時間 (ATT [s])，60 [s] 以上の長待ち乗客率 (LWP [%]) について示す．Table 3.6 ではテストシミュレーション 1000 試行の各指標の平均値と標準偏差を示しており，表中の太字の数値は提案モデルが最良となるケースを示す．

Table 3.6 より，概ね全ての時間帯において提案モデルは固定ルールによる 3 手法よりも各指標に対して性能が改善している．特に交通条件が厳しい昼食時では顕著な性能の改善を確認できる．これは全時間帯を通じて固定のルールによる群管理を行う 3 手法に対し，提案モデルおよび基本モデルの GNP は 6 種類の GNP を交通条件の変化に応じて適切に使い分けることで効率的な群管理を実行できたためと考えられる．また，提案モデルでは新規アルゴリズムの導入によって基本モデルより柔軟な進化が可能となり，特に退勤時において顕著な性能の改善を達成している．

勤務時では交通条件が比較的緩やかなため，他時間帯と比べてあまり性能差は見られない．これは緩やかな交通条件ではエレベータホールへ出現する乗客が少なくホール呼びが頻繁に発生せず，どの手法でも十分なサービスが可能となるためである．一方，厳しい交通条件ではホール呼びも多数発生し，効率的な群管理を実行するためには適切なエレベータへの割当てが要求される．提案モデルでは 5 種類の評価項目を考慮した割当てルールを生成するため厳しい交通条件下でも効率的な群管理を実行でき，従来手法との性能差が顕著となる．

また，全時間帯を通じての GNP のループ率 (ループ回数 / GNP の起動回数) の平均は 1.33×10^{-3} であり，提案モデルの GNP のループはほぼ起こらないことがわかる．すなわち GNP は進化を通じて効率的にループ遺伝子を淘汰し，適切な呼びの割当てを行う構造を獲得している．

Table 3.6 Comparison of Performance among Proposed GNP and Other Methods

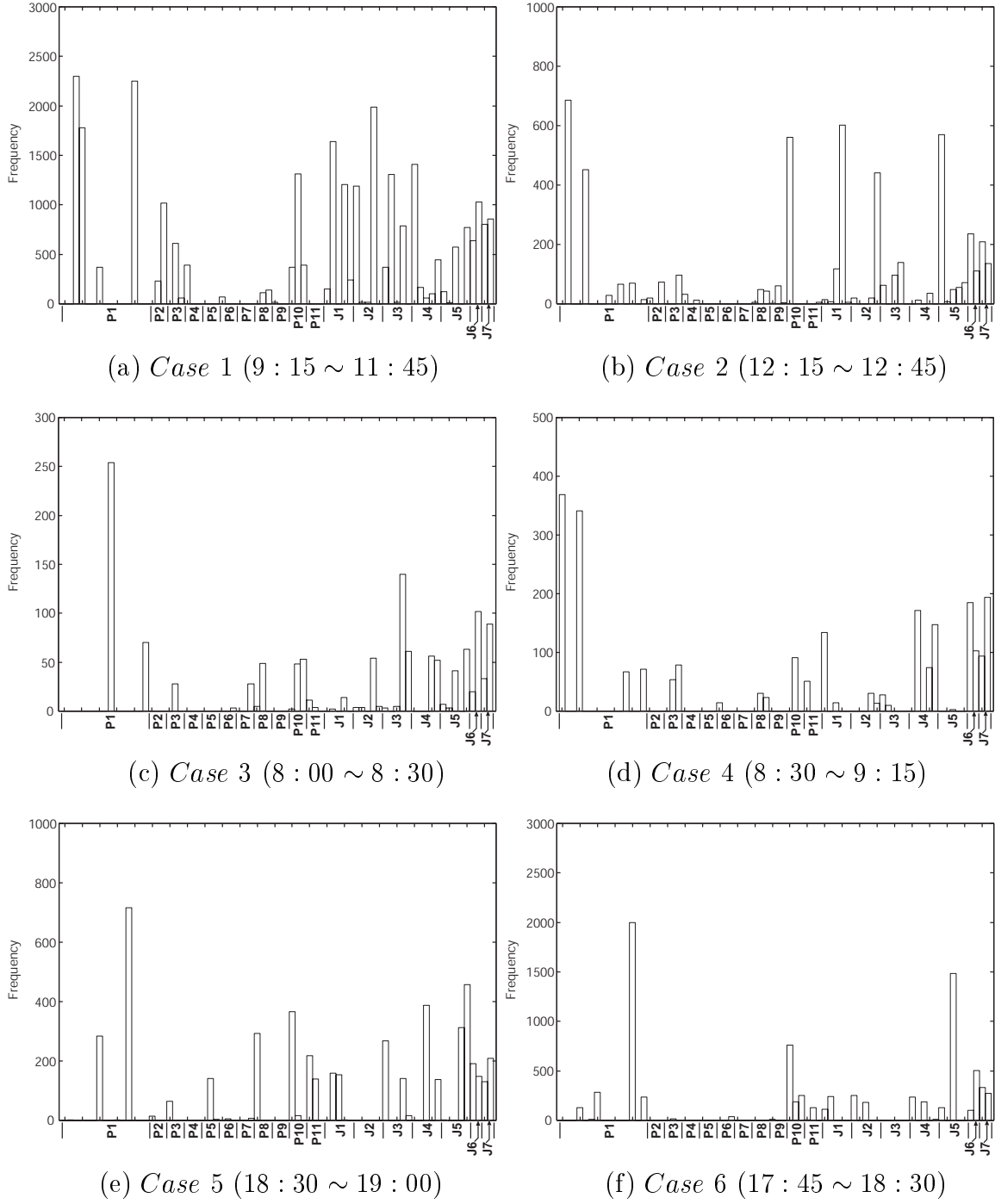
Method	AWT [s]		ATT [s]		LWP [%]	
	Ave.	S.D.	Ave.	S.D.	Ave.	S.D.
Incoming Time (8:00 ~ 9:15)						
Proposed GNP	19.86	2.898	77.16	1.567	6.438	2.964
Basic GNP	20.17	3.052	77.38	1.460	6.525	3.185
AT	21.96	3.427	78.13	1.479	8.475	3.736
EC	21.17	3.498	77.82	1.560	7.692	3.715
THV	20.39	3.158	77.44	1.532	7.360	3.262
Business Time (9:15 ~ 11:45, 13:15 ~ 17:45)						
Proposed GNP	20.85	0.379	57.67	0.629	4.744	0.408
Basic GNP	21.02	0.373	58.45	0.624	6.175	0.412
AT	21.62	0.423	59.28	0.671	6.792	0.471
EC	21.12	0.383	58.26	0.644	6.159	0.430
THV	22.48	0.488	60.27	0.746	8.528	0.500
Lunch Time (11:45 ~ 13:15)						
Proposed GNP	38.01	4.904	77.04	1.245	22.90	4.885
Basic GNP	37.72	4.783	77.75	1.394	21.92	4.736
AT	43.91	6.369	79.83	1.375	27.52	4.858
EC	40.92	5.822	79.17	1.366	24.74	4.785
THV	53.19	8.313	84.19	1.599	31.08	5.267
Outgoing Time (17:45 ~ 19:00)						
Proposed GNP	27.39	1.109	56.22	1.237	8.530	1.529
Basic GNP	28.58	1.127	56.09	1.283	11.32	1.328
AT	28.59	1.179	57.86	1.430	11.79	1.330
EC	28.54	1.163	57.83	1.284	11.38	1.326
THV	36.39	2.701	66.18	1.730	18.05	2.096
Overall Time (8:00 ~ 19:00)						
Proposed GNP	24.68	1.023	63.29	0.501	8.751	1.037
Basic GNP	24.89	1.031	63.89	0.510	9.734	1.031
AT	26.60	1.326	65.05	0.532	11.41	1.084
EC	25.66	1.198	64.28	0.524	10.39	1.023
THV	29.55	1.707	67.30	0.603	13.68	1.163

3.5.3 獲得した群管理ルールの検討

続いて、提案モデルの GNP のノード使用頻度およびノード遷移系列から、GNP が獲得した群管理ルールの検討を行う。Fig. 3.5 に、6 種類の GNP (*Case 1* ~ *Case 6*) のテストシミュレーション 1 試行のノード使用頻度を示す。Fig. 3.5 中の () 内の時間帯は Table 3.5 に $\langle \rangle$ で表示した時間帯に対応する。例えば (a) *Case 1* は 9 : 15 ~ 11 : 45 における *Case 1* の GNP の実行によるノード使用頻度を表す。したがって時間帯や交通条件により GNP の起動回数が異なるため、ノード使用頻度に差が生じている。図中の横軸はノードラベルを表す。

Fig. 3.5 より、各 *Case* の GNP はエレベータ選定処理ノードおよびエレベータ情報判定ノードによって各種の評価項目を活用した群管理を実行していることがわかる。選定処理ノードの場合は評価項目がその後のノード遷移に引数として反映されるため群管理ルールにおいて特に重要な意味を持つが、判定ノードにおける評価項目は処理ノードで使用された評価項目の妥当性を判断するための補足的な意味を持つ。交通流別に見ると、上下方向の乗客の流れが同程度の (a) *Case 1* , (b) *Case 2* では、多様な乗客の流れが存在するため 5 種類の評価項目を満遍なく活用している。一方で乗客の流れに偏りがある (c) *Case 3* ~ (f) *Case 6* ではノード使用にやや偏りが見られ、*SI* , *MS* を比較的多く使用している。これは (c) , (d) の出勤時では乗客が集中する基準階へのサービスが重要となるため、基準階呼びの待ち時間が長くないよう全てのホール呼びへのサービス時間を考慮した *MS* が活用されたものと考えられる。(e) , (f) の退勤時では逆に一般階へのサービスが重要となり、全ホール呼びへのサービス時間を考慮した *MS* と一般階低階層で発生する乗客の積み残しを考慮した *SI* が活用されたと見られる。また交通量で比較すると、交通量が厳しい (b) , (d) , (f) に比べ交通量が緩やかな (a) , (c) , (e) のほうが使用するノードの種類が多くなっている。これは後者の場合特殊な状況（出勤時の一般階呼びや退勤時の基準階呼び）が群管理性能に与える影響が大きくなるため、それらに柔軟に対応するべく多様なノードを使用しているものと推察される。

Table 3.7 に各 *Case* の GNP が獲得した代表的なノード遷移系列を示す。Table 3.7 は使用したノードのラベルと \square 内に示す判定結果をノード遷移順に記述したものである。判



P1: $A(X_n)$,
P2: $S(AT_1)$, **P3:** $S(AT_2)$, **P4:** $S(PN_1)$, **P5:** $S(PN_2)$, **P6:** $S(CN_1)$, **P7:** $S(CN_2)$,
P8: $S(SI_1)$, **P9:** $S(SI_2)$, **P10:** $S(MS_1)$, **P11:** $S(MS_2)$
J1: $J^{AT}(X_n)$, **J2:** $J^{PN}(X_n)$, **J3:** $J^{CN}(X_n)$, **J4:** $J^{SI}(X_n)$, **J5:** $J^{MS}(X_n)$,
J6: JEC , **J7:** JFD

Fig. 3.5 Node Usage Frequency of Proposed GNP in Each Case

Table 3.7 Obtained Node Transition Sequences of Proposed GNP in Each Case

Case	Obtained Node Transition Sequences	
1	1	$JEC[Away] \rightarrow JFD[GHD] \rightarrow S(MS_1) \rightarrow J^{CN}(X_n)[\leq 0.2] \rightarrow A(X_n)$
	2	$JEC[Away] \rightarrow JFD[GLD] \rightarrow S(MS_2) \rightarrow J^{SI}(X_n)[\leq 0.7] \rightarrow J^{MS}(X_n)[\leq 0.45] \rightarrow J^{PN}(X_n)[\leq 0.55] \rightarrow A(X_n)$
	3	$JEC[Close] \rightarrow JFD[GLD] \rightarrow S(AT_2) \rightarrow J^{PN}(X_n)[\leq 0.55] \rightarrow J^{SI}(X_n)[\leq 0.25] \rightarrow A(X_n)$
2	1	$JEC[Close] \rightarrow JFD[GLD] \rightarrow S(AT_2) \rightarrow J^{CN}(X_n)[\leq 0.75] \rightarrow A(X_n)$
	2	$JEC[Away] \rightarrow JFD[GLD] \rightarrow S(SI_1) \rightarrow J^{AT}(X_n)[\leq 0.55] \rightarrow J^{PN}(X_n)[\leq 0.95] \rightarrow J^{AT}(X_n)[\leq 0.4] \rightarrow A(X_n)$
	3	$JEC[Away] \rightarrow JFD[B] \rightarrow S(AT_1) \rightarrow J^{CN}(X_n)[\leq 0.65] \rightarrow J^{PN}(X_n)[\leq 0.95] \rightarrow J^{AT}(X_n)[\leq 0.4] \rightarrow A(X_n)$
3	1	$JEC[Away] \rightarrow JFD[B] \rightarrow S(SI_1) \rightarrow J^{SI}(X_n)[> 0] \rightarrow S(MS_1) \rightarrow J^{MS}(X_n)[\leq 0.75] \rightarrow J^{CN}(X_n)[\leq 0.45] \rightarrow A(X_n)$
	2	$JEC[Close] \rightarrow JFD[GLD] \rightarrow S(MS_2) \rightarrow J^{AT}(X_n)[\leq 0.55] \rightarrow J^{MS}(X_n)[\leq 0.75] \rightarrow J^{CN}(X_n)[\leq 0.45] \rightarrow A(X_n)$
	3	$JEC[Close] \rightarrow JFD[GHU] \rightarrow S(MS_1) \rightarrow J^{AT}(X_n)[\leq 0.75] \rightarrow A(X_n)$
4	1	$JEC[Close] \rightarrow JFD[GLD] \rightarrow S(MS_2) \rightarrow J^{SI}(X_n)[\leq 0.95] \rightarrow A(X_n)$
	2	$JEC[Away] \rightarrow JFD[GHD] \rightarrow S(AT_2) \rightarrow J^{AT}(X_n)[\leq 0.45] \rightarrow J^{SI}(X_n)[\leq 0.95] \rightarrow A(X_n)$
	3	$JEC[Close] \rightarrow JFD[B] \rightarrow S(CN_1) \rightarrow J^{AT}(X_n)[\leq 0.75] \rightarrow J^{CN}(X_n)[\leq 0.45] \rightarrow A(X_n)$
5	1	$JEC[Away] \rightarrow JFD[GLD] \rightarrow S(SI_1) \rightarrow J^{MS}(X_n)[\leq 0.65] \rightarrow J^{SI}(X_n)[\leq 0.3] \rightarrow A(X_n)$
	2	$JEC[Away] \rightarrow JFD[GLU] \rightarrow S(MS_1) \rightarrow J^{CN}(X_n)[\leq 0.1] \rightarrow J^{MS}(X_n)[\leq 1] \rightarrow A(X_n)$
	3	$JEC[Away] \rightarrow JFD[GLD] \rightarrow S(SI_1) \rightarrow J^{MS}(X_n)[> 0.65] \rightarrow S(MS_2) \rightarrow J^{SI}(X_n)[\leq 0.3] \rightarrow A(X_n)$
6	1	$JEC[Close] \rightarrow JFD[GLD] \rightarrow S(MS_1) \rightarrow J^{MS}(X_n)[\leq 0.95] \rightarrow A(X_n)$
	2	$JEC[Away] \rightarrow JFD[GLD] \rightarrow S(MS_2) \rightarrow J^{SI}(X_n)[\leq 0.45] \rightarrow J^{AT}(X_n)[\leq 0.4] \rightarrow J^{PN}(X_n)[\leq 0.75] \rightarrow A(X_n)$
	3	$JEC[Away] \rightarrow JFD[GHD] \rightarrow S(MS_1) \rightarrow J^{PN}(X_n)[\leq 0.8] \rightarrow A(X_n)$

定ノード $J^Y(X_n)$, JEC , JFD の判定結果はそれぞれ $\{\leq r_j^Y, > r_j^Y\}$, $\{\text{Close}, \text{Away}\}$, $\{\text{B}, \text{GLD}, \text{GLU}, \text{GHD}, \text{GHU}\}$ と表記する．したがって Table 3.7 から抽出される群管理ルールは *Case* 1-1 を例にとると“発生呼びに最も早く到着可能な上位2台のエレベータ間距離が離れていて，呼びが一般階高階層/下方向のとき，エレベータ MS_1 の CN の正規化変数が0.2以下ならば呼びを割当てる”となる．

Table 3.7 を見ると，各 *Case* とも Fig. 3.5 のノード使用状況を反映した群管理ルールが生成されている．ノード遷移系列は $JEC \rightarrow JFD \rightarrow S(X_n) \rightarrow J^Y(X_n) \rightarrow A(X_n)$ となるものが一般的であるが，*Case* 3-1 のように $J^Y(X_n)$ から $S(X_n)$ へいったん戻り，エレベータの再選定を行うルールも見られる．また *Case* 2-2 のように同種の判定ノード $J^{AT}(X_n)$ を2回使用し， $J^{AT}(X_n)(\leq 0.55)$ および $J^{PN}(X_n)(\leq 0.95)$ を満たすエレベータを $J^{AT}(X_n)(\leq 0.4)$ で判定するような，選定エレベータに対するきめ細かな判定を行うルールも見られる．以上の議論から，提案モデルのGNPは多様な交通条件に対して各種評価項目を活用した柔軟で一般性に富む群管理ルールを獲得できていると結論できる．

3.6 まとめ

本章では、第2章で検討した基本モデルの問題点を解決するため、ランキング処理およびノード関数最適化を考慮した拡張モデルを提案した。シミュレーションではビルの1日の交通流を想定した交通データを用い、提案モデルの性能評価を行った。

その結果、提案モデルの GNP は6種類の交通条件に対してそれぞれ柔軟な進化を行い、テストシミュレーションではそれらを適切に使い分けることで基本モデルおよび固有のルールによる従来手法よりも性能が改善されることを明らかにした。さらに獲得された群管理ルールの詳細な検討を行い、新規アルゴリズムによる一般性の高いルールの生成を確認した。

また付録Dで議論されているように、交通流の動的変化に伴う GNP の適切な切替えは重要である。本章では切替えのタイミングを予め指定したが、これはシステム自身によって行われることが望ましい。次章第4章では、交通流の変化を検出し最適な GNP への切替えを行う切替え制御型群管理システムの GNP による構成について検討する。

第4章

重要度指標付き GNP による切替え制御型群管理システム

4.1 序

第3章では、ランキング処理とノード関数最適化を導入した GNP による EGSCS の拡張モデルを提案し、複数の交通条件に対して進化させた GNP の使い分けによって動的な交通データに対する効率的な群管理を実現した。しかし実用化を前提とする場合、この呼び割当ての切替え機能はシステム自身が備えていることが望ましい。実際、EGSCS の機能には呼び割当て機能と交通流検出機能があり、交通流検出機能が交通流の変化を検知し、最適な割当て機能を選択する群管理システムの開発が行われている [17]。AI 技術による研究例を見ても、文献 [41] のファジィ理論 (FL) による群管理システムでは、FL によって現在の交通流を特定し、最適な割当て機能 (FL による割当て) への切替えを行う方式が提案されている。

本章では以上をふまえ、交通流の検出と切替えを行う機能の GNP による構成手法として、重要度指標付き GNP を導入した切替え制御型群管理システムを提案する。重要度指標付き GNP は人間の脳の機能局在機能を規範とするモデルであり、提案方式では動的な交通流に対する複数の割当て機能の切替えによる群管理を機能局在化問題と見なすことができる。

以下、提案方式の構成、重要度指標付き GNP による切替えアルゴリズムについて述べ、シミュレーションでは第3章と同じ交通データを使用し、提案方式の切替え制御型群管理の有効性について検討する。

4.2 基本構成

Fig.4.1 に提案する GNP による切替え制御型群管理システムの概要を示す．システムはエレベータシステム，GNP コントローラ，情報管理部より構成される．GNP コントローラは呼びの割当て機能を司る割当て GNP と交通流の検出，切替え機能を司る切替え GNP に分類される．割当て GNP は発生呼びを最適なエレベータへ割当てて機能を有し，特定の交通条件に対して最適化されたものが複数個存在する．切替え GNP は現在の交通流を検出し，それに最適な割当て GNP へ割当て機能を切替える．情報管理部は割当て情報と切替え情報に分類され，それぞれで管理される情報が割当て GNP および切替え GNP へ入力される．またこれらはそれぞれの GNP の起動時（割当て GNP：呼び発生時，切替え GNP：5 分毎）に最新の情報に更新される．上記構成により，提案方式は切替え制御型群管理を実現する．

4.3 割当て GNP

割当て GNP の構成は，第3章で提案した拡張モデルの GNP を使用する．また進化のシミュレーション条件も，Table 3.2，Table 3.3 にそれぞれ従う．よって提案方式の割当て GNP は6種類となり，これらを切替え GNP によって適応的に切替えて群管理を行う．

4.4 切替え GNP

本節では，重要度指標付き GNP による切替え GNP の構成について述べる．切替え GNP では判定ノードで入力情報を判定する際，判定結果毎に設定される重要度指標（Importance Index）を加算する．そして処理ノードへ至る判定ノードの遷移で累積された重要度指標の情報をもとに現在の交通流を特定し，最適な割当て GNP へ割当て機能を切替える．

以下，重要度指標による切替えアルゴリズム，ノード関数および適合度関数について述べる．

4.4.1 重要度指標による切替えアルゴリズム

付録 E の判定ノードの判定変数，処理および重要度指標を，切替え GNP の l 番目の判定結果 J_l ($l \in L$)，割当て GNP m ($\in M, |M| = 6$) への切替え処理 P_m ， $IMX(J_l, P_m)$ ($\in [0, 1]$) とする．そして Fig. 4.2 のように，処理強度 $S(P_m)$ を Eq. (4.1) により求める．

$$S(P_m) = \sum_{l \in L'} IMX(J_l, P_m) \quad (4.1)$$

ここで，

- L' : 処理ノードに至る切替え GNP の判定ノードの判定結果の添字の集合
- L : 切替え GNP の判定ノードの判定結果の添字の集合
- M : 割当て GNP の添字の集合

そして，接続する処理ノードにおいて Eq. (4.2) により切替え処理 P_α を決定する．

$$P_\alpha = \arg \max_{m \in M} S(P_m) \quad (4.2)$$

すなわち， $S(P_m)$ の最大値をとる割当て GNP α が現在の交通流に最適な割当て GNP となる．Fig. 4.2 の例では割当て GNP 3 への切替えとなる．重要度指標 $IMX(J_l, P_m)$ は予め適切な値に設定し，切替え GNP の進化を通じて固定とする．

4.4.2 ノード関数

切替え GNP のノード関数として，以下の 5 種類を用意した．

処理ノード（1 種類）

- 割当て GNP α へ割当て機能を切替える

判定ノード（4 種類）

- (1) 5 分間に各階床に出現した上方向へ移動する乗客と下方向へ移動する乗客の比 UD を判定 { 判定結果: J_1, J_2, J_3 }
- (2) 5 分間に各階床に出現した乗客数 NP [person] を判定 { 判定結果: J_4, J_5 }
- (3) 最後の割当て GNP の切替え実行後の経過時間 ST [min] を判定 { 判定結果: J_6, J_7, J_8 }
- (4) 現在使用している割当て GNP を判定 { 判定結果: $J_9, J_{10}, J_{11}, J_{12}, J_{13}, J_{14}$ }

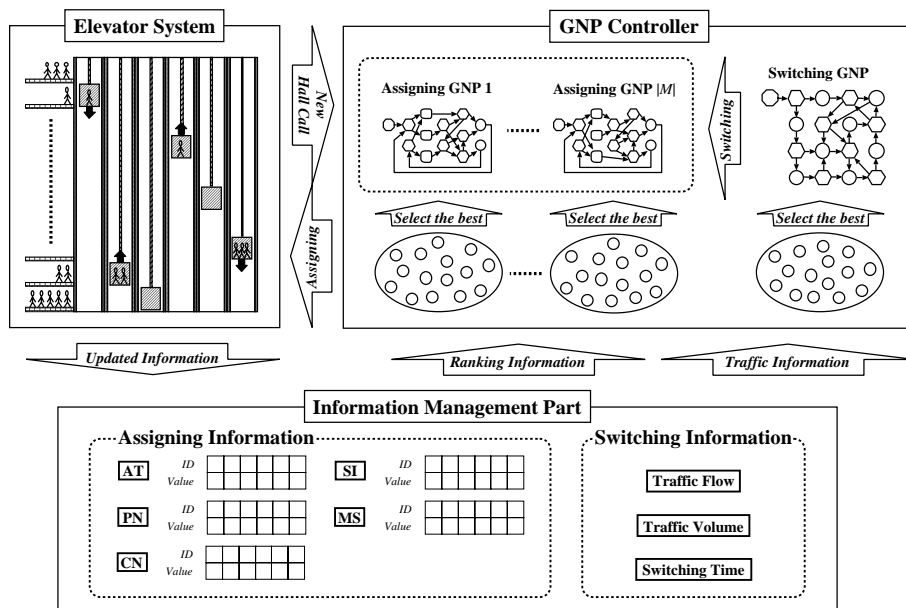


Fig. 4.1 Structure of Proposed Method

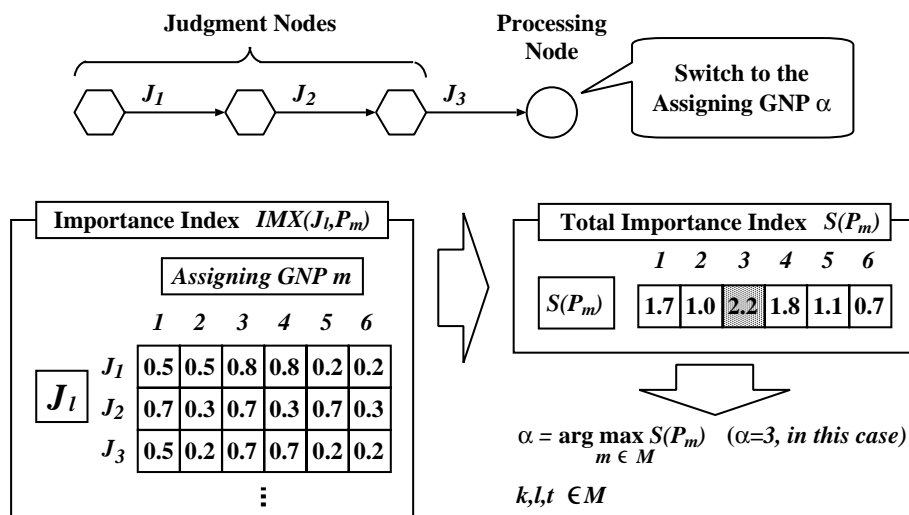


Fig. 4.2 Mechanisms of Switching

ここで，各判定結果 J_l は以下のように表される．

J_1 : $0 < UD \leq 0.5$	J_6 : $ST \leq 15$	J_9 : <i>Assigning GNP 1</i>
J_2 : $0.5 < UD \leq 2$	J_7 : $15 < ST \leq 30$	J_{10} : <i>Assigning GNP 2</i>
J_3 : $2 < UD$	J_8 : $30 < ST$	J_{11} : <i>Assigning GNP 3</i>
J_4 : $0 < NP \leq 150$		J_{12} : <i>Assigning GNP 4</i>
J_5 : $150 < NP$		J_{13} : <i>Assigning GNP 5</i>
		J_{14} : <i>Assigning GNP 6</i>

4.4.3 適合度関数

切替え GNP の適合度関数 *Fitness* は平均 2 乗待ち時間，2 乗最大待ち時間，（単位時間当たりループ回数 + 不適切な切替え回数）の 2 乗を考慮し Eq. (4.3) のように定義した．

$$Fitness = \frac{1}{N} \sum_{n=1}^N (t_n)^2 + w_t \cdot (t_{max})^2 + w_{ls} \cdot (l + s)^2 \quad (4.3)$$

ここで，

N	: 全乗客数
t_n	: n 番目の乗客の待ち時間
t_{max}	: 乗客の最大待ち時間
l	: 1 時間当たりのループ回数
s	: 不適切な切替え回数
w_t, w_{ls}	: 各項の重み係数．

不適切な切替えとは Table 4.1 のように，アップピーク時 → ダウンピーク時の切替えといった通常の交通流の変化から考えて明らかに間違いと思われる切替えを指す．

4.5 切替え GNP の進化のシミュレーション

本節では，第 3 章 Table 3.1～Table 3.3 実行条件に対して進化させた割当て GNP を使用した，切替え GNP の進化のシミュレーションを行う．

4.5.1 実行条件

Table 4.2 に，切替え GNP の進化の実行条件を示す．切替え GNP の評価は，Fig.4.3 に示す 6 種類の動的に変化する交通データに対して実行する．評価の際は汎化能力の向上

Table 4.1 Examples of Inappropriate Switching

Current Assigning GNP		Next Assigning GNP
<i>Assigning GNP 1</i>	→	<i>Assigning GNP 2</i> or <i>Assigning GNP 3</i> or <i>Assigning GNP 4</i>
<i>Assigning GNP 2</i>	→	<i>Assigning GNP 1</i> or <i>Assigning GNP 3</i> or <i>Assigning GNP 5</i> or <i>Assigning GNP 6</i>
<i>Assigning GNP 3</i>	→	<i>Assigning GNP 1</i> or <i>Assigning GNP 2</i> or <i>Assigning GNP 5</i> or <i>Assigning GNP 6</i>
<i>Assigning GNP 4</i>	→	<i>Assigning GNP 2</i> or <i>Assigning GNP 5</i> or <i>Assigning GNP 6</i>
<i>Assigning GNP 5</i>	→	<i>Assigning GNP 1</i> or <i>Assigning GNP 2</i> or <i>Assigning GNP 3</i> or <i>Assigning GNP 4</i>
<i>Assigning GNP 6</i>	→	<i>Assigning GNP 1</i> or <i>Assigning GNP 3</i> or <i>Assigning GNP 4</i>

Table 4.2 Evolutional Conditions of Switching GNP

Items	Value
Generation	300
Population Size	300
Crossover	120
Mutation	170
Elite Preservation	10
Node Size	25 (5/seed)+Initial Node
Crossover Probability P_c	0.1
Mutation Probability P_m	0.01
Evaluation Time [h]	6 (1 [h] \times 6 Traffic Data)
w_t, w_{ls}	0.007, 60

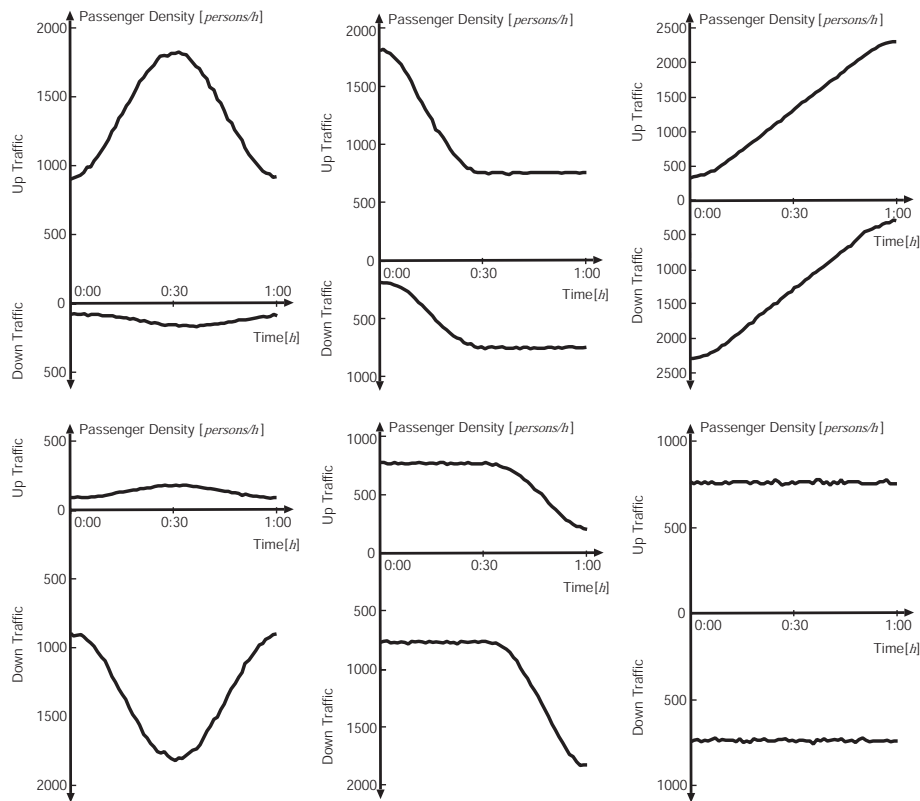


Fig. 4.3 Traffic Data for Evolving Switching GNP

を考慮し、6 種類の交通データをランダムな順序で適用する。Fig.4.3 は時間（横軸）に対する上下方向の乗客密度（縦軸）を示しており、進化のシミュレーションは6 種類の特徴的な交通データに対する適切な切替えルールの獲得を狙いとする。シミュレーションは5 種類の乱数系列に対して実行した。

4.5.2 進化の様子

Fig.4.4 に切替え GNP の最良個体の適合度曲線を示す。結果は5 種類の乱数系列による実行結果の平均である。

Fig.4.4 より、適合度が世代経過と共に改善している様子がわかる。しかし適合度の振幅は Fig. 3.3 よりも大きい。これは交通データをランダムな順番で評価することによるものと考えられる。また、適合度の改善の幅も割当て GNP の場合に比べて小さくなっている。これは、既に最適化された割当て GNP を使用して進化を実行するため、切替え GNP の最適化が進んでいない段階でもある程度の群管理性能が保証されるためと考えられる。

4.6 テストシミュレーションによる性能評価

本節では Fig. 3.4 と同じビルの1 日の交通データを用いたテストシミュレーションを実行し、切替えによる群管理の性能評価および切替え GNP による実際の切替えの様子について検討する。

4.6.1 実行条件

テストシミュレーションの実行条件は全て第3 章と同じとする。提案方式の性能評価は、Table 3.5 による割当て GNP の手動切替え方式および、1 個の GNP (Case 4) のみを使用する非切替え方式の2 種類の方式との比較により行う。シミュレーションは乱数系列の異なる 1000 試行を実行した。

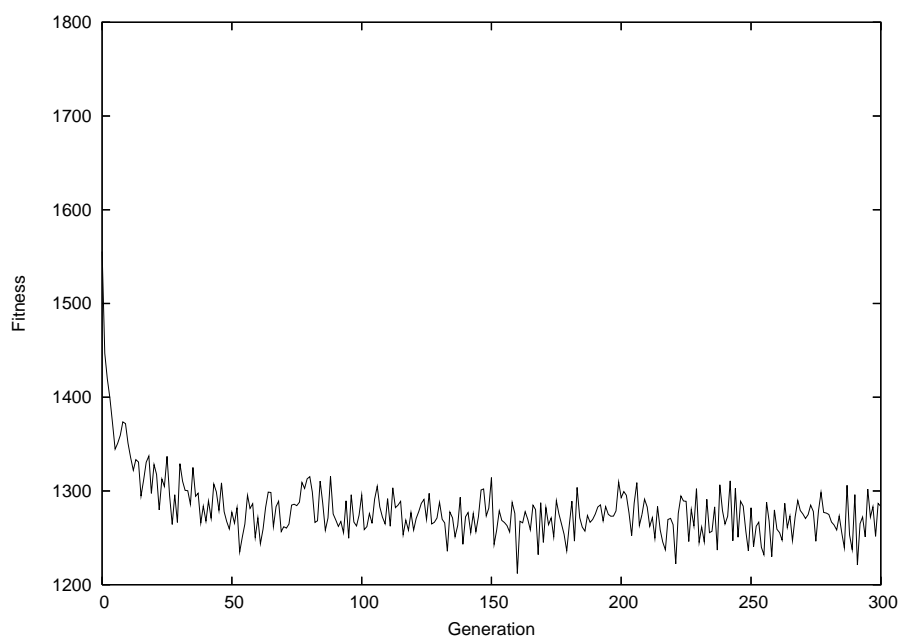


Fig. 4.4 Fitness Curve of Switching GNP

4.6.2 従来手法との性能比較

Table 4.3 に 4 種類の時間帯および全時間帯の上記 3 手法の性能比較を、Table 3.6 と同様平均待ち時間 (AWT [s])、平均乗車時間 (ATT [s])、60 [s] 以上の長待ち乗客率 (LWP [%]) について示す。Table 4.3 中の GNP_{AS} は提案方式、 GNP_{MS} は手動切替え方式、 GNP_{NS} は非切替え方式をそれぞれ示す。また結果はテストシミュレーション 1000 試行に対する各指標の平均と標準偏差を示しており、太字の数値は提案方式が最も良いケースを示す。

Table 4.3 を見ると、提案方式は非切替え方式に対し出勤時以外の各時間帯で性能が改善している。これは非切替え方式ではアップピーク時の交通条件に適合した割当て GNP のみを使用しており、それ以外の交通流が主となる時間帯に対する適切な群管理が困難となるが、提案方式では交通流の変化に追従し割当て GNP を切替える群管理が実行されるためである。また手動切替え方式に対しては、共通の割当て GNP を使用しているが提案方式の性能は多少改善している。これは設計者の設定による切替えよりも適切な切替えルールが提案方式によって獲得可能であることを示している。

4.6.3 切替えの様子

Fig. 4.5 に切替え GNP が実際に行った切替えの様子を示す。Fig. 4.5 は Fig. 3.4 上にその時間帯に実際に使用した割当て GNP を、上下方向の乗客密度が一致するように図示したものである。

Fig. 4.5 の切替えの様子を見ると、各時間帯で概ね適切な割当て GNP が使用されており、Table 4.3 の結果と併せ、切替え GNP が進化によって適切な切替えルールを獲得できていることがわかる。また昼食時 (13:00 頃) や退勤時 (18:30 頃) では一部で実際の交通流に対し最適ではないと思われる割当て GNP を使用しているが、これは実際に使用された割当て GNP でも効率的な群管理が可能であると切替え GNP が判断したためと考えられる。

Table 4.3 Comparison of Performance among Proposed Method and Other Methods

Method	AWT [<i>s</i>]		ATT [<i>s</i>]		LWP [%]	
	Ave.	S.D.	Ave.	S.D.	Ave.	S.D.
Incoming Time (8:00 ~ 9:15)						
<i>GNP_{AS}</i>	19.93	2.919	77.30	1.529	6.298	2.975
<i>GNP_{MS}</i>	19.86	2.898	77.16	1.567	6.438	2.964
<i>GNP_{NS}</i>	19.93	2.800	77.14	1.504	6.433	2.850
Business Time (9:15 ~ 11:45, 13:15 ~ 17:45)						
<i>GNP_{AS}</i>	20.71	0.351	58.48	0.625	4.649	0.386
<i>GNP_{MS}</i>	20.85	0.379	57.67	0.629	4.744	0.408
<i>GNP_{NS}</i>	23.63	0.345	57.16	0.611	8.148	0.426
Lunch Time (11:45 ~ 13:15)						
<i>GNP_{AS}</i>	37.86	4.437	76.69	1.338	22.74	4.453
<i>GNP_{MS}</i>	38.01	4.904	77.04	1.245	22.90	4.885
<i>GNP_{NS}</i>	40.60	4.441	75.45	1.307	24.85	4.373
Outgoing Time (17:45 ~ 19:00)						
<i>GNP_{AS}</i>	26.94	1.045	56.44	1.175	7.924	1.429
<i>GNP_{MS}</i>	27.39	1.109	56.22	1.237	8.530	1.529
<i>GNP_{NS}</i>	30.75	1.162	54.61	1.156	12.85	1.513
Overall Time (8:00 ~ 19:00)						
<i>GNP_{AS}</i>	24.54	0.935	63.71	0.500	8.579	0.972
<i>GNP_{MS}</i>	24.68	1.023	63.29	0.501	8.751	1.037
<i>GNP_{NS}</i>	27.19	0.941	62.51	0.501	11.61	0.951

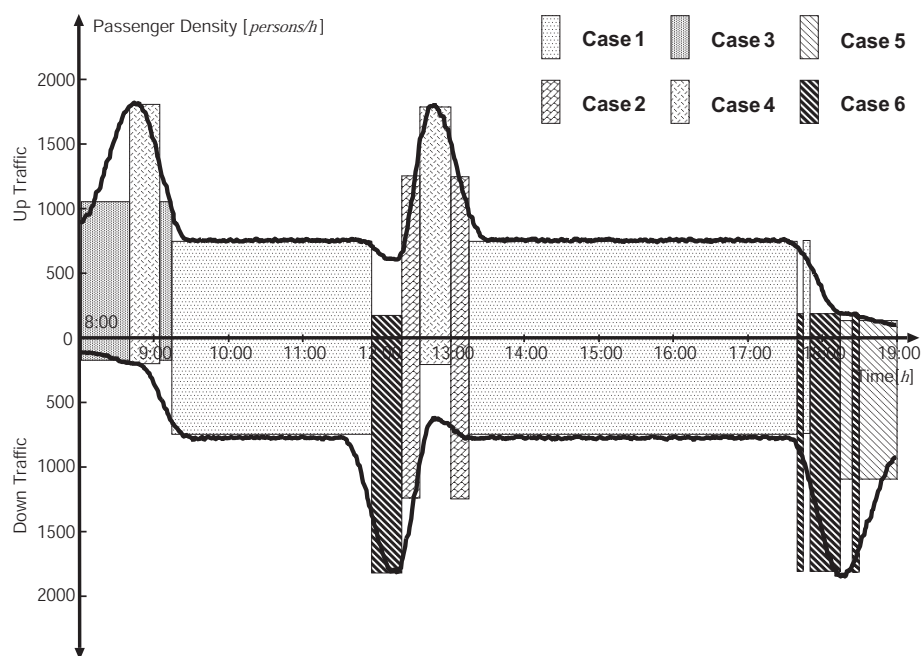


Fig. 4.5 Switching History by Switching GNP

4.7 まとめ

本章では、第3章および付録Dで議論した動的な交通流に対する割当て機能の切替えによる群管理について、重要度指標付き GNP を導入した切替え制御型群管理システムを提案し、第3章と同じ交通データを用いたシミュレーションにより性能評価を行った。シミュレーションの結果から、提案方式の切替え GNP が交通流の変化に対し適応的に割当て GNP を切替えるルールを獲得でき、それによって効率的な群管理が実行されることを確認した。今後は学習する交通流のパターンおよび重要度指標の最適化、異なるテストデータに対する性能評価等の検討を行い、実用化を視野に入れた拡張を期待できる。

以上の第2章から第4章までの検討を通じて、本研究では GNP による EGSCS の基本的枠組みを確立し、その有効性を明らかにした。したがって GNP の次世代型エレベータシステムへの展開を考える場合、これまでの検討結果をふまえた上での効果的な群管理方式の構築を期待できる。次章第5章では、次世代型エレベータシステムのひとつであるダブルデッキエレベータシステム (DDES) への GNP の導入を行い、その基本特性について検討する。

第5章

GNPによるダブルデッキエレベータ群管理システムの基礎検討

5.1 序

本章では第2章から第4章までの検討結果をふまえ，GNPによるダブルデッキエレベータ群管理システムの提案とその基礎検討を行う．

近年，高層化が進むビル内の効率的な人員輸送のニーズに対し，様々な次世代型エレベータシステム [18, 19] の開発が進められている．そのひとつであるダブルデッキエレベータシステム (DDES) [19, 48, 49, 50, 61] は付録 G で説明しているように，エレベータ（以降，かごと表記）を2台垂直に連結し運行することで輸送能力の向上およびビル内エレベータ占有面積の削減を図ることができる．しかしそれに伴い従来型シングルデッキエレベータシステム (SDES) には見られない特有の挙動が存在し，群管理目標の達成にはそれらを効率的に扱うことが必要となる．また特に，セミダブル運転時 [19] には階床の区別なくエレベータがサービスできるため，DDES の挙動は一層複雑化する．

このように DDES の群管理は SDES より困難とされ，これまで以上に AI 技術による効果的な群管理システムの必要性が叫ばれている．AI 技術による DDES の群管理システムはいくつかの研究例が報告されているが [49, 62]，特にセミダブル運転時の群管理に関しては開発途上であり，実用化には至っていない．本章では GNP による DDES の群管理方式の確立を目的として，セミダブル運転時の群管理に焦点を絞り検討を行う．また同運転への効果が期待できる，行先階登録方式エレベータシステム (DFGES) [19, 46, 63, 64] を導入した DDES を前提として議論を進める．

以下，提案する GNP によるダブルデッキエレベータ群管理システムの概要，そしてシミュレーションによる検討結果について述べる．

5.2 GNP によるダブルデッキエレベータ群管理システム

GNP による DDES の群管理システムの提案モデルは，これまでに検討してきたモデルをベースとして，DDES 指向型に拡張したものである．本検討では動的な交通流に対する切替え制御は考慮せず，呼びの割当てによる群管理に焦点を絞る．システムで特に重要となるものは，DDES の挙動を考慮したかご評価項目の定義である．またかごの選定に関して，単一の評価項目を基準とした選定から，GA のかご評価式の最適化による，複数の評価項目を基準とした柔軟なかご選定を行うメカニズムを導入している．

以下，提案モデルの概要およびアルゴリズムについて述べる．

5.2.1 基本構成

システムの基本構成を Fig. 5.1 に示す．システムは DDES，GNP コントローラ，情報管理部からなる．これまでのモデルと同様，DDES からの呼びの発生を受け情報管理部では各種かご評価項目およびエレベータシステムに関する制御情報を計算，管理する．各情報は GNP コントローラへ入力され，GNP の実行を経てコントローラは割当て情報を出力し，群管理を行う．ここで割当て情報とは，エレベータシャフト番号とかごの種類（上かご／下かご）によって特定されるかご ID を指す．

5.2.2 評価項目

提案モデルで使用するかご評価項目は，大別して SDES，DDES 共通の評価項目および DDES 特有の挙動を考慮した評価項目がある．以下に，定義した 12 種類の評価項目の説明およびその導出法について述べる．

【SDES，DDES 共通の評価項目（6 種類）】

(1) *AIWT*

DFGES における待ち時間短縮を目的として考案された指標であり [17]，発生呼びの登録による，それを含む登録ホール呼びへの（到着予想時間＋登録後の経過時間）の増加分の合計として Eq. (5.1) により計算される．

$$AIWT(i) = \sum_{h \in H'} (apt_h(i) + et_h(i)) - \sum_{h \in H} (apt_h(i) + et_h(i)) \quad (5.1)$$

ここで,

- $apt_h(i)$: かが i ($\in I$) の登録ホール呼び h に対する到着予想時間
- $et_h(i)$: かが i のホール呼び h 登録後の経過時間
- I : かが ID の集合
- H : 発生ホール呼びを含まない登録ホール呼びの添字の集合
- H' : 発生ホール呼びを含む登録ホール呼びの添字の集合.

(2) MST

発生呼びを含む登録ホール呼びへの (到着予想時間 + 登録後の経過時間) の最大値として Eq. (5.2) により計算され, 最大待ち時間の短縮に寄与する.

$$MST(i) = \max_{h \in H'} (apt_h(i) + et_h(i)) \quad (5.2)$$

(3) PN

かが i に現在乗車中もしくは乗車予定の乗客数の合計として計算され, かごの運行負荷に関わる指標として最大待ち時間および平均乗車時間の短縮に寄与する.

(4) CN

かが i に現在登録されているホール呼び数と行先階呼び数の合計として Eq. (5.3) により計算される. これは PN と同様にかごの運行負荷に関わる指標である.

$$CN(i) = \sum_{j \in F} \sum_{k \in F} (hcn_{jk}(i) + dcn_{jk}(i)) \quad (5.3)$$

ここで,

- $hcn_{jk}(i)$: かが i の j 階から k 階へ登録されているホール呼び数
- $dcn_{jk}(i)$: かが i の j 階から k 階へ登録されている行先階呼び数
- F : ビル階床の添字の集合.

(5) RR

発生呼びの登録による, それを含む登録ホール呼びへの到着時の予想乗車率の増加

分の合計として Eq. (5.4) により計算される．これにより到着時の乗客の積み残しを解消でき，最大待ち時間の短縮につながる．

$$RR(i) = \sum_{h \in H'} rr_h(i) - \sum_{h \in H} rr_h(i) \quad (5.4)$$

ここで，

$rr_h(i)$: かが i のホール呼び h への到着時の予想乗車率（乗客数 / かが定員数）．

(6) MCN

発生ホール呼びおよび行先階呼びの登録により，既登録ホール呼び，行先階呼びとの同時サービスが可能かを判別する．同時サービスによりかごの停止回数の減少を図ることができ，待ち時間や移動時間の短縮につながる．

【DDES 特有の挙動を考慮した評価項目（6 種類）】

以下の評価項目は DDES 特有の挙動の要因となる，連結するかご間の相互関係を考慮したものである．ここで，評価の対象となるかごを自かご，それに連結するかごを他かごと呼ぶことにする．

(1) OIWT

自かごへの発生呼びの登録による，他かごの登録ホール呼びへの（到着予想時間 + 登録後の経過時間）の増加分の合計として Eq. (5.5) により計算される．

$$OIWT(i) = \sum_{o \in O} (oapt_o'(i) + oet_o'(i)) - \sum_{o \in O} (oapt_o(i) + oet_o(i)) \quad (5.5)$$

ここで，

$oapt_o(i)$: かが i の他かごの登録ホール呼び o に対する到着予想時間

$oet_o(i)$: かが i の他かごのホール呼び o 登録後の経過時間

$oapt_o'(i)$: かが i に発生呼びを登録後の他かごの登録ホール呼び o に対する到着予想時間

$oet_o'(i)$: かが i に発生呼びを登録後の他かごのホール呼び o 登録後の経過時間

O : かが i の他かごの登録ホール呼びの添字の集合．

(2) OST

自かごへの発生呼びの登録による，他かごの登録ホール呼びへの（到着予想時間 + 登録後の経過時間）の最大値として Eq. (5.6) により計算される．

$$OST(i) = \max_{o \in O} (oapt_o^l(i) + oet_o^l(i)) \quad (5.6)$$

(3) PD

自かごと他かごの乗客数の差分として Eq. (5.7) により計算される．これは連結するかご間の運行負荷のバランスを考慮した指標である．

$$PD(i) = PN(i) - OPN(i) + CAPACITY \quad (5.7)$$

ここで，

$OPN(i)$: かご i の他かごの乗客数
 $CAPACITY$: かご定員数 ($PD(i)$ が非負となるように考慮)．

(4) CD

自かごと他かごの登録呼び数の差分として Eq. (5.8) により計算される． PD と同様に運行負荷のバランスを考慮する．

$$CD(i) = CN(i) - OCN(i) + FLOOR \quad (5.8)$$

ここで，

$OCN(i)$: かご i の他かごの登録呼び数
 $FLOOR$: ビル階床数 ($CD(i)$ が非負となるように考慮)．

(5) SSN

発生ホール呼びおよび行先階呼びの自かごへの登録により，他かごの既登録ホール呼びおよび行先階呼びとの同時サービス（両かごサービス）が可能かを判別する．両かごサービスを実行することによって輸送効率の向上だけでなく，乗客の乗り心地の改善も達成できる．

(6) *OCD*

発生行先階呼びが他かごの既登録行先階呼びと重複しないかを判別する．すなわち，同一階床へ移動する乗客の分離乗車の可能性を調べることができる．

上記評価項目は付録 G で示した 5 種類の DDES 特有の挙動のうち，*OIWT*，*OST* は挙動 5，*PD*，*CD* は挙動 4，*SSN* は挙動 1，挙動 2，*OCD* は挙動 3 をそれぞれ考慮したものである．またホール呼びおよび行先階呼びの定義は付録 F による．

5.2.3 呼び割当てのメカニズム

提案モデルの GNP の基本構成を Fig. 5.2 に示す．GNP の構成はシステム情報判定部，かご選定処理部，かご情報判定部および割当て処理部に分類できる．以下に各部の処理内容について述べる．

【システム情報判定部】 GNP の起動後，ここにかご評価項目以外のエレベータシステムに関する情報の判定を行う．具体的には，かご間時間的距離の標準偏差 *ESD*，発生ホール呼び方向とその階床の種類 *HCF* そして発生行先階呼びの階床の種類 *DCF* を判定する．*ESD* はエレベータの運行間隔のばらつきを表す指標で，過去 5 分間の標準偏差の平均 *ESD_{Ave}* と比較してばらつきの状況を判定する．*HCF*，*DCF* は発生ホール呼びおよび行先階呼びの方向や階床を分類し判定する．

【かご選定処理部】 システム情報の判定後，ここで評価項目に関するかご選定処理ノードを使用したかご評価式によって特定のかごを選定する．具体例として，Fig. 5.2 のようにノード遷移が実行された場合，そのノード内容から Eq. (5.9) により各かごの評価式 $e(i)$ を計算する．

$$e(i) = \sum_{p \in P'} w_{X(p)} \cdot x_p(i) \quad (5.9)$$

ここで，

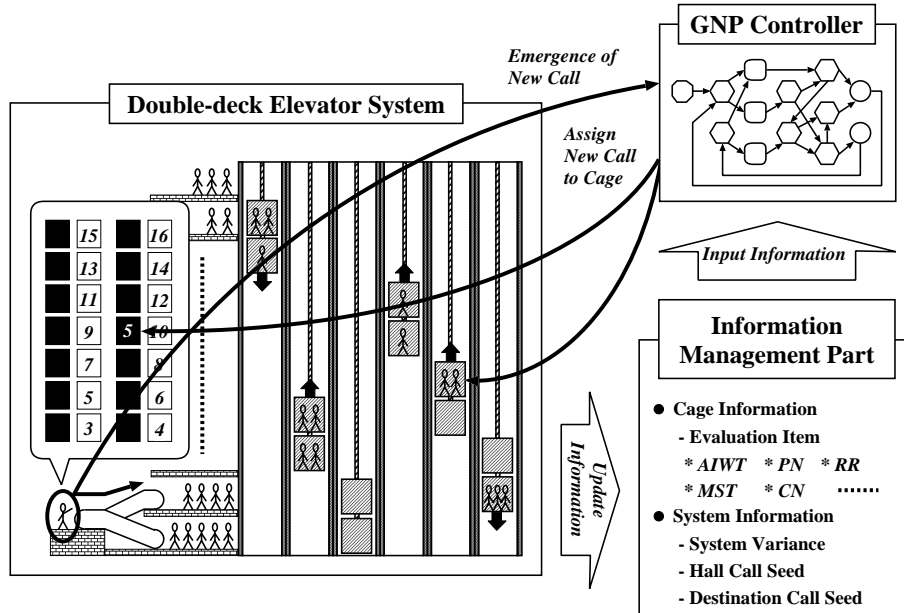


Fig. 5.1 Basic Structure of Proposed Model for DDES

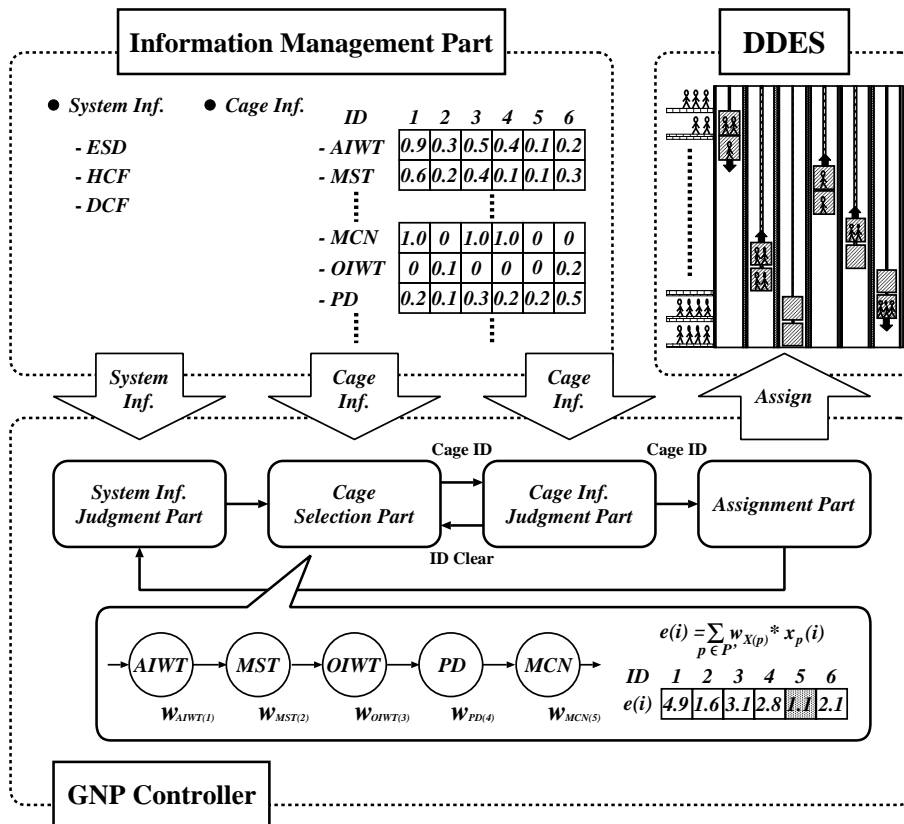


Fig. 5.2 Assigning Mechanisms of Proposed Model

68 第5章 GNP によるダブルデッキエレベータ群管理システムの基礎検討

- P' : かが情報判定部に至るかが選定処理ノードの添字の集合
 $w_{X(p)} \in [0, 1]$: かが選定処理ノード p の評価項目 X に関する重み付けノードパラメータ
 $x_p(i)$: かが i のかが選定処理ノード p が扱う評価項目 X に関する正規化変数

であり，正規化変数 $x(i)$ は次式 Eq. (5.10) により計算する．

$$x_p(i) = \frac{X_p(i)}{X_{AveMax}} \quad (5.10)$$

ここで，

- $X_p(i)$: かが i のかが選定処理ノード p が扱う評価項目 X に関する評価値
 X_{AveMax} : 各かごに対する評価項目 X の最大値の過去 5 分間の平均

である．ここで評価項目 $\{MCN, SSN, OCD\}$ 以外の $X_p(i)$ の値は前述の導出式により求まるが，これら 3 項目に関してはその状況を満たす場合は $x(i) = 0$ ，満たさなければ $x(i) = 1$ と計算する．そして，Eq. (5.11) により選定かご d を決定する．

$$d = \arg \min_{i \in I} e(i) \quad (5.11)$$

すなわち Fig. 5.2 の例では，5 号機のかごが選定される．

【かが情報判定部】 かが d を選定後，それをかが情報判定ノードの引数とし，選定かごの各評価項目に関する情報を判定する．かが情報判定ノードでは $\{MCN, SSN, OCD\}$ 以外の評価項目に関しては Eq. (5.12) の内容の判定を行う．

$$if \ y(d) \leq r_j^Y \quad (5.12)$$

ここで，

- $y(d)$: かが d の評価項目 Y ($\in \Gamma$) に関する正規化変数
 $r_j^Y \in [0, 1]$: 評価項目 Y に関するかが情報判定ノード j ($\in J$) の閾値ノードパラメータ
 J : かが情報判定ノードの添字の集合

であり， $y(i)$ も同様に次式 Eq. (5.13) により計算する．

$$y(i) = \frac{Y(i)}{Y_{AveMax}} \quad (5.13)$$

ここで,

- $Y(i)$: 各 i の評価項目 Y に関する評価値
 Y_{AveMax} : 各 i に対する評価項目 Y の最大値の過去 5 分間の平均

である．一方 $\{MCN, SSN, OCD\}$ に関しては, 各状況を満たすかどうかを判定する．上記判定の結果, 判定条件を満たす場合は割当て処理部への接続が可能であるが, 満たさない場合は i 割当て処理部へ戻り, 再度 i の選定を行う．

【割当て処理部】 ここでは引数 d を受けとり, 割当て処理ノードによって i への呼びの割当てを行う．割当て決定後は再びシステム情報判定部へ戻り, 次に発生する呼びに対し同様の処理を実行する．

上記割当てメカニズムでは i 割当て処理部において各評価項目の荷重和による i 選定のアルゴリズムを導入したため, 第 3 章の評価項目単位のランキング処理による選定よりも柔軟で効果的な i 選定が可能となる．またノードパラメータ $(w_{X(p)}, r_j^Y)$ は付録 C の最適化アルゴリズムにより GNP の進化と併行して最適化する．

5.2.4 ノード関数

提案モデルの GNP のノード関数として, 以下の 28 種類のノードを用意した．

処理ノード (13 種類)

- 割当て処理ノード (1 種類)
 - $A(d)$: i への発生呼びの割当て (d : 引数)
- i 割当て処理ノード (12 種類)
 - $S(X)$: 評価項目 $X_p(i)$ の正規化変数 $x_p(i)$ の重みパラメータ $w_{X(p)}$ との荷重和を計算

$$\begin{cases} X \in \Gamma \\ \Gamma = \{AIWT, MST, PN, CN, RR, MCN, OIWT, OST, PD, \\ CD, SSN, OCD\} \end{cases}$$

判定ノード (15 種類)

• かが情報判定ノード (12 種類)

– $J^Y(d) : y(d) \leq r_j^Y$ を判定 (d : 引数)

$$\begin{cases} Y \in \Gamma', j \in J \\ \Gamma' = \{AIWT, MST, PN, CN, RR, OIWT, OST, PD, CD\} \end{cases}$$

– $J^Z(d) : \text{かが } d \text{ が } Z \text{ の示す状況を満たすかを判定 } (d : \text{引数})$

$$\begin{cases} Z \in \Gamma'' \\ \Gamma'' = \{MCN, SSN, OCD\} \end{cases}$$

• システム情報判定ノード (3 種類)

- ESD : エレベータシステムのバラツキを判定
- HCF : 発生ホール呼びの階床と呼び方向を判定
- DCF : 発生行先階呼びの階床を判定

$A(d)$ は割当て処理部を構成し, 引数かが d に発生呼びを割当てる (有向枝 : 1 本) . $S(X)$ はかが選定処理部を構成し, 遷移により計算されたかが評価式 $e(i)$ から引数かが d を選定する (有向枝 : 1 本) . $J^Y(d)$, $J^Z(d)$ はかが情報判定部を構成し, 引数かが d の評価項目に関する 2 分岐判定を行う (有向枝 : 2 本) . ESD , HCF , DCF はシステム情報判定部を構成する . ESD は現在のかご間隔の標準偏差が過去 5 分間の標準偏差の平均 ESD_{Ave} 以下かを判定し有向枝は 2 本 , HCF はホール呼び発生階と呼び方向が { 基準階 , 一般階低階層 / 下方向 , 一般階低階層 / 上方向 , 一般階高階層 / 下方向 , 一般階高階層 / 上方向 } のいずれかを判定し有向枝は 5 本 , DCF は発生呼びの行先階が { 基準階 , 一般階低階層 , 一般階高階層 } のいずれかを判定し有向枝は 3 本である .

5.2.5 適合度関数

DDES の群管理では乗客の待ち時間の最小化のみならず, 乗客の乗り心地に関わる指標の最適化も考慮しなければならない . そこで従来の適合度関数に新たに片かがサービスとなった乗客の延べ人数の 2 乗を追加した Eq. (5.14) を適合度関数 $Fitness$ として定義した .

$$Fitness = \frac{1}{N} \sum_{n=1}^N (t_n)^2 + w_t \cdot (t_{max})^2 + w_{ocs} \cdot (ocs)^2 + w_l \cdot (l)^2 \quad (5.14)$$

ここで，

N	: 全乗客数
t_n	: n 番目の乗客の待ち時間
t_{max}	: 乗客の最大待ち時間
ocs	: 片かごサービスとなった乗客の延べ人数
l	: 1 時間当たりのループ回数
w_t, w_{ocs}, w_l	: 各項の重み係数．

5.3 シミュレーション

シミュレーションにより提案モデルの有効性を検討する．始めに GNP の進化による最適化を実行し，続いて最適化した GNP のテストシミュレーションによる性能評価を行う．性能評価ではまず固有のかご評価式に基づく従来手法との性能比較およびノード使用頻度の検討を行う．そして従来の SDES に対する DDES の有効性をエレベータシャフト数の削減効果に関して検討する．

5.3.1 実行条件

Table 5.1，Table 5.2 に，ダブルデッキエレベータシミュレータの仕様および GNP の進化の実行条件を示す．Table 5.1 では，16 階床のうち基準階 2 階床（1F，2F），一般階 14 階床（3F～16F）である．Table 5.2 より，最適化を行うノードパラメータ（ $w_{X(p)}, r_j^Y$ ）の数は合計 69 個となる．ノードサイズ，交叉／突然変異率等の GNP の進化条件は全て経験的に設定している．Table 5.2 中の交叉および突然変異個体数の（ ）内の数値はノードパラメータの交叉，突然変異を同時に行う個体数を示す．交通流について，通常セミダブル運転は平常時やダウンピーク時に使用されるが [19]，本検討では行先階登録方式を導入した DDES を前提とするため，アップピーク時でも同運転による効果が期待できる [50]．したがってアップピーク時も考慮した 3 種類の交通流に対して進化を実行する．また GNP の評価はシミュレータ起動後 5 分後より開始し，シミュレーションは異なる 5 種類の乱数系列に対して実行した．

Table 5.1 Specification of Double-deck Elevator Simulator

Items	Value
Number of Floors	16
Number of Shafts (Cages)	6 (12)
Floor Distance [m]	4.5
Max. Velocity [m/s]	2.5
Max. Acceleration [m/s^2]	0.7
Jerk [m/s^3]	0.7
Cage Capacity [$person$]	20
Time for Opening Door [s]	2.0
Time for Closing Door [s]	2.3
Time for Riding [$s/person$]	1.0
Passenger Density [$person/h$]	
—Regular Time	2700
—Up-peak Time	2400
—Down-peak Time	3000

Table 5.2 Evolutional Conditions of GNP

Items	Value
Generation	300
Population Size	300
—Crossover (with Node Para. Cross.)	110 (100)
—Mutation (with Node Para. Mut.)	160 (150)
—Node Parameter Crossover, Mutation	10, 10
—Elite	10
Node Size	91+Initial Node
— $\mathbf{A}(d)$	10 (10/seed)
— $\mathbf{S}(X)$	60 (5/seed)
— $\mathbf{J}^Y(d)$, $\mathbf{J}^Z(d)$	12 (1/seed)
— \mathbf{ESD} , \mathbf{HCF} , \mathbf{DCF}	9 (3/seed)
Time Delay	
— $\mathbf{A}(d)$	5
— $\mathbf{S}(X)$	2
— $\mathbf{J}^Y(d)$, $\mathbf{J}^Z(d)$, \mathbf{ESD} , \mathbf{HCF} , \mathbf{DCF}	1
—Branch	0
Time Delay Threshold	30
Crossover Probability P_c , P_c^{np}	0.1, 0.1
Mutation Probability P_m , P_m^{np}	0.01, 0.01
Evaluation Time [h]	2
w_t , w_{ocs} , w_l	0.007, 0.001, 0.6

5.3.2 進化の様子

Fig. 5.3 (a)~(c) に各交通流における GNP の最良個体の適合度曲線を示す．図中の結果は 5 種類の乱数系列による実行結果（破線）およびその平均（実線）をそれぞれ表す．

Fig. 5.3 ではこれまでの検討結果と同様，各適合度曲線が確率的要素に起因する振動を含みつつ改善され，300 世代経過時にはほぼ収束している様子が見てとれる．すなわち提案モデルの GNP が DDES の群管理に対して最適化されていることがわかる．交通流別に見ると，(b) アップピーク時，(c) ダウンピーク時の適合度の収束値は 1200 ~ 1300 で各試行ともほぼ同じ値に収束しているのに比べ，(a) 平常時の適合度の収束値は 1500 ~ 1800 程度と大きく，また試行によるばらつきも大きい．これはアップピーク時やダウンピーク時の 1 方向へ偏った交通流よりも上下 2 方向の交通が存在する平常時のほうがシステムの挙動が複雑となり，その群管理が難しくなるためと考えられる．

5.3.3 従来手法との性能比較

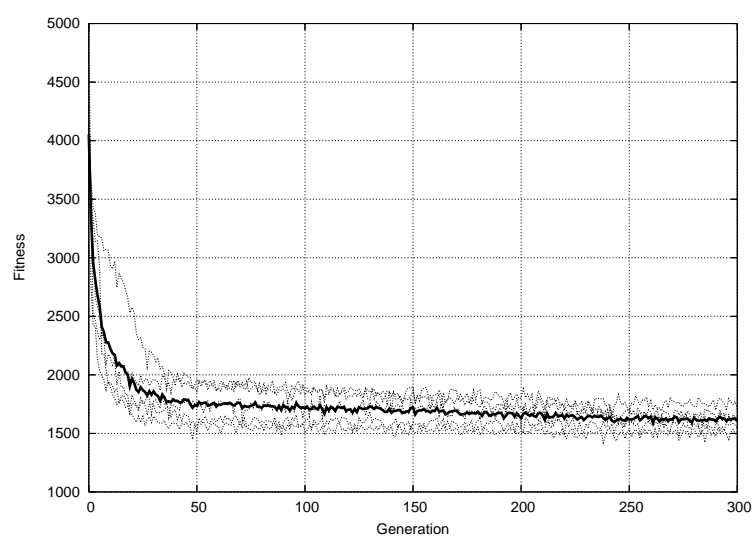
続いて，進化によって得られた GNP の最良個体（5 種類の乱数系列中の最良ケース）と，固有のかご評価式に基づく割当てを行う従来手法との性能比較を行う．従来手法は各かご評価項目の評価値の荷重和を Eq. (5.15) のかご評価式 $e_{conv}(i)$ により求め，その最小値をとるかごへ呼びを割当て方式である．

$$e_{conv}(i) = \sum_{V \in \Gamma} w_V \cdot v(i) \quad (5.15)$$

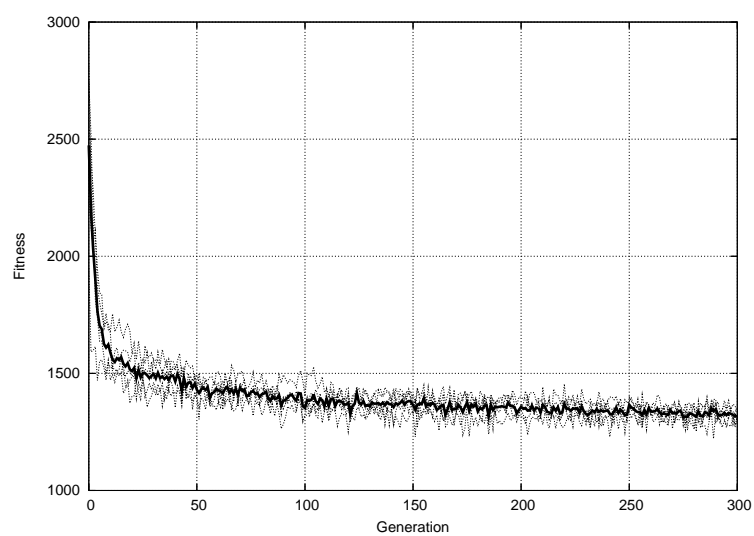
ここで，

$$\begin{aligned} w_V \in [0, 1] & : \text{評価項目 } V \text{ に関する重み付けパラメータ} \\ v(i) & : \text{かご } i \text{ の評価項目 } V \text{ に関する正規化変数} \end{aligned}$$

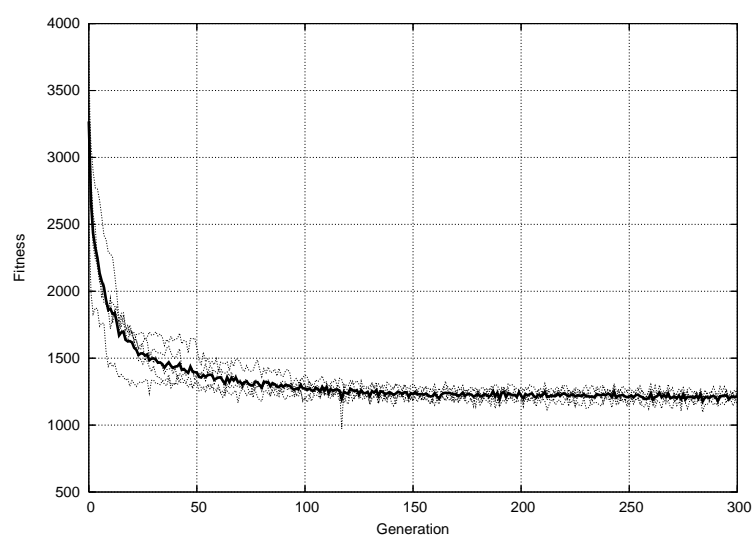
であり，正規化情報値 $v(i)$ も GNP のケースと同様に計算する．すなわち，12 種類の w_V からなる重みベクトル w ($w_V \in w$) の設定により多様な割当て方式が実現可能となる．本検討では経験的に Table 5.3 のように w を設定した 5 手法との比較検討を行う．また評価方法として文献 [32, 49] 等で実施されている方式を採用する．具体的には，システ



(a) Regular Time



(b) Up-peak Time



(c) Down-peak Time

Fig. 5.3 Fitness Curves in Each Traffic Flow

ムの軽負荷から重負荷までの挙動に対する群管理性能の比較検討を行うため、乗客密度を $300 \sim 4500$ [$person/h$] の範囲で 300 刻みで変更し評価する．性能比較は平均待ち時間 (AWT [s])、平均サービス完了時間 (AST [s])、60 [s] 以上の長待ち乗客率 (LWP [%])、片かごサービスとなった乗客の延べ人数 (OCSPN [$person$]) の 4 種類の指標に関して行う．

Fig. 5.4~Fig. 5.6 に、各交通流におけるシミュレーションの実行結果を上記 4 種類の指標について比較したものを示す．評価は 2 [h] のシミュレーションを各乗客密度において 30 回ずつ実行し、結果はその平均値をプロットしたものである．まず平常時の結果 Fig. 5.4 を見ると、AWT に関して軽負荷時は GNP は従来手法に比べ 2 [s] 程度長くなっているが重負荷になるにつれて逆転し、輸送能力の判断基準となる AWT が 30 [s] \sim 40 [s] の付近では最も性能が良くなっている．これは最適化対象をシステムの設計指針に基づき AWT が 30 [s] 程度となる乗客密度に設定したためである（アップピーク時、ダウンピーク時の設定も同様の議論による）．また実際に軽負荷時の 2 [s] 程度の待ち時間の悪化はそれほど問題とならず、したがって提案モデルにより輸送効率の改善が達成されているといえる．AST、OCSPN に関しては GNP は *AOMCS* とほぼ同等であり、その他の 4 手法に対しては明らかな性能改善を見せている．LWT に関しては AWT と同様に GNP は乗客密度 3900 [$person/h$] 程度までは最も性能が良くなっているが、極端な重負荷の場合では性能改善は見られない．したがって提案モデルは重負荷での輸送能力に関して検討の余地を残しているといえる．

次にアップピーク時の結果 Fig. 5.5 を見ると、GNP は AWT に関して軽負荷時は最も性能が悪く、重負荷には最も性能が良くなるという結果が得られている．これは GNP の適合度関数における片かごサービス乗客数の考慮が影響しているものと考えられる．すなわち、GNP は Eq. (5.14) の適合度関数によって待ち時間と片かごサービス乗客数を考慮した群管理の最適化を行うため、後者の最適化による前者へのトレードオフの影響が軽負荷では特に現れたと考えられる．また、GNP は AWT が 30 [s] \sim 40 [s] の付近では最も性能が良く、AST、LWP、OCSPN のいずれに対しても従来手法に対する顕著な性能改善を達成していることがわかる．乗客密度 4500 [$person/h$] 付近で GNP が従来手

Table 5.3 Weighting Parameter Setting of Conventional Methods

Method	w_{AIWT}	w_{MST}	w_{PN}	w_{CN}	w_{RR}	w_{MCN}	w_{OIWT}	w_{OST}	w_{PD}	w_{CD}	w_{SSN}	w_{OCD}
<i>A</i>	1	0	0	0	0	0	0	0	0	0	0	0
<i>AO</i>	1	0	0	0	0	0	1	0	0	0	0	0
<i>AOM</i>	1	1	0	0	0	0	1	0	0	0	0	0
<i>AOMC</i>	1	1	0	0	0	0	1	0	0	1	0	0
<i>AOMCS</i>	1	1	0	0	0	0	1	0	0	1	1	0

法より劣る部分があるが，従来手法ではアップピーク時の極端な重負荷によってシステムがダウン状態に陥っていると考えられ（3600 $[person/h]$ 以降の LWP，OCSPN が停滞状態にある），正当な比較にはならない．以上の結果から，アップピーク時において提案モデルは従来手法よりも優れた群管理性能を有しているといえる．

最後にダウンピーク時の結果 Fig. 5.6 を見ると，GNP は軽負荷での若干の悪化が見られるものの AWT が 30 $[s]$ ～ 40 $[s]$ の付近で最も性能が良くなっているのを始め，概ね全ての指標で従来手法に対する性能改善を達成している．また OCSPN に注目すると GNP は AOMCS とほぼ同等であり，これは両手法において評価項目 SSN を効果的に活用した群管理が実行されていることを示す．すなわち，平常時やダウンピーク時において上下かごの同時サービスを考慮する SSN は，一般階の乗客への効率的なサービスに貢献できる点で特に有効な知見であることがわかる．

以上の結果および考察より，提案モデルは各交通流に対して DDES の群管理目標を考慮した群管理の最適化を行い，固有の評価式に基づく従来手法よりも輸送効率と乗り心地の双方で優れた群管理性能を達成していると結論できる．

5.3.4 ノード使用頻度の検討

次に，GNP のノード使用頻度について検討する．Fig. 5.7(a)～(c) に，各交通流の GNP の最良個体が進化の実行条件と同じ乗客密度のテストシミュレーション 1 試行で使ったノードの頻度を示す．図中の横軸はノードラベルを表す．

ノード $S(X)$ ， $J^Y(d)$ ， $J^Z(d)$ の評価項目に注目して Fig. 5.7 を見ると，各交通流に共通して評価項目 $AIWT$ ， MST ， CD ， $OIWT$ ， SSN を活用した群管理が実行されていることがわかる．これはこれらの項目を考慮する AOMCS が Fig. 5.4～Fig. 5.6 で良好な性能を示していることから理解できる．交通流別に見ると，(a) 平常時では特に乗客数や乗車率を考慮する PD ， RR ， MCN が活用されており，上下方向の複雑な交通流が存在する平常時では，積み残しを起こさない乗客の効率的な乗降への考慮が重要といえる．次に (b) アップピーク時では特に発生呼びと登録呼びの関係を考慮する MCN ， OCD が活用されており，これはアップピーク時の群管理で重要となる，基準階の乗客の効率的

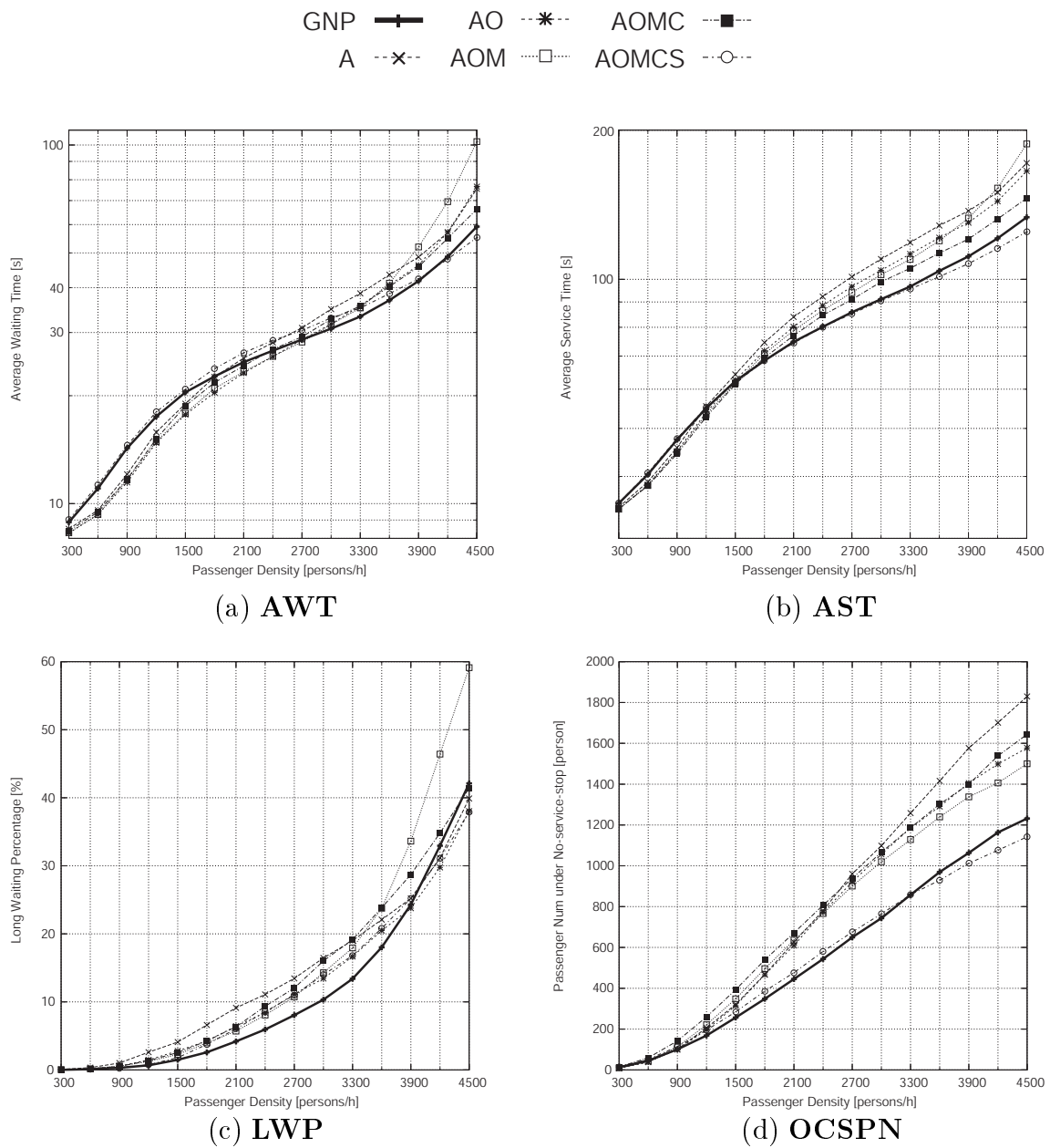


Fig. 5.4 Performance Comparisons among Methods in Regular Time

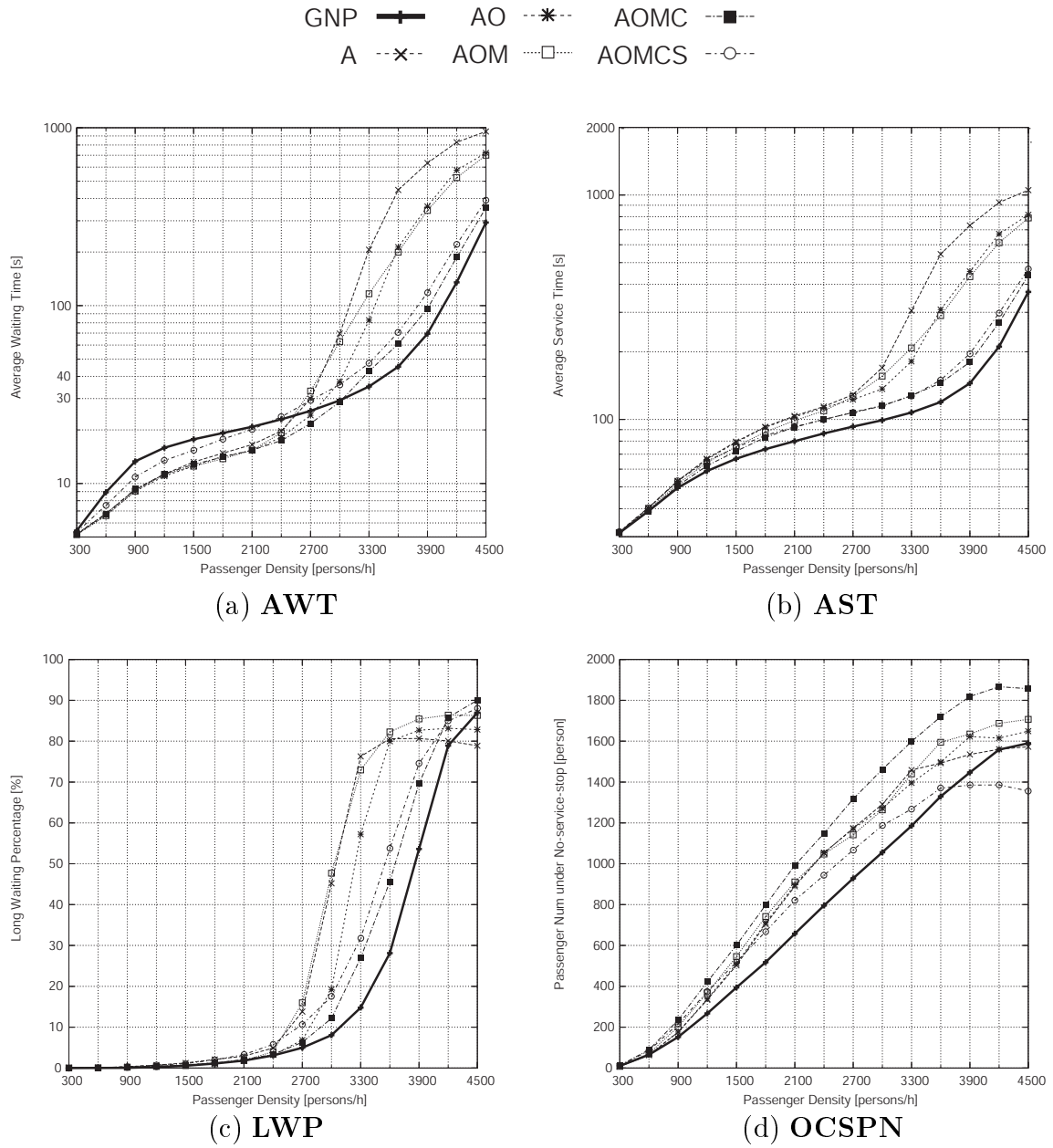


Fig. 5.5 Performance Comparisons among Methods in Up-peak Time

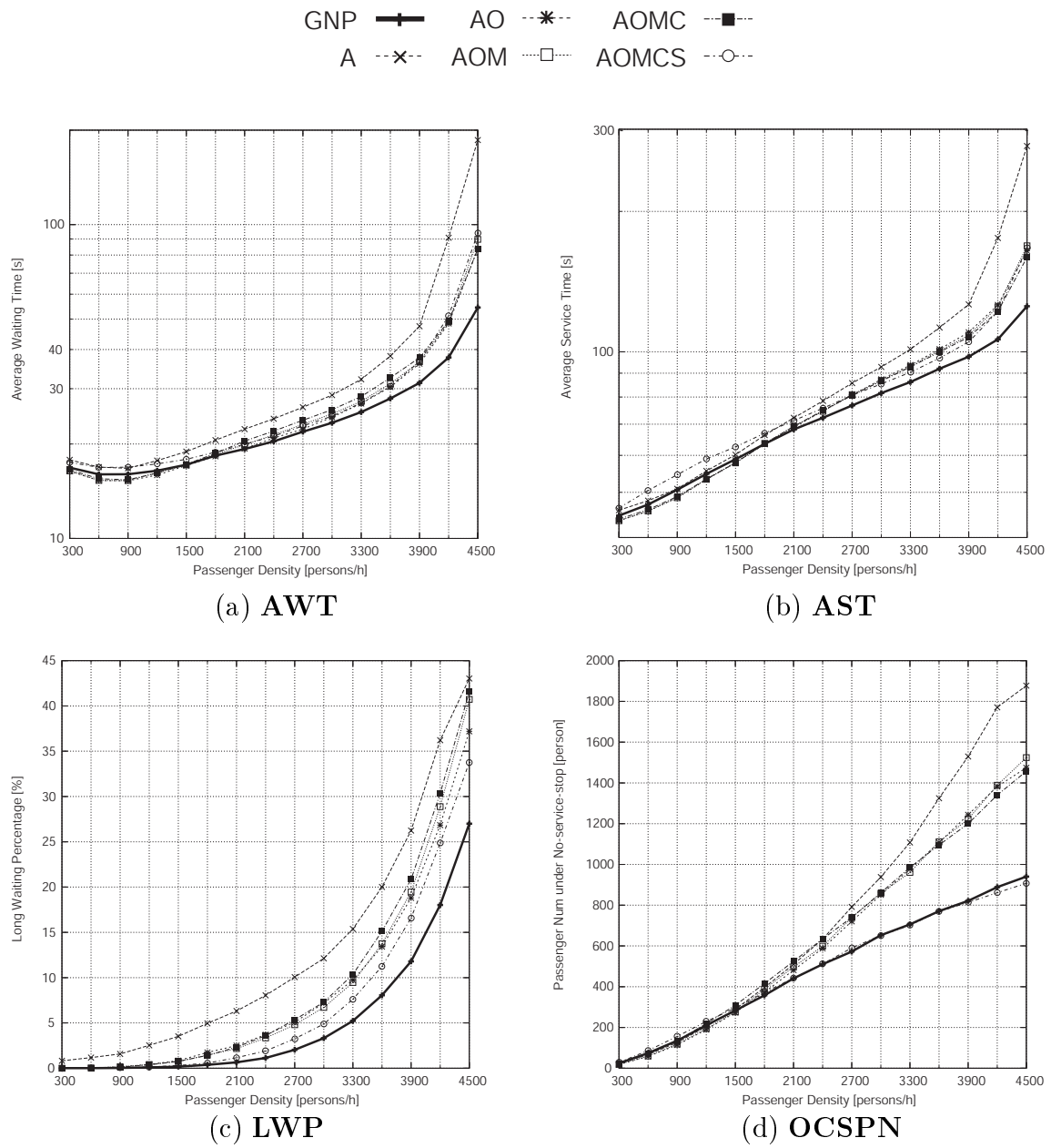


Fig. 5.6 Performance Comparisons among Methods in Down-peak Time

な乗り分けに関する評価項目が効果的に活用されたことを示している．最後に (c) ダウンピーク時では特に運行負荷とそのバランスを考慮する PD , PN , CN が活用されており，一般階へのサービスが主となるダウンピーク時では，上下かごへの効率的な乗車を考慮した群管理が有効とわかる．以上の検討結果から，提案モデルでは各交通流に有効な評価項目による群管理が実行されており，それによって今後の提案モデルの拡張に際して有用な知見の獲得に至ったことが結論できる．

5.3.5 従来システムに対する有効性の検討

最後に，提案モデルと従来の SDES の群管理システムとの比較を行い，DDES の輸送能力およびエレベータシャフト数の削減効果について検討する．比較する SDES の仕様は基本的に Table 5.1 と同条件とし¹，エレベータシャフト数を $\{6, 7, 8, 9, 10, 11, 12\}$ と設定し²それぞれ評価を行う．ここで，シャフト数 x でシャフト当たりかご数 y のエレベータシステムを $SxCy$ と表記する．すなわち本検討の DDES は $S6C2$ となる．また SDES の群管理は Table 5.3 の手法 A により行う．

これまでと同様に，SDES の各仕様に対し 3 種類の交通流の下で乗客密度を 300 ～ 4500 $[person/h]$ の範囲で 300 刻みで変更，評価して得られた AWT を，DDES による結果（GNP および手法 A）と併せて Fig. 5.8(a)～(c) に示す．評価は 2 $[h]$ のシミュレーションを各乗客密度において 30 回ずつ実行し，その平均値を比較している．Fig. 5.8 では各交通流とも $S6C2$ （GNP）および $S6C2$ （A）は同一シャフト数である $S6C1$ よりも AWT が改善しており，DDES によるシャフト数の削減効果が見てとれる．交通流毎に輸送能力の基準となる AWT が 30 $[s]$ ～ 40 $[s]$ の付近で比較すると，(a) 平常時では $S6C2$ （GNP）は $S9C1$ ， $S6C2$ （A）は $S8C1$ に相当し，それぞれ約 33 [%]，25 [%] のシャフト数の削減効果を達成している．同様に (b) アップピーク時では $S6C2$ （GNP）は $S12C1$ ， $S6C2$ （A）は $S10C1$ に相当し約 50 [%]，40 [%] のシャフト数の削減効果を，(c) ダウンピーク時では $S6C2$ （GNP）は $S9C1$ ， $S6C2$ （A）は $S8C1$ に相当し約 33 [%]，25 [%] の

¹SDES では基準階は 1F のみとなるため，ビル階床数は 15 階床とする．

²一般には 9 シャフト以上のエレベータシステムは使用されないが，本検討では比較のため考慮する．

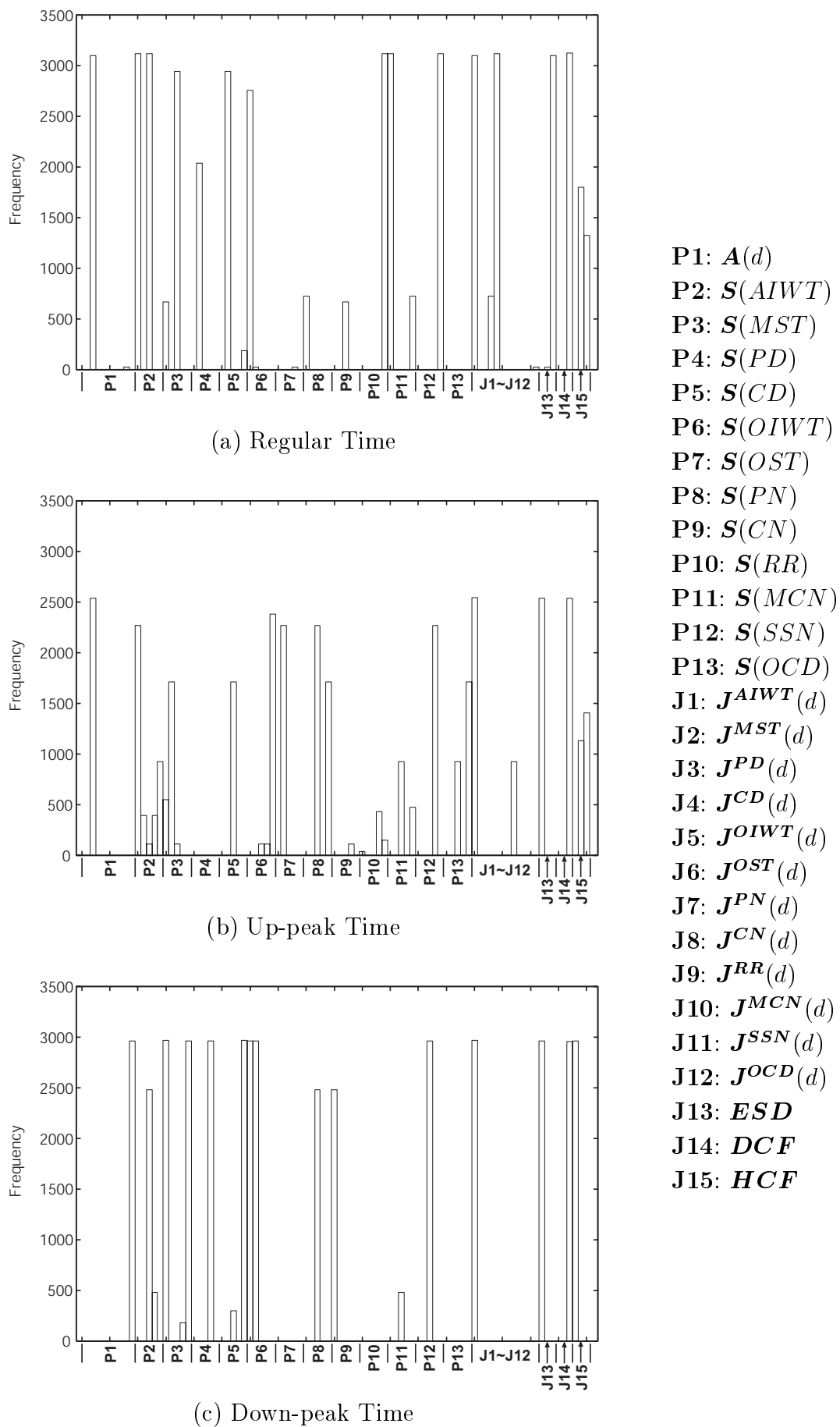


Fig. 5.7 Node Usage Frequency of GNP in Each Traffic Flow

シャフト数の削減効果をそれぞれ達成している．以上の結果から DDES は SDES よりも輸送効率，設備コストの面で優れたシステムであるといえる．また $S6C2$ (GNP) と $S6C2$ (A) では SDES におけるシャフト数 1~2 本分の性能差があり， $S6C2$ (GNP) は重負荷時でも著しい性能悪化は見られない点から考えて，提案モデルが DDES の有効な群管理方式であることも理解できる．

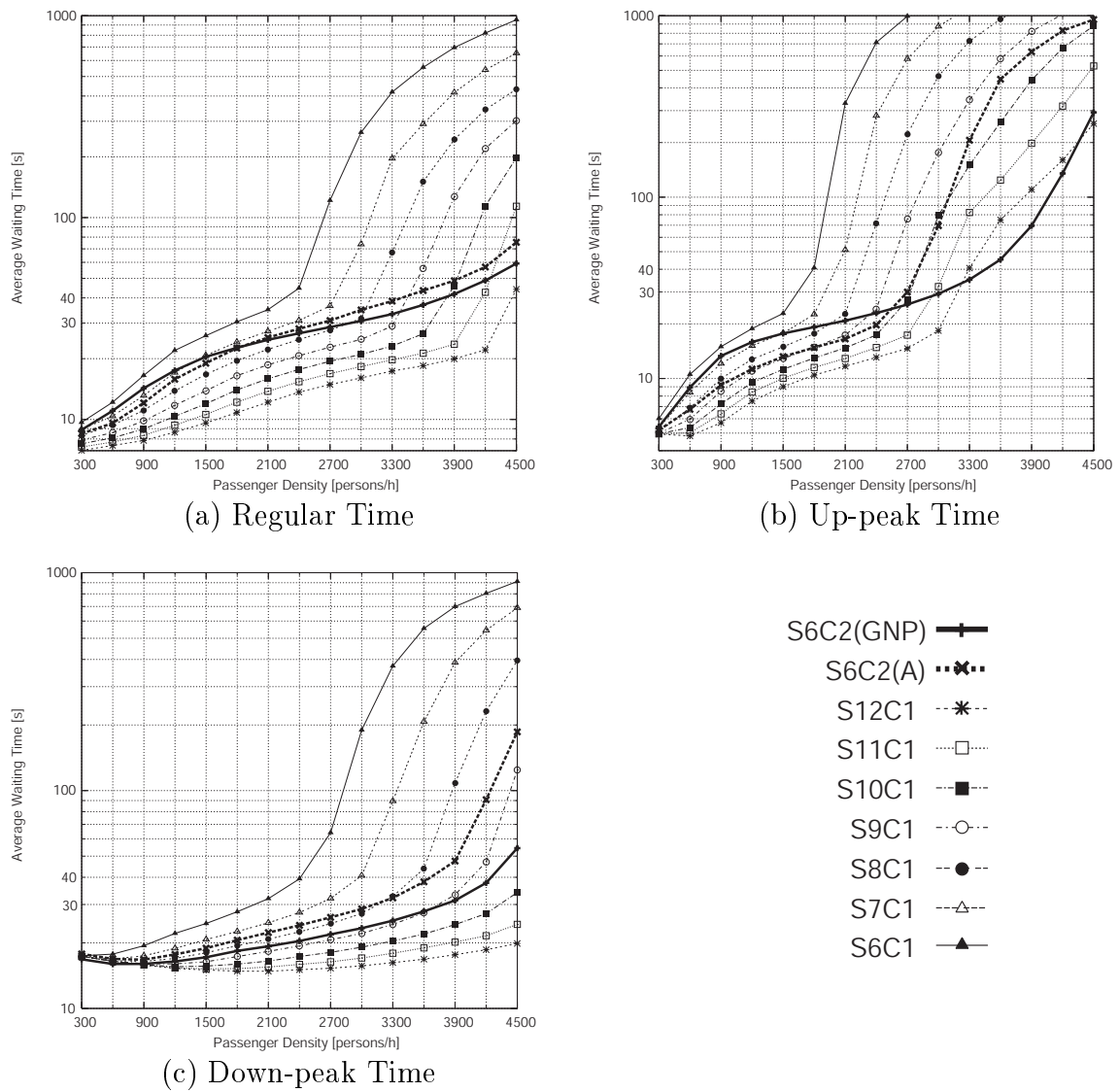


Fig. 5.8 Performance Comparisons between DDES and SDES in each Traffic Flow

5.4 まとめ

本章では、第2章から第4章までの検討結果をふまえ、GNPによるダブルデッキエレベータ群管理システムを提案し、その基礎検討を行った。提案モデルとして、DDES特有の挙動を考慮した12種類のかご評価項目を定義し、またかご評価式の最適化を応用した柔軟なかご選定および評価を行うGNPを構成した。

シミュレーションでは5種類の従来手法との性能比較、ノード使用頻度および従来のSDESに対するシャフト数の削減効果に関して、提案モデルの性能評価を行った。その結果、各方面での提案モデルの有効性を確認した。したがって本モデルは、行先階登録方式を導入したDDESのセミダブル運転時の有効な群管理方式であると結論できる。今後はこの検討結果をもとに、各種の有効なアルゴリズムを導入し提案モデルを拡張していくことを期待する。

第6章

結論

本研究では近年提案された新しい進化論的計算手法遺伝的ネットワークプログラミング (GNP) の、大規模複雑システムの代表例であるエレベータ群管理システム (EGSCS) への導入とその最適化制御に関する検討を行ってきた。

第2章では、GNP の EGSCS への導入に関する利点を整理し、その導入可能性および基本特性を検討するための基本モデルを提案した。シミュレーションの結果、GNP は EGSCS に対して進化による適切な構造の最適化を実行し、単純なルールに基づく従来手法との比較において性能の改善が見られた。また同時に問題点も浮彫りにした。第2章の検討結果より、GNP による EGSCS の基礎検討段階での有効性および課題点を示した。

第3章では、基本モデルの問題点の解決のため、新たにランキング処理とノード関数最適化を導入した拡張モデルを提案した。シミュレーションでは実用性を考慮し、動的に変化するビルの1日の交通データによる性能評価を行った。その結果、拡張モデルの GNP は基本モデルの問題点を解消する柔軟な進化と一般性に富む群管理ルール生成を実現し、従来手法よりも優れた群管理性能が得られた。第3章の検討結果より、高い適応性と一般性を備えた GNP による呼び割当て方式を確立した。

第4章では、第3章の拡張モデルによる呼びの割当て方式に交通流検出・切替え機能を付加した、切替え制御型群管理システムを提案した。提案方式の切替え GNP は重要度指標付き GNP によって構成し、第3章と同条件下でのシミュレーションにより、切替え GNP が交通流の変化に対して割当て GNP を切替え、効率的な群管理を実行することを確認した。第4章の検討結果より、GNP による呼び割当て機能と交通流検出・切替え機能が連動する切替え制御型群管理システムを確立し、その有効性を示した。

以上第2章から第4章までの検討結果より，GNPによるEGSCSの基本的枠組みを確立し，次の目標であるダブルデッキエレベータシステム（DDES）への導入の足掛かりとすることができた．

第5章ではこれまでの検討結果をふまえたGNPによるダブルデッキエレベータ群管理システムを提案し，その基礎検討を行った．提案モデルでは第3章の拡張モデルをさらに発展させ，かご評価式による柔軟な群管理ルール生成を行うアルゴリズムを導入した．シミュレーションの結果，提案モデルは従来手法に対する輸送効率や乗り心地，設備コスト等の顕著な性能改善を達成し，その有効性を確認した．第5章の検討結果より，GNPによるダブルデッキエレベータ群管理システムの有効性を示し，今後の実用化へ向けた拡張および発展へとつなげることができた．

本研究を通じて，GNPによるエレベータ群管理の最適化に関して各種の知見や技術を駆使した独創的かつ効果的な群管理モデルを確立し，シミュレーションによりその有効性を明らかにすることができた．またこれからの研究開発が期待されているダブルデッキエレベータの群管理に関して，GNPによる群管理モデルの有効性を示した．これらの研究成果はGNPによるエレベータ群管理研究の核となる技術として，これからの応用展開に貢献できる．今後は本研究による群管理技術がさらなる発展を遂げ，実用化へと結び付くことを期待し，以上の本研究に対する結論をもって本論文の結びとする．

参考文献

- [1] 加納秀明, システムの最適理論と最適化, コロナ社, 1987.
- [2] 田村坦之, 大規模システム, 昭晃堂, 1986.
- [3] 志村正道, 人工知能, 森北出版, 1994.
- [4] 有田隆也, 人工生命, 科学技術出版, 2000.
- [5] 伊庭斉志, 進化論的計算の方法, 東京大学出版会, 1999.
- [6] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, 1975, MIT Press, 1992.
- [7] 北野宏明, 遺伝的アルゴリズム, 産業図書, 1993.
- [8] 北野宏明, 遺伝的アルゴリズム 3, 産業図書, 1997.
- [9] 坂和正敏, 田中雅博, 遺伝的アルゴリズム, 朝倉書店, 1995.
- [10] J. R. Koza, *Genetic Programming, On the Programming of Computers by Means of Natural Selection*, MIT Press, 1992.
- [11] J. R. Koza, *Genetic Programming III, Darwinian Invention and Problem Solving*, MIT Press, 1999.
- [12] 伊庭斉志, 遺伝的プログラミング, 東京電機大学出版局, 1996.
- [13] L. J. Fogel, A. J. Owen and M. J. Walsh, *Artificial Intelligence through simulated Evolution*, John Wiley & Sons, 1966.

- [14] D. B. Fogel, *Evolutionary Computation: Towards a New Philosophy of Machine Intelligence*, IEEE Press, 1995.
- [15] G. Barney and S. dos Santos, *Elevator Traffic Analysis, Design and Control, Second Ed*, Peter Peregrinus Ltd, 1985.
- [16] マルコン, 佐々木, 安田, “エレベータ運行管理システム”, エレベータ界, pp. 30–34, 10月号 1998.
- [17] G. Barney, *Elevator Traffic Handbook*, Spon Press, 2003.
- [18] 匹田, 阿部, “エレベータ群管理制御における AI 技術の応用”, 人工知能学会誌, Vol. 17, No. 1, pp. 57–62, 2002.
- [19] M. Amano, M. Yamazaki and H. Ikejima, “The Latest Elevator Group Control System”, In *Proc. of ELEVCAN'95*, pp. 88–95, 1995.
- [20] 平澤宏太郎, 大久保雅文, 胡敬炉, 村田純一, “蟻の行動進化における Genetic Network Programming と Genetic Programming の性能比較”, 電気学会論文誌 C, Vol. 121-C, No. 6, pp. 1001–1009, 2001.
- [21] H. Katagiri, K. Hirasawa, J. Murata, J. Hu and M. Kosaka, “Network Structure Oriented Evolutionary Model: Genetic Network Programming-Its Comparison with Genetic Programming-”, 計測自動制御学会論文集, Vol. 38, No. 5, pp. 485–494, 2002.
- [22] 片桐広伸, 平澤宏太郎, 古月敬之, 村田純一, “Genetic Network Programming とそのマルチエージェントシステムへの応用”, 電気学会論文誌 C, Vol. 122, No. 12, pp. 2149–2156, 2002.
- [23] 間普真吾, 平澤宏太郎, 古月敬之, “学習進化型遺伝的ネットワークプログラミングとそのタイルワールド問題への適用”, 計測自動制御学会論文集, Vol. 40, No. 11, pp. 1105–1113, 2004.

- [24] 平澤, 葛貫, 岩坂, 金子, 河竹, “エレベータ群管理における呼び割付け方式”, 電気学会論文誌 C, Vol. 99, No. 2, pp. 1–6, 1979.
- [25] 黒沢, 平澤, 葛貫, 米田, 坂井, 萩中, “エレベータの知能群管理方式”, 情報処理学会論文誌, Vol. 26, No. 2, pp. 278–287, 1985.
- [26] 黒沢, 平澤, 葛貫, 米田, 坂井, 萩中, “エレベータの知能群管理方式 その2 : 学習論理”, 情報処理学会論文誌, Vol. 28, No. 3, pp. 299–309, 1987.
- [27] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.
- [28] 田中一男, アドバンストファジィ制御 ファジィ制御の新たなる展開, 共立出版, 1994.
- [29] 飛田, 藤野, 瀬川, 米田, 市川, “遺伝的アルゴリズムを応用したエレベータ群管理用パラメータ調整方法”, 電気学会論文誌 D, Vol. 117, No. 3, pp. 306–313, 1997.
- [30] A. Fujino, T. Tobita, K. Segawa, K. Yoneda and A. Togawa, “An Elevator Group Control System with Floor-attribute Control Method and System Optimization using Genetic Algorithms”, *IEEE Trans. on Ind. Electron.*, Vol. 44, No. 4, pp. 546–552, 1997.
- [31] X. Bi, C. Zhu and Q. Ye, “A GA-Based Approach to the Multi-Objective Optimization Problem in Elevator Group Control System”, In *Elevator World*, pp. 58–63, June 2004.
- [32] 鈴木, 高橋, 佐野, 須藤, マルコン, 喜多, “遺伝的アルゴリズムによるマルチカーエレベータ制御ルールのシミュレーションベース最適化”, 計測自動制御学会論文集, Vol. 40, No. 4, pp. 466–473, 2004.
- [33] P. Cortes, J. Larraneta and L. Onieva, “Genetic Algorithm for Controllers in Elevator Groups: Analysis and Simulation during Lunchpeak Traffic”, *Applied Soft Computing*, Vol. 4, pp. 159–174, 2004.

- [34] 宮武, 古関, 曽根, “ロープレスエレベータシステムの提案とその有効性評価”, 電気学会論文誌 D, Vol. 119, No. 11, pp. 1353–1360, 1999.
- [35] T. Sudo and S. Markon, “The Performance of Multi-car Linear Motor Elevators”, In *Elevator World*, pp. 81–85, March 2002.
- [36] マルコン, 喜多, 西川, “ニューラルネットワークを利用した適応型エレベータ群管理システム”, システム制御情報学会論文誌, Vol. 7, No. 12, pp. 487–497, 1994.
- [37] 岩田, 匹田, 駒谷, “ニューラルネットワークによるエレベータ交通流の検出”, 計測自動制御学会論文集, Vol. 33, No. 3, pp. 209–215, 1997.
- [38] B. A. Powell, D. J. Sirag and B. L. Whitehall, “Artificial Neural Networks in Elevator Dispatching”, In *Elevatori*, pp. 44–57. 2000.
- [39] H. Wan, C. Liu and H. Liu, “NN Elevator Group-Control Method”, In *Elevator World*, pp. 149–154, February 2003.
- [40] 匹田, 駒谷, “ファジィルールベースを用いた新しいエレベータ群管理システム”, 計測自動制御学会論文集, Vol. 25, No. 1, pp. 99–105, 1989.
- [41] C. Kim, K. Seong, H. Lee-Kwang and J. O. Kim, “Design and Implementation of a Fuzzy Elevator Group Control System”, *IEEE Trans. on System, Man and Cybernetics, PART-A*, Vol. 28, No. 3, pp. 277–287, 1998.
- [42] R. Gudwin, F. Gomide and M. Netto, “A fuzzy elevator group controller with linear context adaptation”, In *Proc. of FUZZ-IEEE98, WCCI’98—IEEE World Congress on Computational Intelligence*, pp. 481–486, 1998.
- [43] R. Crites and A. Barto, “Elevator Group Control Using Multiple Reinforcement Learning Agents”, *Machine Learning*, Vol. 33, pp. 235–262, 1998.

- [44] 小越, 木村, 広瀬, 大里, “マルチエージェントシステムを用いたエレベータ群管理システム”, 電子情報通信学会論文誌 D-I, Vol. J84-D-I, No. 2, pp. 191–202, 2001.
- [45] S. Hikita, M. Amano and H. Ando, “The Latest Elevator Group Control System”, In *Proc. of ELEVCN2001*, pp. 35–43, 2001.
- [46] 匹田, 岩田, 阿部, “行先階登録と適応制御によるエレベータ群管理”, 電気学会論文誌 C, Vol. 124, No. 7, pp. 1471–1477, 2004.
- [47] T. Beielstein, C-P. Ewald and S. Markon, “Optimal Elevator Group Control by Evolution Strategies”, In *Proc. of Genetic and Evolutionary Computation Conference*, pp. 1963–1974, 2003.
- [48] J. W. Fortune, “Modern Double-deck Elevator Applications and Theory”, In *Elevator World*, pp. 63–68, August 1996.
- [49] M-L. Siikonen, “Double-deck Elevators: Saving in Time and Space”, In *Elevator World*, pp. 65–69, July 1998.
- [50] J. W. Fortune, “Predestination Hall Call Selection for Double-deck Lifts (3D Encoding)”, In *Elevator World*, pp. 126–133, August 2005.
- [51] 江口徹, 周金, 平澤宏太郎, 古月敬之, マルコンシャンドル, “Genetic Network Programming によるエレベータ群管理システムの基礎検討”, 電気学会論文誌 C, Vol. 125, No. 7, pp. 1055–1062, 2005.
- [52] T. Eguchi, K. Hirasawa, J. Hu and S. Markon, “Elevator Group Supervisory Control System Using Genetic Network Programming”, In *Proc. of IEEE Congress on Evolutionary Computation 2004*, pp. 1661–1667, 2004.
- [53] 江口徹, 平澤宏太郎, 古月敬之, マルコンシャンドル, “Genetic Network Programming によるエレベータ群管理システムの評価”, 電気学会電子・情報・システム部門大会講演論文集, pp. 526–531, 2004.

- [54] 江口徹, 周金, 平澤宏太郎, 古月敬之, マルコンシャンドル, “ランキング処理とノード関数最適化を考慮した Genetic Network Programming によるエレベータ群管理システム”, 計測自動制御学会論文集, Vol. 42, No. 3, 2006.
- [55] T. Eguchi, K. Hirasawa, J. Hu and S. Markon, “Elevator Group Supervisory Control System Using Genetic Network Programming –Ranking Processing and Node Function Optimization–”, In *Proc. of SICE Annual Conf. 2005*, pp. 1–6, 2005.
- [56] 江口徹, 平澤宏太郎, 古月敬之, マルコンシャンドル, “Genetic Network Programming によるエレベータ群管理システムの構築”, SICE システム・情報部門学術講演会 2004 講演論文集, pp. 73–78, 2004.
- [57] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs, Second, extended edition*, Springer Press, 1994.
- [58] T. Eguchi, J. Zhou, S. Eto, K. Hirasawa, J. Hu and S. Markon, “Elevator Group Supervisory Control System using Genetic Network Programming with Functional Localization”, *Journal of Advanced Computational Intelligence and Intelligent Informatics*, Vol. 10, 2006.
- [59] T. Eguchi, K. Hirasawa, J. Hu and S. Markon, “Elevator Group Supervisory Control System Using Genetic Network Programming with Functional Localization”, In *Proc. of IEEE Congress on Evolutionary Computation 2005*, pp. 328–335, 2005.
- [60] 江藤慎治, 平澤宏太郎, 古月敬之, “重要度指標付き機能局在型 Genetic Network Programming の機能切り替えについて”, 第 15 回インテリジェント・システム・シンポジウム予稿集, pp. 73–78, 2005.
- [61] Y. Fujita, T. Shimazaki, N. Kondou, K. Mishima and Y. Nakada, “Double-deck Elevators with Adjustable Floor Height”, In *Elevator World*, pp. 136–141, August 2003.

- [62] J. Sorsa, M-L. Siikonen and H. Ehtamo, “Optimal Control of Double-deck Elevator Group using Genetic Algorithm”, *International Trans. in Operational Research*, Vol. 10, No. 2, pp. 103–114, 2003.
- [63] J. Schroeder, “Advanced Dispatching”, In *Proc. of ELEVCON’90*, pp. 189–198, 1990.
- [64] R. Smith and R. Peters, “ETD Algorithm with Destination Dispatch and Booster Options”, In *Elevator World*, pp. 136–143, July 2002.
- [65] A. G. Barto and R. S. Sutton, *Reinforcement Learning - An Introduction*, MIT Press, 1998.
- [66] 片桐広伸, 平澤宏太郎, 古月敬之, 村田純一, “ノード数可変型 Genetic Network Programming”, *電気学会論文誌 C*, Vol. 123, No. 1, pp. 57–66, 2003.
- [67] S. Eto, T. Eguchi, K. Hirasawa and J. Hu, “Evolutional Acquisition of Communication Between Agents Using Genetic Network Programming”, *J. Machine Intelligence & Robotic Control*, Vol. 5, No. 1, pp. 25–32, 2003.
- [68] 森茂男, 平澤宏太郎, 古月敬之, “Genetic Network Programming による株価予測と売買モデル”, *電気学会論文誌 C*, Vol. 125, No. 4, pp. 631–636, 2005.
- [69] 嶋田香, 平澤宏太郎, 古月敬之, “獲得した情報を用いる遺伝的ネットワークプログラミングによるデータマイニング”, *情報処理学会論文誌*, Vol. 46, No. 10, pp. 2576–2586, 2005.
- [70] A. Teller and M. Veloso, “PADO, Learning Tree-structured Algorithm for Orchestration into an Object Recognition System”, Technical report library, Carnegie Mellon University, 1995.
- [71] M-L. Siikonen, T. Susi and H. Hakonen, “Passenger Traffic Flow Simulation in Tall Buildings”, In *Elevator World*, pp. 117–123, August 2003.

- [72] X. Bi, C. Zhu and Q. Ye, “The Development of Elevator Group Control Simulation Testbed”, In *Elevator World*, pp. 97–102, December 2003.
- [73] D. O. Hebb, *The Organization of Behavior - A Neuropsychological Theory*, John Wiley & Son, 1949.
- [74] 澤口俊之, 知性の脳構造と進化, 海鳴社, 1989.
- [75] 石田亨, 片桐恭弘, 桑原和宏, 分散人工知能, コロナ社, 1996.
- [76] K. Hirasawa, M. Obayashi, S. Sakai and J. Hu, “Learning Petri Network and Its Application to Nonlinear System Control”, *IEEE Trans. on System, Man and Cybernetics, PART-B*, Vol. 28, No. 6, pp. 781–789, 1998.
- [77] Q. Xiong, K. Hirasawa, J. Hu and J. Murata, “A Functions Localized Neural Network with Branch Gates”, *Neural Networks*, Vol. 16, pp. 1461–1481, 2003.

付 録 A

遺伝的ネットワークプログラミング (GNP)

ここでは、遺伝的ネットワークプログラミング (Genetic Network Programming: GNP) について述べる。GNP は近年開発された新しい進化論的計算手法であり、有向グラフ構造によるプログラムの自動生成を行う。一般にグラフ構造のプログラム合成手法は木構造プログラムによる手法よりも表現能力の点で優れているとされるが、GNP もこれまでに木構造の進化論的計算手法である遺伝的プログラミング (GP) [10, 11, 12] との性能比較によってその有効性が報告されている [20, 21, 23]。また GNP は、ニューラルネットワークや強化学習等の他の AI 技術の導入による効果的なアルゴリズムの拡張 [23, 66, 67] および株価予測モデル、データマイニング、EGSCS といった実用的応用分野への導入等 [51, 68, 69] の多岐にわたる検討が行われ、各方面において有効性を確認している。

以下、GNP の基本概念、基本構成、進化のアルゴリズムおよび遺伝的オペレータについて述べる。

A.1 基本概念

GNP はノード関数 (以降、ノードと表記) が有向枝によって相互にリンクされた有向グラフ構造を持つ。ノードはプログラム上の判断や行動を処理する最小単位であり、このノード間の接続を辿り処理を繰返すことで GNP は環境に対する行動系列を生成する。従来の木構造 GP およびグラフ構造による他手法との相違点は、行動終了後に起点となるノードには戻らず次の有向枝の接続先ノードより処理を実行する点である。すなわち GNP では常に以前の判断や行動の影響下で行動系列を生成していくことになる。これを

GNP の暗黙的記憶と呼ぶことにする．また，GNP の行動系列は部分観測マルコフ決定過程 (Partially Observable Markov Decision Process: POMDP) を保証すると考えることができ，これは GNP が適切なノードの設定によって環境の部分的な情報のみを入力として行動系列を生成できることを示す．換言すると，GNP は環境からの情報が不完全な部分観測マルコフ環境下での適用が可能な手法といえる．またグラフ構造の利点としてプログラムサイズ固定による構造の最適化¹が可能であり，交叉や突然変異等の遺伝的オペレータの実装が容易に行えることも特徴のひとつである．

上記特徴に基づき，GNP と代表的な進化論的計算手法である GP, Parallel Algorithm Discovery and Orchestration (PADO) [70] および進化的プログラミング (EP) [13] との相違点について以下に整理する．

【GP との相違点】 GP は木構造を基本とする手法であり，一般に *root* と呼ばれるプログラム開始ノードより処理が開始され，終端ノードで行動を実行後，次回行動時にも再び *root* ノードより処理を開始する．すなわち過去の行動は現在の行動に対して直接的な影響を及ぼさず，過去の行動を活用する際には外部記憶を必要とする．また木構造の最適化では *bloat* と呼ばれる交叉の繰返しによるプログラムサイズの爆発的増加が問題となる．

【PADO との相違点】 PADO はグラフ構造を持つ GP のひとつであり，音声認識や画像認識といった応用分野への適用が行われている．PADO も GP と同じく行動決定後はプログラム開始ノードへ処理が戻る．そして過去の行動結果を記憶しておく外部記憶領域を必要とする．

【EP との相違点】 EP は有限状態オートマトン (Finite State Automaton: FSA) の最適化を行うグラフ構造による進化論的計算手法である．FSA の特徴として，ある状態における全入力に対する応答を用意する必要がある．すなわち EP はマルコフ決定過程 (Markov Decision Process: MDP) を前提とした環境に対してのみ適用可能な手法である．またそのような環境においても，莫大な入力情報が存在する環境下では構造の最適化は困難となり，実用的ではない．

¹文献 [66] のようにプログラムサイズの制御を行うアルゴリズムも提案されている．

以上の相違点から，GNP は他の進化論的計算手法の問題点を克服できる特徴を持ち，従来手法よりも効率的な問題解決を期待できる．

A.2 基本構成

Fig. A.1 に GNP の基本構成を示す．既に述べたとおり，GNP は有向グラフ構造により個体を表現する．そして複数の個体群による集団を構成し，遺伝的オペレータを施すことでグラフ構造の最適化を行う．以下，GNP 個体に関して要点を述べる．

A.2.1 構造および実行手順

GNP は判定ノード，処理ノードおよび初期ノードが有向枝により相互に接続された構造を持つ．判定ノードは環境からの入力情報の判定とその結果に従った条件分岐を担い，処理ノードは GNP が環境に対して定められた処理（行動）を実行する役割を持つ．判定ノードからは判定結果に対応する有向枝が伸びているが，処理ノードからは次の遷移先を決める 1 本の枝のみが伸びる．初期ノードはプログラム開始時に処理の起点となり，接続先ノードへ処理を遷す役割を持つ．各ノードの機能は設計者によって予め適切なものが用意され，各々にノードラベルとして記載される．ノードは関数ライブラリで管理され，ノード数（プログラムサイズ）は基本的に GNP の進化を通じて一定である．

GNP は環境からの特定のタイミングで起動し処理を開始する．GNP プログラムの実行は，上記のノード遷移則に従って判定ノード，処理ノード上の遷移処理を行なう．処理ノードの実行後は初期ノードには戻らず，次の行動時にその処理ノードの接続先のノードから処理を再開する．

A.2.2 プログラムの表現

Fig. A.2 に GNP の表現型（Phenotype）と遺伝子型（Genotype）を示す．表現型と遺伝子型間のコード化は直接的なコード化を採用しており，遺伝子はノード遺伝子の集合からなる．Fig. A.2 に示されるノード i の各遺伝子はそれぞれ以下の情報を持つ． NID_i

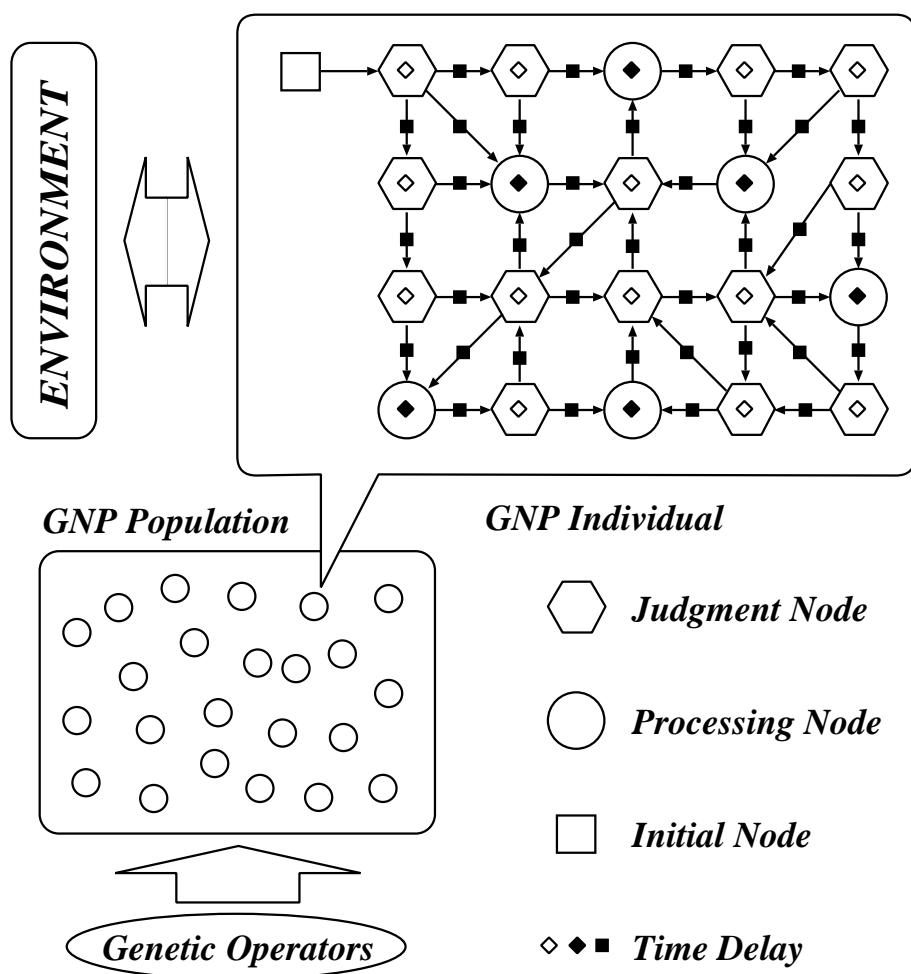


Fig. A.1 Basic Structure of GNP

はノード i のノード ID であり, ノード毎に固有値が設定されている. NT_i はノード型 ID を表し (0: 初期ノード, 1: 判定ノード, 2: 処理ノード) のようにノードの関数型を規定する. NF_i はノード関数 ID を表し, 上記 2 種類の ID からノードラベルを特定する. そして C_{ik} はノード i から伸びる k 番目の有向枝の接続先情報であり, 接続するノード ID が記載される. ここで $N(i)$ はノード i の持つ有向枝数を表す.

A.2.3 遅れ時間による実時間処理

Fig. A.2 に示すように, GNP は各ノード処理およびノード間の遷移に要する遅れ時間情報 (d_i, d_{ik}) を持つ. ここで d_i はノード i の処理に要する遅れ時間, d_{ik} はノード i から k への遷移時に要する遅れ時間を表す. これらは人間の脳の意思決定プロセスが, ある時間経過の中で実行されるメカニズムを GNP 上でモデル化したものである. これは GNP の適用問題によっては, 限られた時間内で行動を実行するリアルタイム処理が要求されるためである.

上記要求を満足するため, GNP では以下の処理を考慮している. 実行に際して, 遷移開始時の判定ノードから処理ノードに至るまでの GNP のノード処理および遷移に要した遅れ時間の累積値を計上する. この累積値が予め設定した遅れ時間閾値を超過すると, GNP は処理がループに陥ったと判断し実行を停止する. 停止した GNP は行動がキャンセルされるため, 代わりとなる行動が必要な場合はこれを外部処理により実行する. またループに陥った回数を計上し, これを適合度計算に反映させることでループ遺伝子の淘汰を考慮した進化が可能となる. 以上の遅れ時間を用いたループ検出機能により, GNP は上記要求に対する適切なリアルタイム処理を実現できる.

A.2.4 適合度

適合度 (Fitness) は GNP 個体の適用問題の達成度を数値化したものであり, その計算式は問題に依存して設計者によって適切に設定される. 個体の適合度に基づいて遺伝的オペレータが実行され, 適合度は世代経過に伴い改善されていく. すなわち, GNP の構造の最適化は適合度の最適化に対応する.

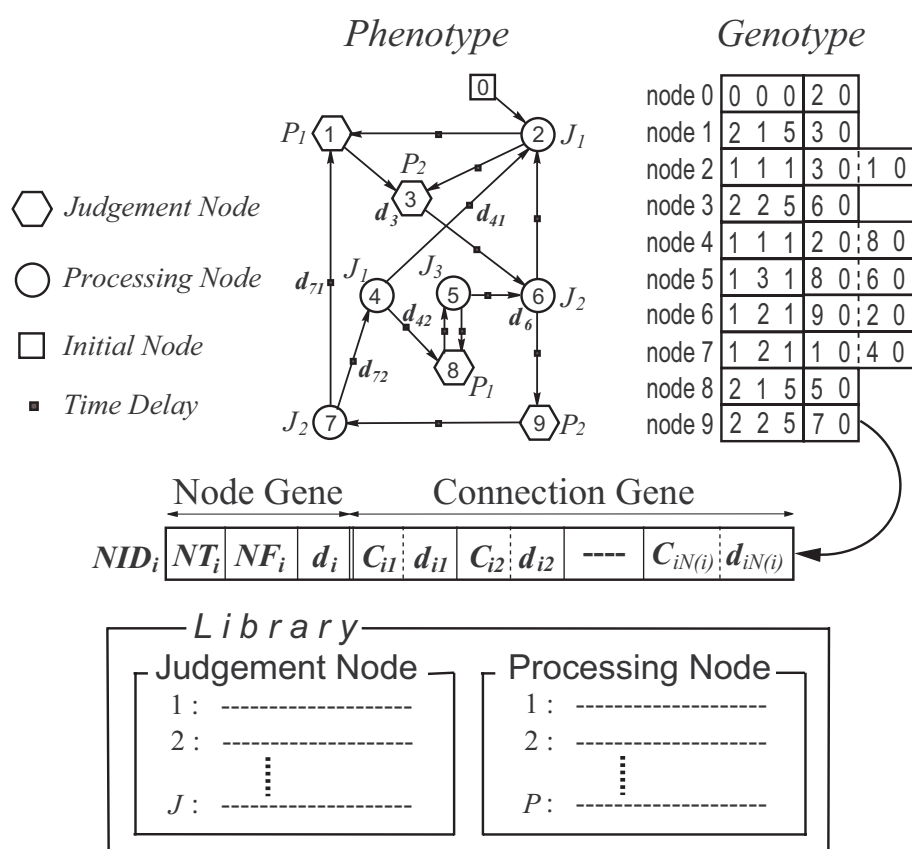


Fig. A.2 Phenotype and Genotype of GNP

A.3 進化のアルゴリズム

GNP の進化アルゴリズムは一般的な進化論的計算手法の枠組みに基づいている．具体的には世代毎に集団の個体数と同数の個体を遺伝的オペレータにより生成し，個体の入れ替えを行う．各遺伝的オペレータにより生成される個体数は予め決定され，常に一定数の個体が生成される．GNP の進化のアルゴリズムの一般的なフローを以下に示す．

進化のアルゴリズム

- (1) 初期 GNP 個体群を生成する（各個体の接続先はランダムに決定）
- (2) 初期個体群の適合度を計算する．
- (3) 遺伝的オペレータにより親個体群から子個体群を生成する．
- (4) 子個体群の適合度を計算する．
- (5) 次世代の親個体群を決定する．
- (6) 終了条件を満たせば終了する．満たさなければ 1 世代進め (3) へ戻る．

A.4 遺伝的オペレータ

GNP は遺伝的オペレータを使用して構造の最適化を行う．オペレータは選択（Selection），交叉（Crossover），突然変異（Mutation）を使用する．以下，上記オペレータを本論文で使用しているものに絞り説明する．

A.4.1 選択

交叉や突然変異の実行個体を選択する際，ランク選択により適合度の良い個体を優先的に選択する．このように集団に淘汰圧をかけることによって，適合度の良い個体の遺伝子が次世代へ受け継がれる．以下にランク選択のフローを示す．

ランク選択

- (1) 個体群に対し適合度を基準にランク付けを行う．
- (2) 各個体のランクの高さに比例した選択確率を計算する．
- (3) 選択確率に従って個体を選択する．

またランク選択から交叉，突然変異により生成される個体の他に，その世代におけるランクの高い個体を一定数そのまま次世代へ残すエリート保存選択も使用している．

A.4.2 交叉

交叉は 2 個の親個体同士で部分プログラムを交換することにより実現する．本論文では Fig. A.3 に示すように一様交叉を使用している．以下にそのフローを示す．

一様交叉

- (1) 選択法により 2 個の親個体を選択する．
- (2) 交叉率 P_c で親個体のノードを選択する．
- (3) 親個体間で選択されたノードとその有向枝を交換する．

Fig. A.3 では親個体間の対応するノードラベルが同じとなるため，見かけ上は有向枝の交換となる．

A.4.3 突然変異

突然変異は 1 個体の遺伝子をランダムに変更することにより実現する．本論文では Fig. A.4 に示すように有向枝の突然変異を使用している．以下にそのフローを示す．

有向枝の突然変異

- (1) 選択法により 1 個の親個体を選択する．
- (2) 突然変異率 P_m で親個体の有向枝を選択する．
- (3) 選択された有向枝の接続先をランダムに変更する．

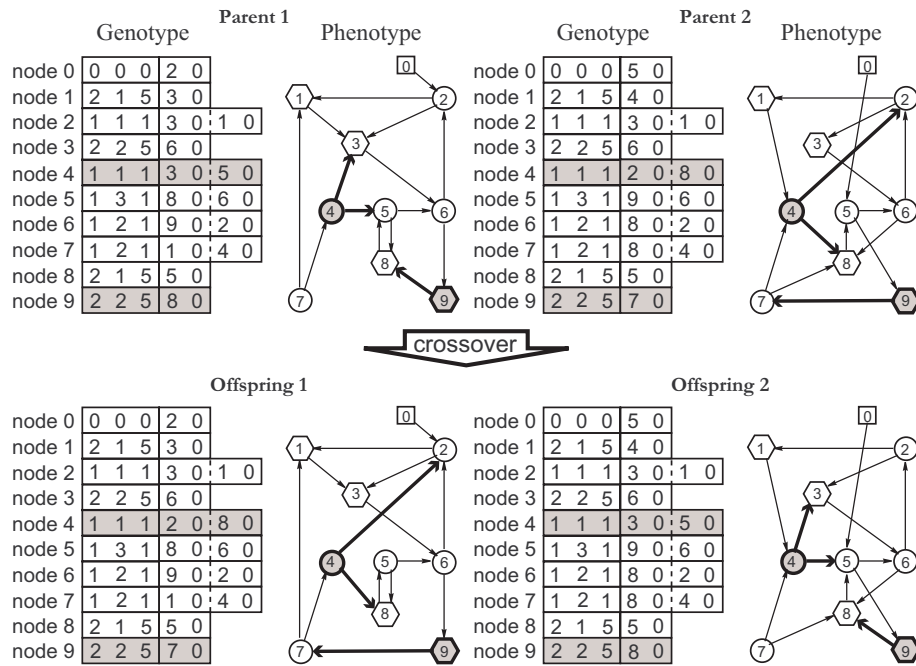


Fig. A.3 Uniform Crossover of GNP

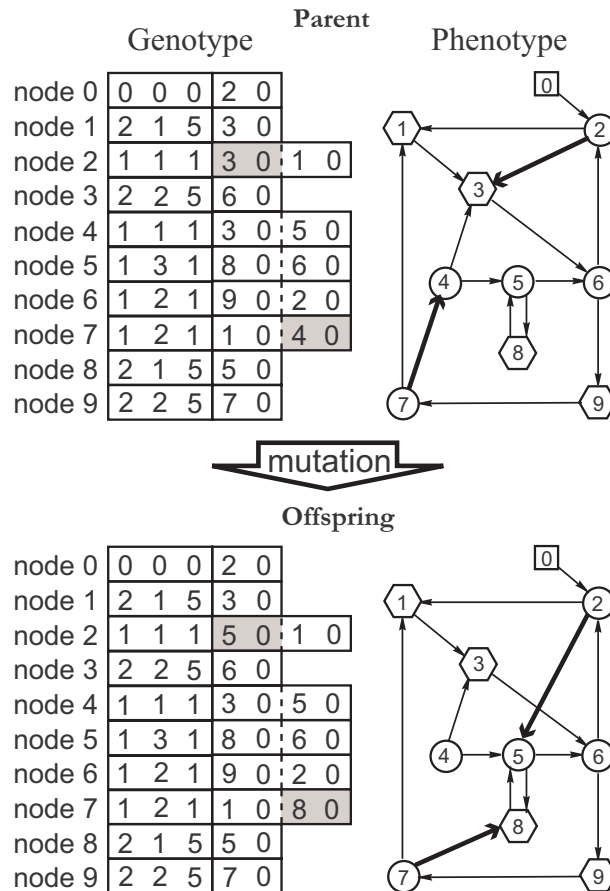


Fig. A.4 Branch Mutation of GNP

付 録 B

エレベータ群管理システム（EGSCS）

ここでは，エレベータ群管理システム（Elevator Group Supervisory Control System: EGSCS）[15, 16, 17] について，その目的，概要およびシミュレーション技術等を述べる．

B.1 エレベータ群管理の目的

EGSCS はビル内を移動する乗客を効率的に輸送するために，群管理コントローラによって複数のエレベータの運行管理を行なうシステムである．効率的な輸送とは，乗客を待たせることなく速やかに行先階まで移動させることであり，この間の乗客の待ち時間の最小化が EGSCS の主たる目的である．

群管理システムでは，その目的を達成するために（１）呼び割当て機能および（２）交通流検出機能に関する研究開発が進められてきた．呼び割当て機能はエレベータホールで発生したホール呼びをエレベータの運行状況および交通流のパターン等を考慮して適切なエレベータへ割当てする機能であり，EGSCS の性能を評価する上で核となる機能である．一方，交通流検出機能は時々刻々と変化するビル内交通流を検知し，現在の交通流のパターンを把握する機能であり，交通流毎に用意した呼び割当て機能との組合せによって効率的な群管理を実現する．

B.2 システムの概要

ここでは，現在最も一般的な UP/DOWN ボタン方式のエレベータシステムを前提とし，EGSCS の基本構成，ビル階床および交通流の定義，群管理指標等について述べる．

B.2.1 基本構成

Fig. B.1 に一般的な UP/DOWN ボタン方式の EGSCS の概要を示す．システムはエレベータ，エレベータホール（ビル階床），ホール呼びボタン，かご呼びボタン，群管理コントローラより構成される．エレベータは各エレベータホールに停止する際に乗客の乗降を行ない，登録呼び情報に従ってビル階床間を移動して乗客を輸送する．ホール呼びボタンは各階床のエレベータホールに設置され，乗客が行先階方向のボタン（UP/DOWN）を押すと，ホール呼びを受け取ったコントローラは呼び割当て機能により適切なエレベータへ呼びの割当てを行う．各エレベータ内にはかご呼びボタンが設置され，乗客は乗車後に行き先階のかご呼びボタンを押すとその階床の登録呼び情報が更新される．エレベータの運行ルールは基本的に登録呼び情報に従っており，移動方向と同じ方向のホール呼びおよびかご呼びから優先的にサービスを行い，それらのサービスが終了するまで折り返さない．また呼びの割当てのタイミングとして（１）ホール呼び発生後，即時に割当てる即時割当て方式（２）一定時間経過後に割当てる非即時割当て方式（３）割当て後のエレベータ変更を許す割当て変更方式等が存在するが，実装の容易さ等の理由から現在では即時割当て方式が一般的である．

B.2.2 ビル階床および交通流

EGSCS のビル階床は基準階と一般階の２種類に分類できる．基準階はビル内で最も交通量が多い階床（主に１階）であり，一般階は基準階以外の階床を指す．また一般階の階床間の交通量は全て等しいものとする．乗客の移動の大半を占めるのが（基準階 ↔ 一般階）の移動であり，この乗客へのサービス状況によって群管理の性能がほぼ決定される．

上記階床間の乗客移動の比率により，交通流が規定される．実際の交通流には様々なものがあるが，それらは概して平常時（Regular Time），アップピーク時（Up-peak Time）およびダウンピーク時（Down-peak Time）の３種類に分類できる．Table B.1 に（基準階 → 一般階）（一般階 → 基準階）および（一般階 → 一般階）の移動乗客数の比を示す．Table B.1 において，例えば平常時では（基準階 → 一般階）：（一般階 → 基準階）：（一般階 → 一般階）＝ 5 : 5 : 1 となる．

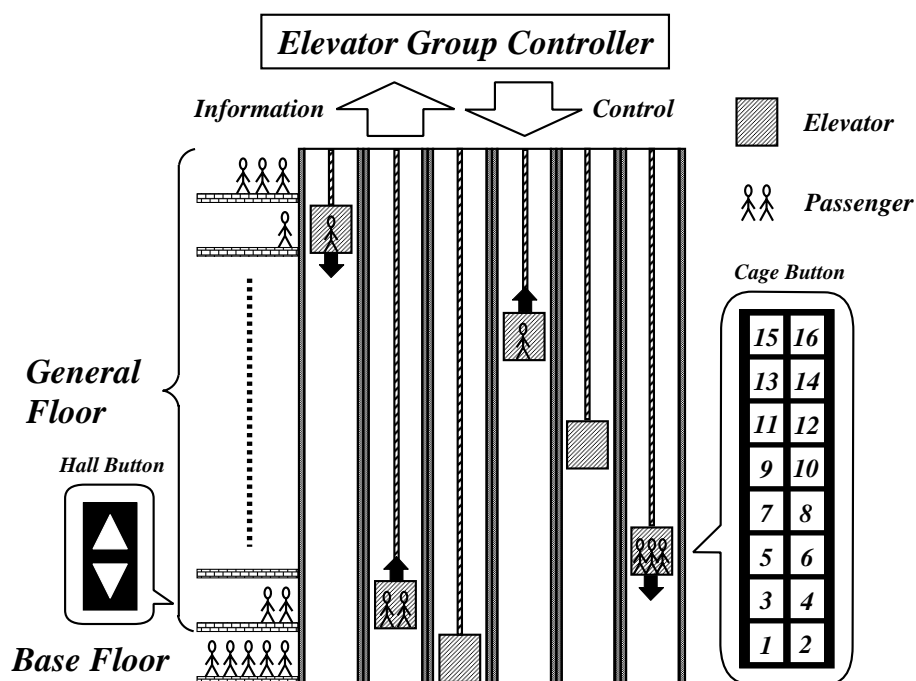


Fig. B.1 Outline of EGSCS

Table B.1 Ratio of Passengers of Each Traffic Flow

	Regular		Up-peak		Down-peak	
	Base	Gen.	Base	Gen.	Base	Gen.
Base Floor	—	5	—	19	—	2
General Floor	5	1	1	2	19	1

B.2.3 群管理の性能評価の指標

前述のように, EGSCS の群管理目標は全ての乗客の待ち時間を最小化することである. 群管理目標に基づく性能評価の指標として, 以下の 5 種類が挙げられる.

- 平均待ち時間 (AWT) 全乗客の待ち時間の平均であり, 最も標準的な指標である.
- 平均乗車時間 (ATT) 全乗客のエレベータ乗車時間の平均であり, 特に後述の行先階登録方式のエレベータシステムでは標準的に用いられる. また (AWT+ATT) を平均サービス完了時間 (AST) と呼ぶ.
- 最大待ち時間 (MWT) 全乗客の待ち時間の最大値であり, これは “全乗客の待ち時間の最小化” という EGSCS の群管理目標に基づく.
- 長待ち乗客率 (LWP) 全乗客中で待ち時間が 60 [s] を超えた人数の割合 [%] であり, MWT と同様に長待ち乗客の解消に対する群管理性能を評価する.
- 運行コスト (RNC) エレベータの移動距離や停止回数によって消費されたエネルギーコストに基づく指標である. 最近の群管理では待ち時間の最小化が重視されるため, 考慮されない場合が多い (本研究でも考慮しない).

B.3 エレベータシミュレータによる評価

本研究では, 提案手法の評価をエレベータシミュレータを用いたシミュレーションにより行う. EGSCS のシミュレータによる評価は計算機の性能向上と共に主流となり, 現在では各企業とも独自に開発したシミュレータにより研究開発を進めている [18, 71, 72].

本研究で使用するシミュレータは Microsoft Visual C++ 6.0 を開発環境とする C++ 言語を使用し, オブジェクト指向プログラミング (OOP) により構築した. OOP によって実際のエレベータシステムの細部に至る挙動の再現が可能となった. したがって本シミュレータによる結果は実用化に際して十分な信頼性を保証する.

シミュレータ内の事象は0.1秒毎の離散事象としてモデル化され、乗客の出現はポアソン分布に基づく確率的事象として扱う。またO/D表と呼ばれる乗客の出現階と行先階の分布を規定したデータを読み込むことで、前述の様々な交通流を実現可能となる。そしてビル階床数やエレベータ数（シャフト数）、エレベータの速度や加速度、乗客密度（単位時間当たりの乗客発生数）といったEGSCSのスペックを決定するパラメータが概ね設定可能となっている。Table B.2に本シミュレータで設定可能なパラメータを示す。

Table B.2 Available Parameters for Setting Specification of Elevator Simulator

Building Parameters	Elevator Parameters	Passenger Parameters
Number of Floors	Max. Velocity	Passenger Density
Number of Elevators (Shafts)	Max. Acceleration	Traffic Flow (O/D Table)
Floor Distance	Jerk (Acceleration Rate)	
	Cage Capacity	
	Time for Opening Door	
	Time for Closing Door	
	Time for Riding	

付 録 C

実数値 GA による ノードパラメータの最適化

ここでは, GNP のノードパラメータの最適化アルゴリズムについて述べる. 最適化アルゴリズムは実数値 GA (Real-coded Genetic Algorithm) [57] の交叉, 突然変異オペレータによりノードパラメータを最適化する.

交叉アルゴリズム

- (1) 選択法により 2 個の親個体を選択する.
- (2) 交叉率 P_c^{np} で選択した親個体のノードパラメータ (r_1, r_2) を選択する.
- (3) ランダムに発生させた変数 $a (\in [0, 1])$ を用いて新しいノードパラメータ (r'_1, r'_2) を Eq. (C.1) により求める.

$$\begin{cases} r'_1 = (1 - a) \cdot r_1 + a \cdot r_2 \\ r'_2 = a \cdot r_1 + (1 - a) \cdot r_2 \end{cases} \quad (\text{C.1})$$

突然変異アルゴリズム

- (1) 選択法により 1 個の親個体を選択する.
- (2) 突然変異率 P_m^{np} で親個体のノードパラメータ r を選択する.
- (3) r を $[0, 1]$ の範囲でランダムに変更する.

ノードパラメータが離散値をとる場合は, 計算後のパラメータを適切に丸める処理を行う.

付 録 D

時間帯における GNP 選択の重要性

ここでは、第 3 章 Table 3.5 で示される各時間帯における GNP 選択の重要性について議論する。Table 3.5 による切替え方式との比較を行うため、テストシミュレーションの全時間帯に対し Table 3.3 の *Case 1* ~ *Case 6* の GNP をそれぞれ単独で使用するシミュレーションを実行した。Fig. D.1 に、シミュレーションの結果得られた各 *Case* の GNP の適合度 (Eq. (3.3) により計算) の平均値を各時間帯について示す。図中の太線は Table 3.5 による結果を表している。結果は全て乱数系列の異なる 1000 試行の平均である。

Fig. D.1 が示すとおり、各 *Case* の GNP を単独で使用した場合、Table 3.5 の切替え方式と比べて適合度値は同等か劣るという結果を得た。特に、交通流の変化が激しい(c) 昼食時には各 *Case* とも性能が悪化している。また (a) 出勤時: *Case 4*, (b) 勤務時: *Case 1*, (d) 退勤時: *Case 5*, *Case 6* のように各時間帯の交通流に近い条件で最適化を行った GNP を使用したケースが良い性能を示しており、一方で *Case 3*, *Case 4* は (a) 出勤時以外では大きく性能が悪化している。以上の考察から交通流が動的に変化する条件下では、その変化に追従した適切な GNP の選択が重要となる。

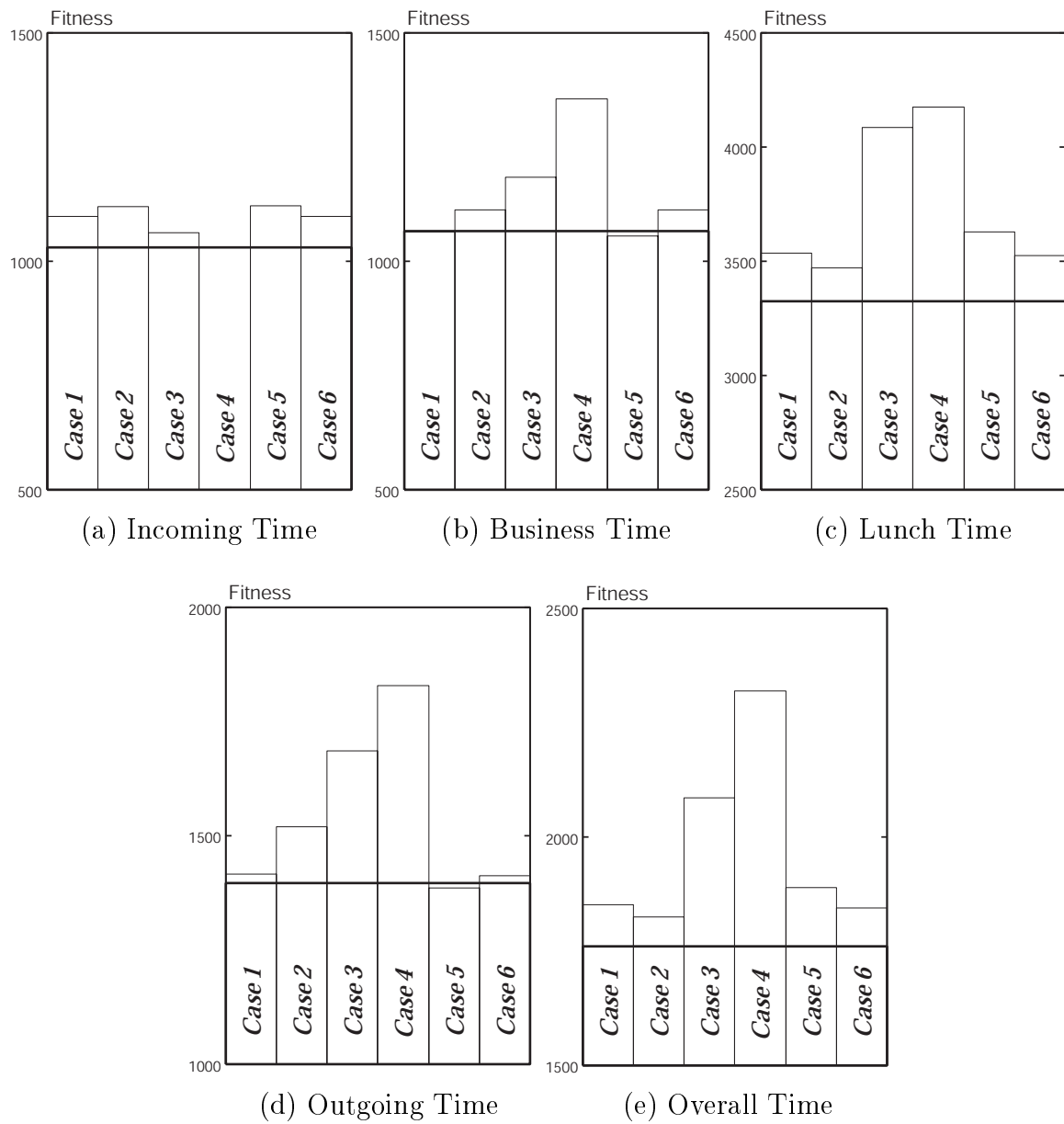


Fig. D.1 Comparisons of Fitness when using One GNP Optimized for a Special Traffic Flow

付 録 E

重要度指標付き GNP による機能切替え

ここでは、重要度指標付き GNP[60] による機能切替えのアルゴリズムについて述べる。

E.1 基本概念

一般に人間の脳は異なる処理系を持つ機能局在構成をしている [73, 74]。また現実の問題もその多くが複雑な構造をしており、人間がそれを解く場合、いくつかの部分的な問題として分割しそれぞれを異なる処理系により解決する。このように複雑な問題を部分的な問題の集合として捉え、異なる処理系によって解決する方式は工学的分野においても重要な知見とされ、分散人工知能等へ発展させた研究が行われている [75]。

一方、脳の機能局在構成のニューラルネットワーク (NNs) による実現を試みる研究も行われており [76, 77]、均一構造のネットワークの活性化される部分を制御することにより機能局在構成を実現している。しかし GNP のような進化型の手法では均一構造内に機能局在構成を実現することは難しい。そこで新たな機能局在モデルとして、部分的な問題 (タスク) に対応した処理系を GNP によって適宜切替えることにより機能局在構成を実現する方式を提案する。本方式では、Fig. E.1 に示す異なる機能を持つ $|M|$ (M : Sub-GNP の添字 m の集合, $m \in M$) 個の Sub-GNP を問題の状況に応じて Switch-GNP によって使い分けることで高い処理能力を目指す。Sub-GNP は予め特定の条件 (部分的に分割されたタスク) に対し進化を行い、Switch-GNP はそれらの Sub-GNP を使用して進化を行い、適切な切替えルールを獲得する。

E.2 重要度指標による切替えアルゴリズム

Switch-GNP の切替えルールは、重要度指標付き GNP のアルゴリズムにより実現する。重要度指標付き GNP では判定ノードの判定変数 J_l ($l \in L$) に処理 P_n ($n \in N$) の重要度指標 $IMX(J_l, P_n)$ を設定し、処理ノードではそこに至るまでの判定変数に該当する重要度指標を使用して処理 P_n の強度 $S(P_n)$ を Eq. (E.1) により計算する。

$$S(P_n) = \sum_{l \in L'} IMX(J_l, P_n) \quad (E.1)$$

ここで

- L' : 処理ノードに至る判定ノードの判定変数の添字の集合
- L : 判定ノードの判定変数の添字の集合
- N : 処理の種類の添字の集合

そして、処理ノードでは Eq. (E.2) により強度 $S(P_n)$ が最大となる処理 P_{opt} を選択する。

$$P_{opt} = \arg \max_{n \in N} S(P_n) \quad (E.2)$$

以上の重要度指標計算を行う機構を Switch-GNP に設ける。また Sub-GNP 数 $|M|$ が処理の数 $|N|$ に対応する。したがって Switch-GNP では、判定ノードの遷移によって得られる重要度指標をノード遷移に伴い加算し、得られた処理強度が最大となる Sub-GNP への切替え命令を実行する。

PROBLEM

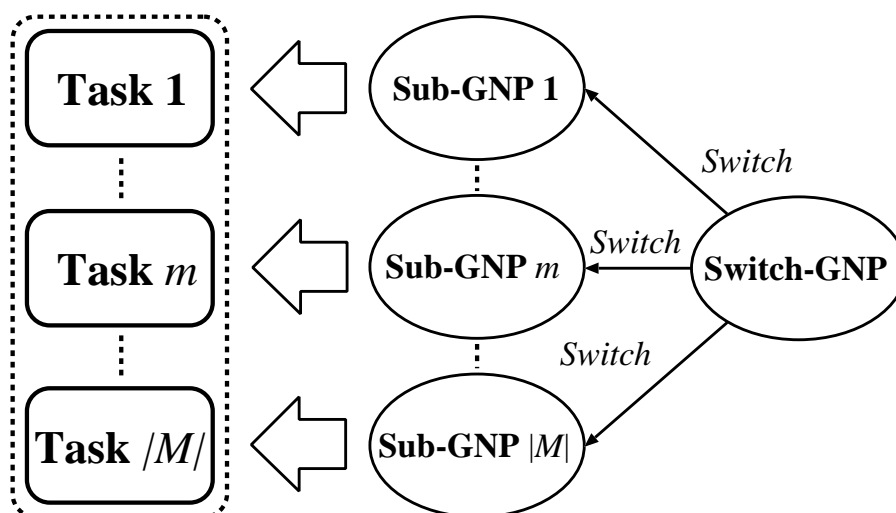


Fig. E.1 Basic Structure of Functional Localization by GNP

付 録 F

行先階登録方式エレベータシステム（DFGES）

ここでは、行先階登録方式エレベータシステム（Destination Floor Guidance Elevator System: DFGES）[19, 46, 63, 64] について述べる。DFGES は乗客がエレベータホールで行先階呼びを登録するシステムであり、近年、従来型 UP/DOWN ボタン方式のエレベータシステムに替わる新しいエレベータシステムとして普及が進められている。

DFGES は従来型エレベータシステムと比較して、以下の点で輸送効率の向上が期待できる。

- (1) エレベータホールではホール呼びと同時に行先階呼び（従来システムにおけるかご呼び）が入力されるため、これが割当てエレベータ決定に有益な入力情報となる。
- (2) 行先階が既知となるため、同一階床を行先階とする乗客を極力同じエレベータへ乗車させることができ、エレベータの停止回数を減少させることができる。

以下、システムの概要およびその群管理方式について述べる。

F.1 システムの概要

DFGES の概要を Fig. F.1 に示す。各階床のエレベータホールには図中に示す行先階登録ボタンが設置されており、乗客はホールへ出現後、自分の行先階のボタンを押す。このボタン入力によってホール呼び情報と行先階呼び情報が同時に入力される。コントローラは発生呼びを割当てるエレベータを決定し、そのエレベータ ID を押された行先階ボタンの隣に表示する。乗客は表示されたエレベータの扉の前に移動し、その到着後に乗車

する．したがって従来型エレベータシステムでは呼びを割当てられたエレベータ以外のエレベータへの乗車も可能であったが，DFGESでは基本的に割当てられたエレベータへの乗車しか行わない．DFGESでは従来システムにおけるかご呼び登録による不確定な要素が解消されるため，到着予想時間や予想乗車率といったエレベータの評価指標の精度向上を図ることができる．

F.2 行先階登録方式エレベータの群管理方式

DFGESの群管理方式として，従来型エレベータシステムと同様に Eq. (1.1) のエレベータ評価関数による呼び割当て方式が一般に用いられている．また DFGES の特徴として，(1) 発生呼びの予想サービス完了時間（ホール呼びへの到着予想時間 + 行先階呼びへの予想移動時間）が得られること (2) エレベータの登録呼び数に制限を設けることにより停止回数の減少を図れることが挙げられ，文献 [46] ではこれら进行评估する群管理方式が提案され，群管理性能の向上を達成している．

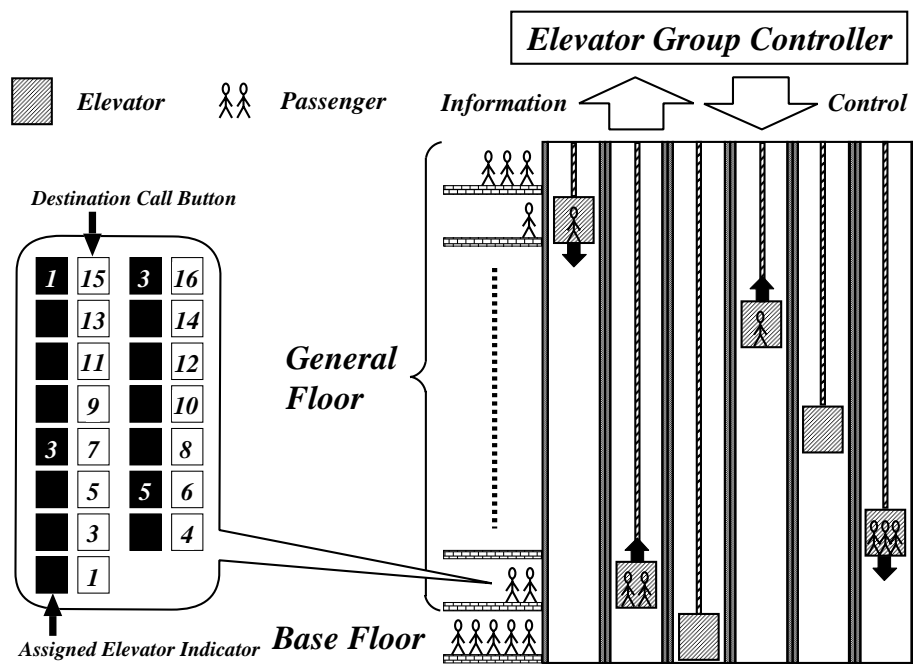


Fig. F.1 Outline of DFGEs

付 録 G

ダブルデッキエレベータシステム (DDES)

ここでは、ダブルデッキエレベータシステム (Double Deck Elevator System: DDES) [19, 48, 49, 50, 61] について述べる。DDES はエレベータの移動スペースであるエレベータシャフト内に通常のエレベータ (以降、かごと表記) を垂直に 2 台連結し設置したシステムである。ここで、従来の 1 シャフト内に 1 台のかごが設置されたシステムをシングルデッキエレベータシステム (SDES) と呼ぶことにする。

DDES は 1 シャフト当たりの輸送可能人数は 2 倍となり、輸送効率の向上およびビル内エレベータ占有面積の減少が期待できる。しかし DDES 特有の挙動や運行上の制約、乗客の乗り心地等の問題が生じるため、その群管理は SDES に比べ困難となる。

以下、システムの概要、DDES 特有の運行モードおよび挙動、群管理方式そして付録 F で説明した行先階登録方式の導入の可能性について述べる。

G.1 システムの概要

Fig. G.1 に DDES の概要を示す。DDES は 2 台のかごが連結された構造を持つため、特に基準階のエレベータホールの構造が SDES と大きく異なる。乗客は基準階から乗る場合、エスカレータ等によって自分が乗るべきかごが到着する階床 (上基準階 / 下基準階) へ誘導される。一般階での利用に際しては基本的に SDES と同様であるが、乗客は自分がどちらのかご (上かご / 下かご) に乗車するかはわからず、基準階への移動の場合は到着する基準階を選択不可となる¹。また最近ではカメラやモニタ、音声アナウンス等を通じて他方のかごの状況を把握できるようになっている。

¹ダブル運転の場合は、乗車かごおよび降車時の基準階の種類は特定可能である。

DDES は古くは 1930 年代より一部のビルに導入が行われてきたが、その実装および運用の難しさから近年まで普及には至らなかった。しかしエレベータ技術の発展に伴い DDES の有効性が見直され、現在高層ビルを中心にその普及が進められている。また DDES は階床間距離がフロア毎に異なるビルには導入が困難とされてきたが、近年かご間の距離を調節可能なエレベータが開発され [61]、それによって DDES の導入に一層拍車がかけられている。

G.2 運行モード

DDES には複数の運行モードが考案されており [19]、交通状況に合わせて適切な運行モードが使用される。以下に各運行モードについて説明する。

【ダブル運転】 上かごは偶数階、下かごは奇数階にサービスを行う。単純にかごが停止可能な階床数が半分となるため、常に両方のかごに乗客を乗せている場合には SDES に比べ効率的な乗客の輸送が可能となる。しかし、一般階間の移動を行う乗客へのサービスは想定していないため、通常はアップピーク時に使用される。

【セミダブル運転】 基準階のサービスのみ上かごは偶数階行き、下かごは奇数階行きと乗客を分けるが、一般階の乗客はどのかごへも乗車が可能である。特に一般階からの乗客が多い交通流（平常時、ダウンピーク時）に使用される。セミダブル運転ではダブルデッキ特有の挙動が発生しやすく、その群管理はダブル運転よりも困難となる。

【シングル運転】 一方のかごの機能を停止して、通常の SDES と同様の運行を行う。乗客の利用が少ない時間帯（早朝、深夜）にかご 1 台でもサービスが充分可能な場合、運行コストの節減を目的として使用される。

G.3 ダブルデッキエレベータ特有の挙動

DDES は SDES には見られない特有の挙動が存在する。それらは効率的な群管理に関わる重要な知見であり、その考慮により群管理性能の向上を期待できる。

【1. 片かごサービス】 Fig. G.2 (a) のように、あるかごが停止してサービスを行っている間、他方のかごはサービスを行わず停止している状況を指す。このとき、サービスを行わないかご内の乗客に対しては他方のかごがサービス中であることが知られる。このような状況は輸送効率の悪化だけでなく、乗客に心理的負担を与えることとなる（乗り心地の悪化）。乗客の乗り心地の改善はDDESの群管理の際に新たに考慮されるようになった群管理目標である。

【2. 両かごサービス】 Fig. G.2 (b) のように、1回の停止で上下かごともサービスを与える状況を指す。片かごサービスとは逆に、両かごサービスの回数が増えるように群管理を行うことで、輸送効率と乗客の乗り心地の改善を達成できる。

【3. 同一行先階の乗客の分離乗車】 Fig. G.2 (c) のように、同一の行先階へ移動する乗客が上下別のかごに乗車している状況を指す。この場合エレベータはその行先階に2回停止することになり、他の乗客へのサービスに悪影響を及ぼす結果となる。この状況は、行先階登録方式の導入によって予め乗客の行先階を考慮した呼びの割当てを行うことで解消できる。

【4. 運行負荷の偏り】 Fig. G.2 (d) のように、上下かごの乗車人数または登録呼び数に偏りが生じることによって、負荷が小さいかごのサービスが負荷が大きいかごのサービスを悪化させてしまう状況を指す。例えば、登録呼びを1個持つ上かごのサービスに10 [s] を要した場合、登録呼びを10個持つ下かごの待ち時間は少なくとも100 [s] 悪化することになる。したがって上下かごの運行負荷の平均化によって、輸送効率の向上を図ることができる。

【5. ゼロコストのサービス】 Fig. G.2 (e) のように、乗客が行先階（7F）へ移動する際にどちらのエレベータシャフトのかごに乗りしても両かごサービスが可能となり、連結するかごのサービスに対する時間的コストの増加なしの輸送が可能となる状況を指す。このようなゼロコストのサービスは従来のUP/DOWNボタン方式のエレベータシステムでは難しく、行先階登録方式の導入により実現が可能となる。

G.4 ダブルデッキエレベータの群管理方式

最も古典的な DDES の群管理方式として、発生呼びを基本的に後かご²に割当て、後かごとの両かごサービスが可能な場合にもう一方のかご（先かご）へ以降の呼びを割当てる方式がある [48]。この方式では両かごサービスの回数が増え、効率的な群管理が実行できる。しかし交通流の激しい時間帯では先かごを効率良く使用できないため、近年の大規模ビルのエレベータ群管理へのニーズを満たすことは難しい。

近年では上記方式に替わり、AI 技術による群管理方式が提案されている [49, 62]。この方式では前後かごにかかわらず最適なかごへ呼びを割当て、その前後の階床で両かごサービスが可能な呼びがあれば適宜割当てる。その際システムへの入力情報として、待ち時間や移動時間の予測値、上下かごの運行負荷のバランス等の指標が考慮される。

G.5 行先階登録方式の導入の可能性

DDES 特有の挙動を解消し輸送効率の向上を計るため、行先階登録方式を導入する試みが検討されている [50]。その導入によるメリットは以下のように整理できる。

【基準階サービスの効率化】 Fig. G.3 のように乗客は基準階で行先階を登録するとその場で乗るべきかご ID が表示されるため、従来の偶数階 → 上かご、奇数階 → 下かごの乗り分けをする必要がなくなり、効率的な乗客へのサービスが期待できる。

【DDES 特有の挙動の解消】 発生ホール呼びおよび行先階呼びが全て既知となるため、それらを考慮することにより片かごサービス、同一行先階の乗客の分離乗車等の挙動を解消し効率的な群管理が可能となる。

【突出し階の必要性の解消】 UP/DOWN ボタン方式では最上階が奇数階である場合、下かごを最上階へ停止させるため、突出し階を余分に設ける必要があった。しかし行先階登録方式の導入により下かごの最上階へのサービスを禁止できるため、突出し階は必要がなくなりコストを節減できる。

² 上方向移動中の場合は下かご、下方向移動中の場合は上かごとなる。

上記のメリットは特にセミダブル運転時において効果が期待できる．またデメリットとしてはシステムの導入コストが挙げられ，それゆえ現時点では基準階のみへの導入を前提とした検討が進められている（本研究では，全階床に導入している）．

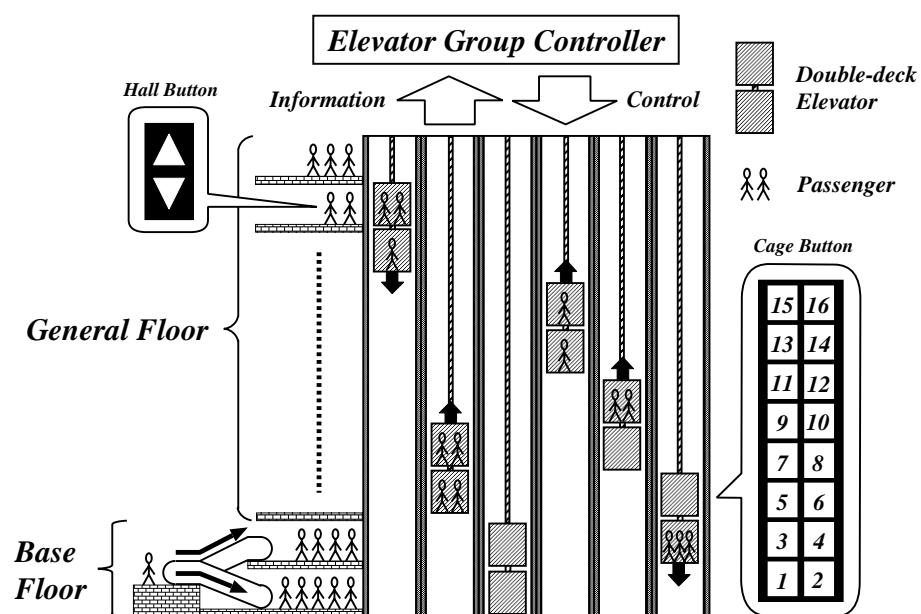


Fig. G.1 Outline of DDES

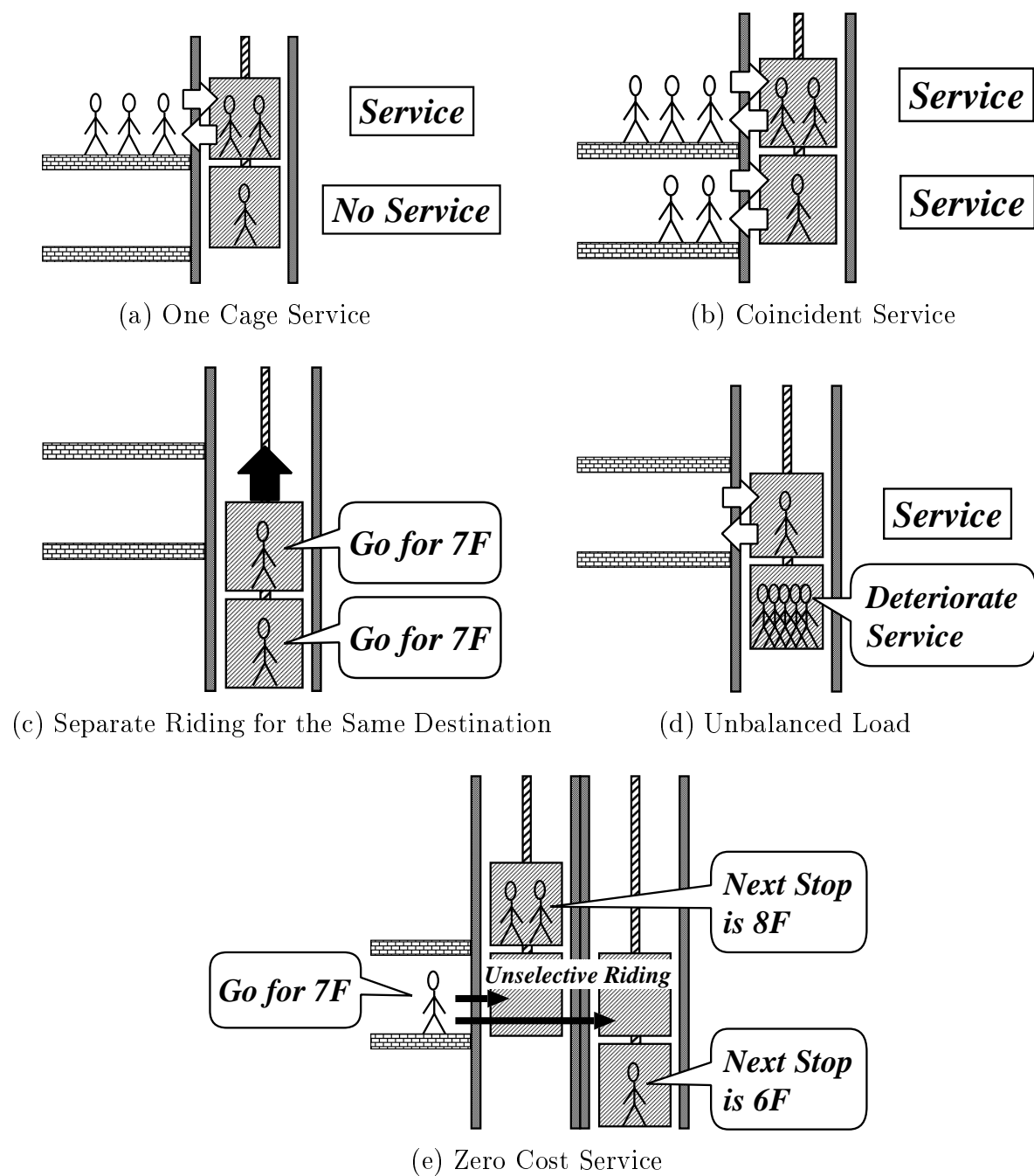


Fig. G.2 Specific Behaviors in DDES

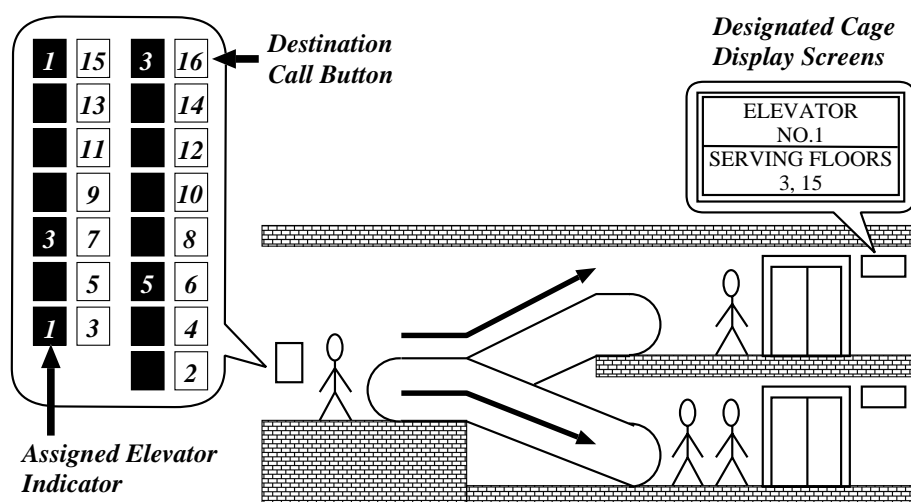


Fig. G.3 Outline of the Base Floors of DDES Introducing DFGES

謝辞

本論文をまとめるにあたり，多くの方々からのご支援と激励を頂きました．この場をかりて御礼を申し上げたいと思います．

始めに，九州大学における3年間および早稲田大学へ進学してからの3年間の合計6年間，研究に関する御指導を賜りました本学情報生産システム研究科の平澤宏太郎教授には，懇切丁寧な御指導に加え研究の面白さや奥深さを御教授して頂きました．私が研究を始め様々な面で成長できたことは，ひとえに先生の御指導の賜物だと感じています．心から御礼申し上げます．そして論文作成にあたり，本学情報生産システム研究科の玄光男教授，和多田淳三教授には研究全般に関する御指摘，御助言から論文の構成等に至るまで，多くの御指導を賜りました．ここに謹んで諸先生方に感謝の意を表します．

本研究を遂行する上で，フジテック（株）マルコンシャンドル氏には，エレベータ群管理に関する数々の御協力や御助言を頂きました．氏の御支援なくしては本研究を成し得ることはできませんでした．ここに心から感謝の意を申し上げます．

そのほか，6年間の研究活動の日々において研究の進め方，心構え等数々の御助言，御支援を賜った本学情報生産システム研究科の古月敬之助教授，九州大学大学院システム情報科学府の村田純一助教授にも厚く御礼申し上げます．また，共に研鑽を磨き研究に邁進した片桐広伸先輩，間普真吾君，笹川隆史君をはじめとする研究室の諸氏にも深く感謝致します．

最後になりましたが，本研究および6年間の研究活動を続けるにあたり，両親の理解と支援は何にも増して感謝しなければならないことであり，今後はこの6年間で培ったものを活かし，少しでも親孝行に役立てていきたいと思っています．本当にありがとうございました．

研究業績

学術誌原著論文

- (1) 江口徹, 周金, 平澤宏太郎, 古月敬之, マルコンシャンドル, “ランキング処理とノード関数最適化を考慮した Genetic Network Programming によるエレベータ群管理システム”, 計測自動制御学会論文集, Vol.42, No.3, 2006.
- (2) Toru Eguchi, J. Zhou, S. Eto, K. Hirasawa, J. Hu and S. Markon, “Elevator Group Supervisory Control System using Genetic Network Programming with Functional Localization”, Journal of Advanced Computational Intelligence and Intelligent Informatics, Vol.10, 2006.
- (3) Toru Eguchi, K. Hirasawa, J. Hu and N. Ota, “A Study of Evolutionary Models Based on Symbiosis”, IEEE Trans. on System, Man and Cybernetics, Part B, Vol.35, No.1, pp.179-193, 2006.
- (4) 江口徹, 周金, 平澤宏太郎, 古月敬之, マルコンシャンドル, “Genetic Network Programming によるエレベータ群管理システムの基礎検討”, 電気学会論文誌 C, Vol.125, No.7, pp.1055-1062, 2005.
- (5) 江口徹, 平澤宏太郎, 古月敬之, “マルチエージェントシステムの共生進化モデルの構築”, 情報処理学会論文誌：数理モデル化と応用, Vol.45, No. SIG 2 (TOM 10), pp.144-156, 2004.
- (6) S. Eto, Toru Eguchi, K. Hirasawa and J. Hu, “Evolutional Acquisition of Communication Between Agents Using Genetic Network Programming”, Machine Intelligence & Robotic Control, Vol.5, No.1, pp.25-32 2003.

- (7) 江口徹, 平澤宏太郎, 古月敬之, 村田純一, “Genetic Network Programming を用いた共生学習進化型マルチエージェントシステム”, 電気学会論文誌 C, Vol.123, No.3, pp.517-526, 2003.
- (8) 平澤宏太郎, 中西賢精, 江口徹, 古月敬之, “共生学習進化型マルチエージェントシステムとその応用”, 電気学会論文誌 C, Vol.123, No.1, pp.67-74, 2003.

査読のある国際会議・シンポジウム論文

- (1) Toru Eguchi, K. Hirasawa, J. Hu and S. Markon, “Elevator Group Supervisory Control System using Genetic Network Programming with Functional Localization”, in Proc. of IEEE Congress on Evolutionary Computation 2005, pp.328-335, Edinburgh, 2005.
- (2) J. Zhou, Toru Eguchi, K. Hirasawa, J. Hu and S. Markon, “Elevator Group Supervisory Control System using Genetic Network Programming with Reinforcement Learning”, in Proc. of IEEE Congress on Evolutionary Computation 2005, pp.336-342, Edinburgh, 2005.
- (3) Toru Eguchi, K. Hirasawa, J. Hu and S. Markon, “Elevator Group Supervisory Control System Using Genetic Network Programming -Ranking Processing and Node Function Optimization-”, in Proc. of SICE Annual Conference 2005, pp.1-6, Okayama, 2005.
- (4) Toru Eguchi, K. Hirasawa, J. Hu and S. Markon, “Elevator Group Supervisory Control System using Genetic Network Programming”, in Proc. of IEEE Congress on Evolutionary Computation 2004, pp.1661-1667, Portland, 2004.
- (5) Toru Eguchi, K. Hirasawa and J. Hu, “Symbiotic Evolutional Models in Multiagent Systems”, in Proc. of IEEE Congress on Evolutionary Computation 2003, pp.739-746, Canberra, 2003.
- (6) S. Eto, Toru Eguchi, K. Hirasawa and J. Hu, “Evolutional acquisition of communication of GNP”, in Proc. of SICE Annual Conference 2003, pp.1215-1220, Fukui, 2003.

- (7) Toru Eguchi, K. Hirasawa, J. Hu and J. Murata, “Multiagent Systems with Symbiotic Learning and Evolution using Genetic Network Programming”, in Genetic and Evolutionary Computation Conference 2002 Late Breaking Papers, pp.130-137, New York, 2002.

講演

- (1) 江口徹, 平澤宏太郎, マルコンシャンドル, “Genetic Network Programming によるエレベータ群管理制御の最適化”, 日本機械学会技術講演会〔No.05-68〕昇降機・遊戯施設等の最近の技術と進歩 講演論文集, pp.17-20, 東京, 2006.
- (2) 江口徹, 平澤宏太郎, 古月敬之, マルコンシャンドル, “Genetic Network Programming によるエレベータ群管理システムの構築”, SICE システム・情報部門学術講演会 2004 講演論文集, pp.73-78, 浜松, 2004.
- (3) 江口徹, 平澤宏太郎, 古月敬之, マルコンシャンドル, “Genetic Network Programming によるエレベータ群管理システムの評価”, 平成 16 年電気学会電子・情報・システム部門大会 講演論文集, pp.526-531, 宇都宮, 2004.
- (4) 江口徹, 平澤宏太郎, 胡敬炉, “タイルワールドモデルを用いた共生学習進化型マルチエージェントシステム”, 第 9 回情報処理学会 MPS シンポジウム 進化的計算シンポジウム 2002 予稿集, 京都, 2003.
- (5) 江藤慎治, 平澤宏太郎, 胡敬炉, 江口徹, “コミュニケーションを考慮した GNP によるマルチエージェントシステム”, 第 21 回 SICE 九州支部学術講演会 講演論文集, pp-269-272, 大分, 2002.
- (6) 江口徹, 平澤宏太郎, 胡敬炉, “共生学習進化型マルチエージェントシステムとそのタイルワールドモデルへの適用”, SICE システム・情報部門学術講演会 2002 講演論文集, pp.207-212, 横浜, 2002.
- (7) 江口徹, 平澤宏太郎, 胡敬炉, 村田純一, “共生学習進化型マルチエージェントシステムにおけるパレート均衡解の探索”, SICE システム・情報部門学術講演会 2001 講演論文集, pp.85-90, 宮崎, 2001.
- (8) 江口徹, 平澤宏太郎, 胡敬炉, 村田純一, “GNP を用いた共生学習進化型マルチエージェントシステム”, 計測自動制御学会学術講演会 2001 講演論文集, 名古屋, 2001.