

## 暫定理想点法を用いた経路生成手法の提案とドローン宅配によるコスト削減効果の定量的評価

大曲, 宏樹

<https://doi.org/10.15017/4060172>

---

出版情報：九州大学, 2019, 博士（工学）, 課程博士  
バージョン：  
権利関係：



---

令和元年度 博士論文

暫定理想点法を用いた経路生成手法の提案と  
ドローン宅配によるコスト削減効果の定量的評価

---

大曲 宏樹

Hiroki Omagari

2020 年 1 月

九州大学大学院工学府 航空宇宙工学専攻

指導教員 東野 伸一郎 准教授

飛行力学研究室



KYUSHU UNIVERSITY

# 概要

ネット通販の普及と流通網の多様化により、誰もが時間や場所を問わず手軽にショッピングを楽しむ時代が到来して久しい。今や日本国内の物流業界全体における宅配便取扱個数は 42 億個以上といわれ、今後もこの増加傾向は継続すると見込まれている。一方、物流業界では、宅配ドライバーの担い手不足が深刻化しており、従業員の高齢化も進んでいる。この傾向は、過疎化が進行する地方では特に顕著であり、所謂「宅配難民」と呼ばれる人々が増加する要因となっている。その解決策として近年注目されているのが、AI や IoT などの先進技術を駆使した物流業務の効率化である。業界では、これを「物流革命」と呼んでいる。ドローン宅配サービスはその一翼を担う存在として注目されている。本研究では、このドローン宅配サービスの実現に向けた取り組みの一環として、その最適な宅配経路を生成するための計算手法の提案と、そのコスト削減効果の定量的評価を目的とする。

一般的に、宅配経路問題は制約付き多目的最適化問題として定式化することができる。この問題では、異なる目的関数同士が互いにトレードオフの関係となるため、意思決定者は自身の選好情報に最も合致する妥協点、すなわち選好解を 1 つだけ選択しなければならない。選好解を得るための手法としてこれまで数多くの解法が提案されてきたが、制約付き多目的最適化問題における意思決定者の選好解をただ 1 つだけ効率的に生成する手法についてはこれまでほとんど提案されていない。宅配経路問題などの実問題の解法に至っては皆無といえるのが現状である。

本研究では、制約条件の満足化と多目的最適化による選好解の生成を同時に実現する遺伝的アルゴリズムをベースとした「暫定理想点法」を提案する。本手法の有用性については、GO 問題及び DTLZ 問題と呼ばれる 2 つのベンチマーク問題を用いて示す。一方、本手法を用いて宅配経路問題を解く際には、2 つの問題点に留意する必要がある。1 つ目は、この提案手法の評価関数値が無限大となり、解同士の優劣関係を判定できなくなる場合があることである。2 つ目は、解探索が初期段階で局所的最適解に早期収束してしまうことである。

これらの問題点を解決する新たな手法として、変換係数を用いて全ての目的関数を宅配コストに変換すると同時に、解探索にタブーサーチを組み合わせることを提案する。この改良手法の有用性については、ベンチマーク問題である TSPLIB と実際の宅配経路問題を解くことによって示す。この改良手法を用いてドローン宅配サービスのコスト削減効果に関する定量的評価を行った結果、その知見としてトラックに対するドローンの相対コスト比をできるだけ小さくすることや、複数の宅配場所を連続して宅配できるドローンを使用することが有効であることを示した。また、その効果を最大限享受できる宅配スタイルは、相対コスト比に応じて変化することや、ドローンの性能諸元の 1 つを単独で向上させてもコスト削減効果の向上に寄与するとは限らないこと等を示した。

# 目次

## 第1章 序論

1.1	はじめに.....	1
1.2	研究背景.....	1
1.2.1	少子高齢化と人口減少社会.....	1
1.2.2	物流業界の現状.....	2
1.3	ドローン宅配への期待とその実現に向けた課題.....	5
1.4	ドローン宅配実現に向けた取り組み.....	11
1.4.1	国内事業.....	11
1.4.2	海外事業.....	12
1.5	本研究の目的.....	14
1.6	論文構成.....	14

## 第2章 最適化問題と解探索アルゴリズム

2.1	最適化問題.....	16
2.1.1	一般式.....	16
2.1.2	最適化問題の分類と宅配経路問題の特徴.....	16
2.1.3	大局的最適解と局所的最適解.....	17
2.1.4	多目的最適化問題の特徴.....	18
2.1.5	複雑性クラス.....	19
2.2	解探索アルゴリズム.....	21
2.2.1	メタヒューリスティクスの基本戦略.....	21
2.2.2	解探索の集中化と多様化.....	22
2.2.3	代表的なメタヒューリスティクス.....	22
2.2.4	近似解法の選択.....	26
2.3	多目的最適化手法.....	26

### 第 3 章 暫定理想点(PIP)法の提案

3.1	はじめに.....	28
3.2	PIP 法の解探索プロセス.....	28
3.3	PIP 法の評価方法.....	32
3.3.1	解表現.....	33
3.3.2	交叉・交配.....	33
3.3.3	突然変異.....	34
3.3.4	選択・淘汰.....	35
3.3.5	解探索フローチャート.....	35
3.4	GO 問題を用いた PIP 法の妥当性評価.....	36
3.5	DTLZ 問題を用いた PIP 法の妥当性評価.....	42

### 第 4 章 宅配経路問題の定式化

4.1	はじめに.....	47
4.2	解表現と遺伝的操作.....	50
4.2.1	Truck 宅配.....	50
4.2.2	Drone 宅配.....	51
4.2.3	Hybrid/M-Hybrid/Multi-Agent 宅配.....	51
4.3	問題設定.....	53
4.3.1	地図データの作成.....	53
4.3.2	事前計算.....	54
4.4	定式化.....	55
4.4.1	Truck 宅配.....	55
4.4.2	Drone 宅配.....	56
4.4.3	Hybrid/M-Hybrid/Multi-Agent 宅配.....	56
4.5	連続宅配.....	58
4.5.1	C-Drone 宅配.....	59
4.5.2	C-Hybrid/C-Multi-Agent 宅配.....	61

## 第5章 PIP法の改良

5.1	はじめに.....	65
5.2	従来型 PIP 法の問題点.....	65
5.2.1	DBZ 問題.....	65
5.2.2	IC 問題.....	66
5.3	改良型 PIP 法.....	67
5.3.1	変換係数による DBZ 問題の回避.....	68
5.3.2	タブーサーチによる IC 問題への対処.....	70
5.4	妥当性評価.....	72
5.4.1	TSPLIB への適用結果.....	72
5.4.2	DDP への適用結果.....	73
5.4.3	実行可能解の生成確率.....	75

## 第6章 計算結果と考察

6.1	はじめに.....	77
6.2	宅配経路生成.....	77
6.3	宅配コスト削減効率と計算条件の関係.....	85
6.3.1	相対コスト比.....	85
6.3.2	移動速度.....	87
6.3.3	離着陸時間.....	88
6.3.4	飛行可能時間.....	89
6.4	宅配経路とタスク割当の同時最適化問題.....	90
6.5	まとめ.....	92

## 第7章 結論

7.1	まとめ.....	93
7.2	今後の課題.....	94

Appendix A.....	95
Appendix B.....	100
参考文献.....	103
謝辭.....	110

# 目次

1.1	日本の人口推移.....	2
1.2	物流業界の事業分野別営業収入.....	3
1.3	宅配便取扱個数の推移.....	3
1.4	ドライバーの年間所得額の推移.....	4
1.5	ドライバーの年間労働時間の推移.....	4
1.6	年齢階級別就業者構成比.....	4
1.7	用途別のドローンの例.....	6
1.8	米国 UPS 社による Tandem Delivery の実証実験.....	7
1.9	無人航空機の飛行許可が必要となる空域.....	8
1.10	特別の許可を必要とする飛行方法.....	8
1.11	目視外領域におけるドローンの運行管理システムの構想図.....	9
1.12	ドローンハイウェイの構想図.....	10
1.13	物流用ドローンのポートシステム.....	10
1.14	Amazon Prime Air での使用が見込まれるドローン.....	13
2.1	大局的最適解と局所的最適解.....	18
2.2	2 目的最適化問題の解空間を表す概念図.....	18
2.3	複雑性クラスの関係図.....	21
2.4	GA の基本的な解探索フローチャート.....	23
3.1	実行可能解と実行不可能解の概念図.....	28
3.2	解探索第 2 段階の概念図.....	30
3.3	パレートフロント上に生成される選好解の概念図.....	31
3.4	バイナリー形式による実数の表現.....	33
3.5	交叉・交配による遺伝的操作.....	34
3.6	突然変異による遺伝的操作.....	35
3.7	PIP 法による解探索フローチャート.....	36
3.8	go01 への適用結果.....	37
3.9	go02 への適用結果.....	38

3.10	go03 への適用結果.....	39
3.11	go04 への適用結果.....	40
3.12	go05 への適用結果.....	41
3.13	go06 への適用結果.....	41
3.14	go07 への適用結果.....	41
3.15	go08 への適用結果.....	41
3.16	go09 への適用結果.....	41
3.17	go10 への適用結果.....	41
3.18	DTLZ1 の解探索結果.....	43
3.19	DTLZ3 の解探索結果.....	44
3.20	DTLZ7 の解探索結果.....	46
4.1	Truck 宅配の概念図.....	47
4.2	Drone 宅配の概念図.....	48
4.3	Hybrid 宅配の概念図.....	48
4.4	M-Hybrid 宅配の概念図.....	49
4.5	Multi-Agent 宅配の概念図.....	50
4.6	$T^i$ の概念図.....	50
4.7	$D^j$ の概念図.....	51
4.8	$T^i, D^{ij}$ 及び $u^{ij}$ の概念図.....	52
4.9	$D_a^{ij}$ の飛行経路の一例.....	52
4.10	宅配エリア.....	53
4.11	連続宅配の概念図.....	59
4.12	C-Drone 宅配経路の解構造.....	60
4.13	C-Hybrid 宅配経路の解構造.....	62
5.1	経路生成問題における「交叉・交配」の例.....	66
5.2	宅配コストの内訳.....	69
5.3	GA を用いて生成した最短トラック宅配経路.....	70
5.4	GA と GA+TS の比較.....	72
5.5	従来型及び改良型 PIP 法の適用結果.....	73
5.6	$C_{total}(\mathbf{x})$ に関する従来型 PIP 法と改良型 PIP 法の比較.....	75
5.7	実行可能解の生成確率の比較.....	76

6.1	Truck 宅配経路.....	79
6.2	Drone 宅配経路.....	80
6.3	C-Drone 宅配経路.....	81
6.4	Hybrid 宅配経路.....	82
6.5	C-Hybrid 宅配経路.....	83
6.6	M-Hybrid 宅配経路.....	84
6.7	宅配コスト削減率と $R$ の関係.....	85
6.8	コスト削減率とドローンの飛行速度の関係.....	87
6.9	コスト削減率とドローンの離着陸時間の関係.....	88
6.10	コスト削減率とドローンの飛行可能時間の関係.....	89
6.11	宅配コスト削減率と $R$ の関係.....	91
6.12	ドローンが宅配する $WP$ の割合と $R$ の関係.....	91
B.1	DTLZ1 のパレートフロンティア形状.....	101
B.2	DTLZ1 のパレートフロンティア形状.....	101
B.3	DTLZ1 のパレートフロンティア形状.....	102

# 表目次

1.1	道路貨物運送業の就業者数推移.....	4
1.2	荷物 1 個あたりのトラックの走行距離の比較.....	12
2.1	2つの多目的最適化手法の利点と欠点.....	27
3.1	PIP 法を GO 問題に適用した計算結果.....	36
5.1	計算条件.....	74
6.1	共通の計算条件.....	78
6.2	宅配スタイルごとの計算条件.....	78
6.3	各宅配経路の目的関数値と宅配コストの対比表.....	78
6.4	共通の計算条件.....	90
6.5	宅配スタイルごとの計算条件.....	90

# 記号表

$A$	: 使用可能なトラックの台数
$B$	: トラック 1 台あたりに搭載可能なドローンの台数
$CF_I$	: $F_I(x)$ の変換係数
$C_{ideal}$	: 暫定理想点座標
$C_{sol}(x)$	: 解点座標
$C_{total}$	: 合計宅配コスト
$DC$	: $WP$ 間の飛行距離コスト
$D_{is}$	: $C_{ideal}$ 及び $C_{sol}(x)$ 間のユークリッド距離
$D^j, D^{ij}$	: ドローンが宅配する $WP$ の集合
$E_n$	: 次世代に継承するエリート個体の数
$F(x)$	: 評価関数
$F_I(x)$	: $I$ 番目の目的関数
$F_{I_{pro}}$	: $F_I(x)$ の暫定最適値
$g_J(x)$	: $J$ 番目の制約条件
$g, g'$	: 世代数
$G_{max}$	: 解探索の最大世代数
$G_{tabu}$	: 解探索の停滞を判断する世代数
$h_{aij}$	: $D_a^{ij}$ 内に含まれる $WP$ の数
$I_{max}$	: 目的関数の数
$J_{max}$	: 制約条件の数
$L_d$	: デポの位置
$LD_{max}$	: 最大移動可能距離
$LT_{max}$	: 最大移動可能時間
$m$	: 制約条件の数
$M$	: 目的関数の数
$ND_j$	: Drone 宅配において $j$ 番目のドローンが宅配する $WP$ の数
$NH_i$	: Hybrid 宅配において $i$ 番目のトラックが宅配する $WP$ の数
$NT_i$	: Truck 宅配において $i$ 番目のトラックが宅配する $WP$ の数
$N_{WP}$	: $WP$ の総数
$p, p'$	: 解集合
$P_n$	: 個体数
$P(x)$	: ペナルティー関数
$R$	: トラックに対するドローンの相対コスト比
$s$	: $p$ もしくは $p'$ に含まれる解の個数
$S$	: 実行可能領域

$S'$	: $S$ の一部
$TC$	: $WP$ 間の移動距離コスト
$t_d$	: ドローンの飛行可能時間
$T^i$	: $i$ 番目のトラックの宅配経路
$TL$	: タブーリスト
$t_p$	: $WP$ におけるトラックの駐停車時間
$t_{re}$	: バッテリーの交換時間
$t_{tl}$	: ドローンの離着陸時間
$u^{ij}, u_a^{ij}$	: ドローンの飛行経路を決定する値もしくはその集合
$V_D$	: ドローンの飛行速度
$V_T$	: トラックの走行速度
$w_b^j, w_{a,b}^{ij}$	: $D_b^j$ もしくは $D_{a,b}^{ij}$ における荷物の重量
$W_d$	: ドローンの積載可能重量
$WP$	: デポもしくは宅配場所
$x, x'$	: 生成解
$x^*$	: 集合 $p$ 内における最良解
$x_{best}$	: 選好解
$x_F, x'_F, x''_F$	: 実行可能解
$x_F^*$	: 集合 $p$ 内における実行可能解の最良解
$x_G$	: 大域的最適解
$x_i$	: $i$ 番目の設計変数
$x_L$	: 局所的最適解
$x_{opt}$	: ベンチマーク問題における既知の最適値

# 第1章 序論

## 1.1 はじめに

近年、ドローンは軍事目的に限らず、インフラ点検や観測飛行、農薬散布、空撮、物資輸送など様々な領域でその利活用が拡大している<sup>[1]-[5]</sup>。また、ホビー用品としても幅広い年齢層で親しまれるようになった<sup>[6]</sup>。これは、電子機器やセンサ類の高度化、小型軽量化、低価格化が進んだことが大きな要因として挙げられる<sup>[7]-[9]</sup>。そんな中、ドローンを宅配サービスに活用しようとする試みが世界中で活発に行われている<sup>[10],[11]</sup>。これまでの伝統的な宅配サービスは、宅配ドライバーが各家庭を一か所ずつ順番に訪問するスタイルが一般的であった。この一連のプロセスの一部、もしくはその全てをドローンに代替させることによって、従来よりも少ない宅配コストで荷物を届けることを可能とするのがドローン宅配サービスである。

本章では、本格的な少子高齢化・人口減少社会を迎えた我が国の物流業界が直面する現状の課題について説明するとともに、ドローン宅配サービスの実現がその解決策として有望であることを示す。また、ドローン宅配サービスを実現する上で今後検討しなければならない課題と、国内外におけるドローン宅配サービスの実現に向けた研究開発状況についても示す。最後に、本研究の位置づけと論文構成について示す。

## 1.2 研究背景

### 1.2.1 少子高齢化と人口減少社会

総務省の国勢調査によると、我が国の生産年齢人口(15歳以上 65歳未満)は1995年を境に減少を始め、2008年には総人口そのものが減少に転じたと報告されている<sup>[12]</sup>。また、国立社会保障・人口問題研究所が発表した統計資料によると、今後約40年間で日本の人口はピークの約1億2700万人から約8600万人まで縮小することが予想されている(図1.1参照)<sup>[13]</sup>。この時の高齢化率(総人口に占める65歳以上の割合)は約40%にまで達し、生産年齢人口は2005年の半分(全人口の51%以下)に落ち込むと試算されている。我が国は既に少子高齢化

社会から超高齢化・人口減少社会へと移行したといえる。このような社会構造の変容は、我が国の需要と供給の両面で悪影響を与え、社会全体としての経済成長を中長期的に阻害すると懸念されている。

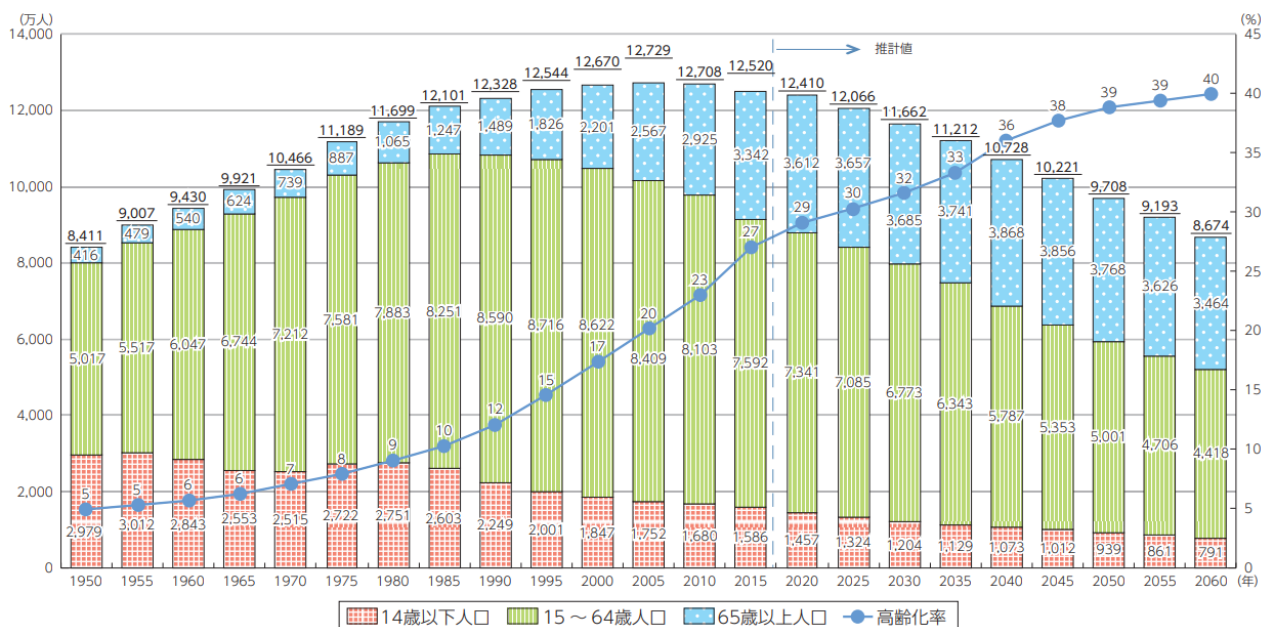


図 1.1：日本の人口推移

## 1.2.2 物流業界の現状

日本物流団体連合会の調べでは、トラック運送事業の規模は平成 28 年度時点で運送業全体の約 6 割以上を占め、営業収入は 16 兆円近くまで拡大している(図 1.2 参照)<sup>[14]</sup>。また、近年の宅配便取扱個数は、平成元年には約 10 億個だった実績が、その 10 年後には約 1.8 倍、さらに 20 年後には約 3.2 倍、そして平成 29 年には 4.2 倍にまで増加している(図 1.3 参照)<sup>[15]</sup>。これらの背景には、インターネットに代表される情報通信技術の一般社会への浸透や、アマゾンや楽天、ヤフーなどの大手ベンチャー企業が提供する電子商取引(Electronic Commerce(EC))サービスの登場などの要因が挙げられる。消費者は 24 時間、ネットを通じて場所を問わず自分が欲しいもの、売りたいものをいつでも取引できるようになり、さらにはその配達時間や受け取り場所さえも柔軟に指定できるようになった。今や肉や魚、野菜類といった生鮮食品や調理済みの料理さえもネットを介して手軽に注文できるようになっており、しかも注文した品は数時間以内には手元に届くというサービスまで登場してきている。今や、このような宅配サービスは我々が日常生活を送る上でごく当たり前に存在する社会インフラとしての地位を確立しているといっても過言ではない。

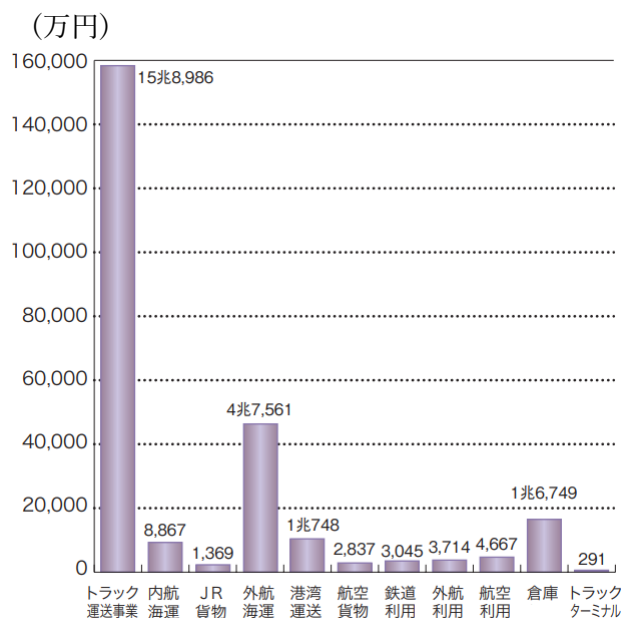
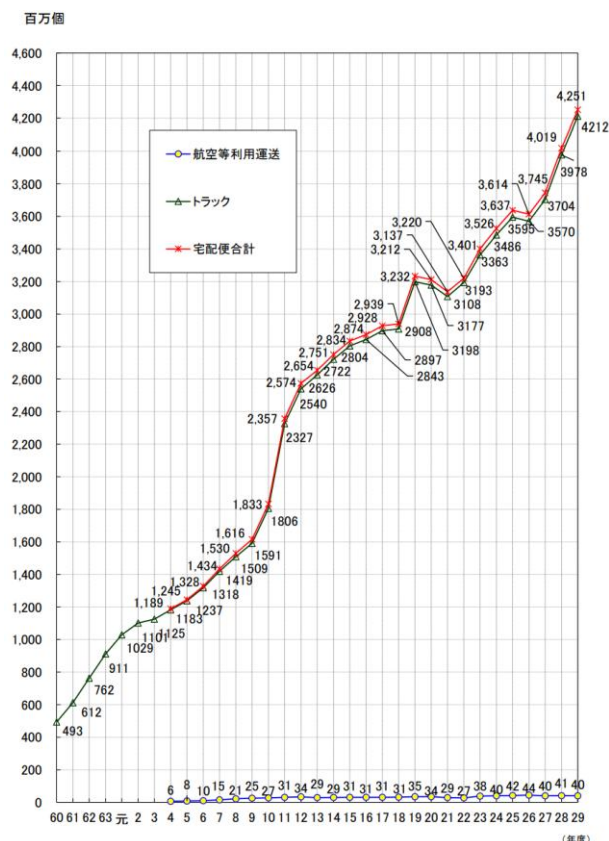


図 1.2：物流業界の事業分野別営業収入



ら 20 代のドライバーに至っては全体の 1 割にも満たない(図 1.6 参照)。一方、総務省の労働力調査によれば、道路貨物輸送業に従事する就業者数は平成 20 年には 183 万人存在したが、その 10 年後には 5.5% 増の 193 万人ほどしか増えていない(表 1.1 参照)<sup>[19]</sup>。この期間に宅配便の取扱個数が約 3 割近く増加していることを考慮すれば、明らかにドライバー 1 人当たりの負担が増加していることが窺える。

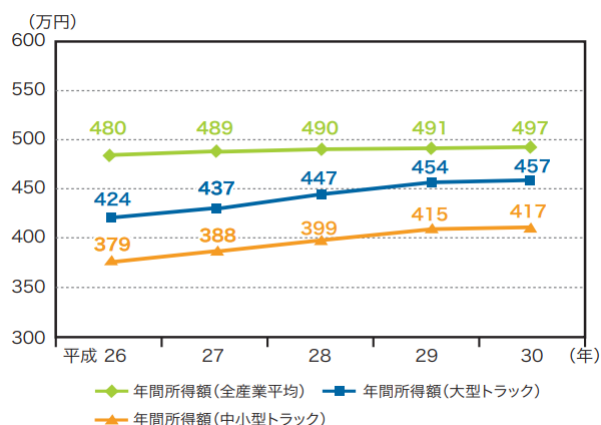


図 1.4：ドライバーの年間所得額の推移

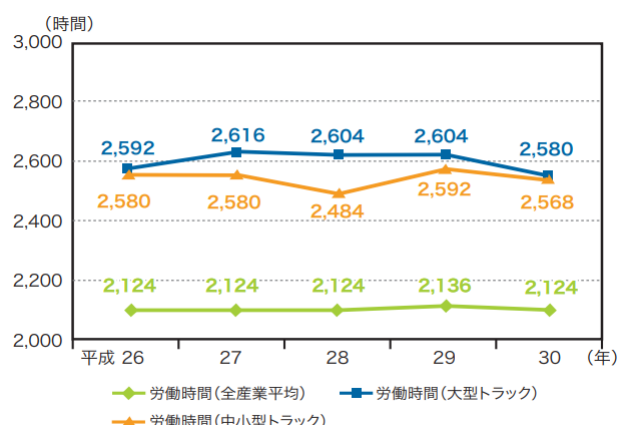


図 1.5：ドライバーの年間労働時間の推移

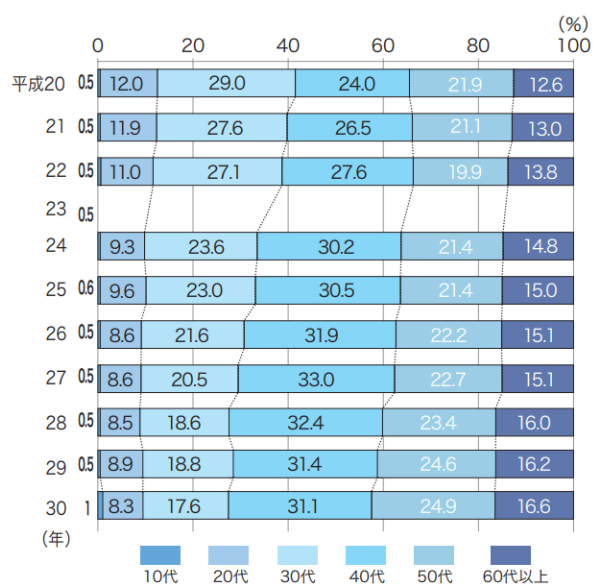


図 1.6：年齢階級別就業者構成比

表 1.1：運送業の就業者数推移(単位：万人)

年	道路貨物運送業					
	就業者数			輸送・機械運転従事者数		
	総数	男	女	総数	男	女
平成20	183	152	31	79	77	2
21	185	152	33	80	78	2
22	181	148	33	79	77	2
23	—	—	—	—	—	—
24	182	150	32	83	81	2
25	187	153	34	84	83	2
26	185	151	33	83	81	2
27	185	151	34	80	78	2
28	188	153	35	83	81	2
29	191	156	35	83	81	2
30	193	155	38	86	84	2

このように、若年層を中心に輸送業務を担当する労働者が減少しており、しかも従業員 1 人当たりの負担は年々増加傾向にあることがわかる。この現状を放置すれば、少子化による影響に加え、若年層の就職先としてますます忌避されるようになり、新たな人材の確保もより困難となる。ひいては、この現状が宅配サービスの利便性低下や宅配コストの上昇などといった形で顕在化し、今後の経済発展を阻害する要因になる。以上のような背景から、特に宅配サービスの省人化は喫緊の課題となっているといえる。

### 1.3 ドローン宅配への期待とその実現に向けた課題

ドローン宅配は、宅配サービスの省人化・効率化を図る有効な手段の1つとして既に世界中で注目されているテクノロジーである一方、その実現に向けた国内における課題は山積している。以下にその主要なテーマのいくつかを列挙する。

#### (1)機体の選定

現在、ドローンは民生用・業務用を含めて数多くの製品が販売されており、それらの機体の大きさや性能、価格、機能についても千差万別である。はじめに、ドローンの用途別にどのような機体が存在しているのかいくつか例を挙げる。図 1.7 は用途別のドローンの一例を示す。

図 1.7(a)は、Ycco が販売する長さ 5cm、幅 5cm、高さ 2.5cm の小型ホビー用ドローンであり、重さは僅か 50g 程しかないが、10 分以上の滞空時間を持ち、6 軸ジャイロを搭載しているので誰でも簡単な操作で飛行を制御することが可能となっている<sup>[20]</sup>。また、価格も 4 千円前後に抑えられており、ドローンの初心者用として販売されている。

図 1.7(b)は、DJI 製の「Mavic 2」と呼ばれる空撮専用のドローンであり、価格は約 20 万円と高価格帯に属しているものの、民生用としては初めて全方向障害物検知システムと自動回避機能を搭載した画期的なモデルとなっているのが特徴である<sup>[21]</sup>。大きさは展開した状態で、長さ 32.2cm、幅 24.2cm、高さ 8.4cm、機体重量は 1kg 以下に抑えられている。ただし、飛行可能時間については実質 20 分程度しかない。

図 1.7(c)は、農薬散布での使用を想定した「エアロ・レンジ II」と呼ばれるガソリンエンジン搭載の業務用ドローンであり、4 本のアームと 8 個のプロペラを搭載し、機体幅は 1 メートルを超える<sup>[22]</sup>。航続時間は 3 時間、航続距離は 100km、積載可能重量は 15kg と高スペックであるものの、導入コストは 500 万円と高額である。

図 1.7(d)は、現在中国で開発が進められている人が搭乗可能な「Ehang 184」と呼ばれる大型ドローンである<sup>[23]</sup>。この機体も 4 本のアームと 8 個のプロペラを搭載しており、積載可能重量は 200kg、飛行速度は 130km/h に達すると報告されている。

以上で示したように、ドローンには多種多様なモデルが実用化されており、より複雑で困難な用途への応用も期待されている。一般的に、小型のドローンであれば導入コストや墜落時のリスクを低減した上で空撮などの簡単な任務を遂行するには都合がよい。一方で、航続距離や積載可能重量が限られていることや、突風などの影響を受けやすいといった小型機特有のデメリットも存在する。反対に、機体サイズを大型化することによってこれらのデメリットは解消されるが、導入コストが高額になることや墜落時のリスクが増大することなどのデメリットが顕在化する。したがって、宅配ドローンとして使用するモデルは、その用途に適合する機体性能であるとともに、その導入コストや操作性、安全性などといった諸般の要素を全て勘案した上で最もバランスの良い機体を選定しなければならない。



(a) : ホビー用ドローン



(b) : 空撮用ドローン



(c) : 業務用ドローン



(d) : 人が搭乗可能なドローン

図 1.7 : 用途別のドローンの例

## (2) ドローンの運用方法

宅配サービスにおけるドローンの運用方法について、Chase らはドローンを用いて宅配拠点から各宅配場所へ直接宅配するスタイルや、トラックにドローンを搭載して2か所同時に宅配するスタイル(彼らは「Flying Sidekick Delivery」と表記しているが、「Last One Mile Delivery」や「Tandem Delivery」などと呼ばれる場合もある)について言及している<sup>[24]~[29]</sup>。前者の宅配スタイルは、宅配拠点近傍をドローンで宅配し、その遠方をトラックでカバーすることによって宅配ドライバーの負担軽減を実現する。一方、後者の宅配スタイルは、宅配場所の近傍まではトラックで移動し、残りの距離をドローンでカバーすることによって宅配ドライバーの負担を減らすことを目的としている。後者の宅配スタイルは米国のUPS社によって既に実証実験が行われている<sup>[30]</sup>。図1.8はUPS社が想定しているTandem Deliveryの概念図を表す<sup>[31]</sup>。この図によると、ドライバーは駐車中のトラック内でドローンに荷物を搭載後、その近傍の宅配場所へ向けてドローンを離陸させる。ドローンが飛行中は、ドライバーも同時並行で別の宅配場所に向けて移動を開始する。宅配を終えたドローンはトラックの移動先である宅配場所にてドライバーに回収される。

以上に示した2つの宅配スタイルは、いずれもドローンの運用方法に違いはあるものの、

ドローンが宅配作業の一部を肩代わりすることで宅配コストを低減させるという点で一致している。この例から分かるように、宅配サービスにおけるドローンの活用方法は1つではない。したがって、どのような宅配スタイルがドローンの運用手段として最も合理的か、検討する余地はある。

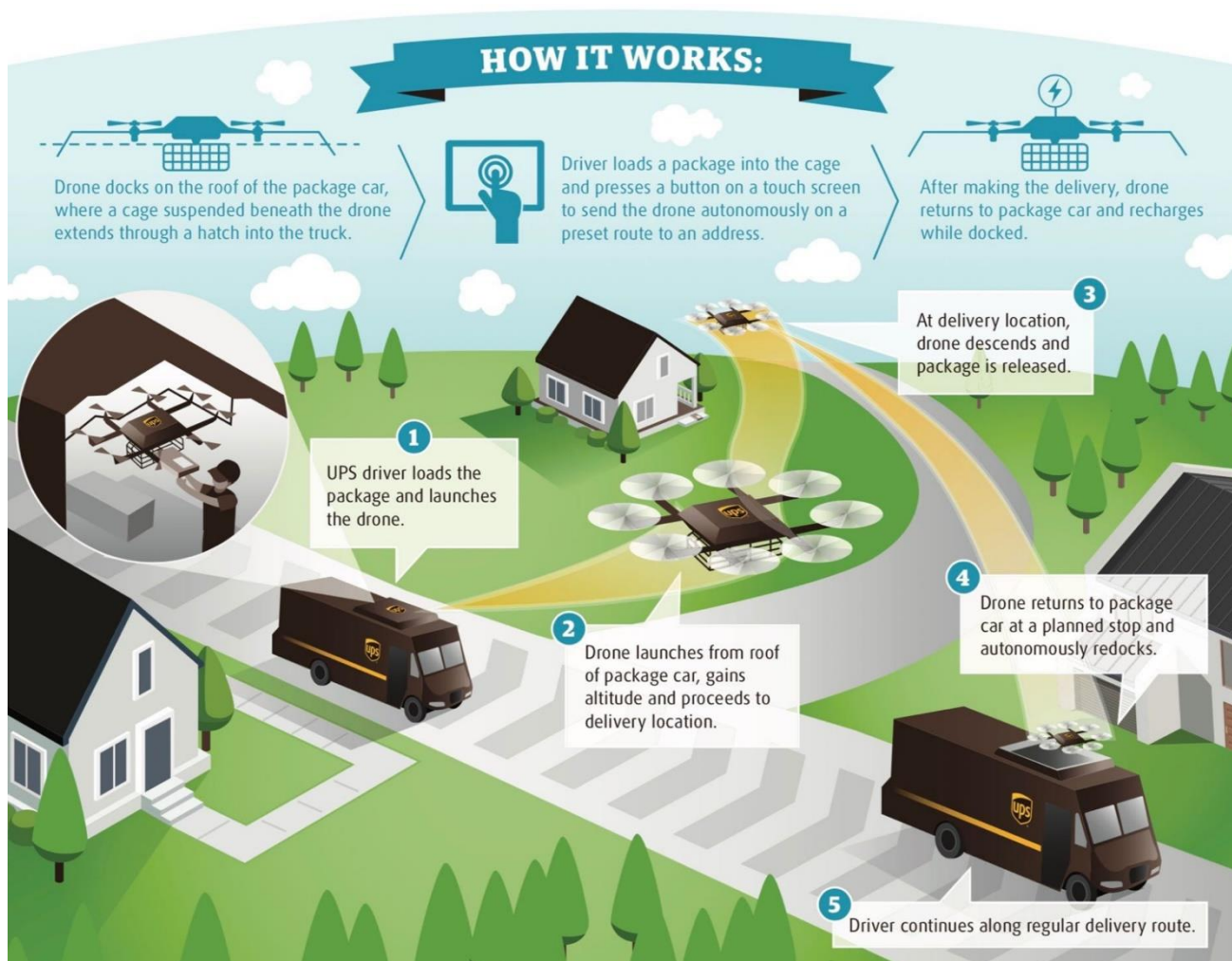


図 1.8：米国 UPS 社が想定している Tandem Delivery の概念図

### (3)法整備

国内で発生した数件のドローン墜落事故を受け、ドローンの飛行を規制する改正航空法が2015年12月に施行された<sup>[32]</sup>。以来、「無人航空機」の定義が明文化され、その飛行禁止空域が図1.9で示すように明示されるようになった。また、図1.10に該当する飛行は、国土交通省による許可が必要となった<sup>[33]</sup>。このような厳しい法規制の下でのドローン宅配サービスは実現不可能であるため、規制緩和にむけた検討が必要といえる。

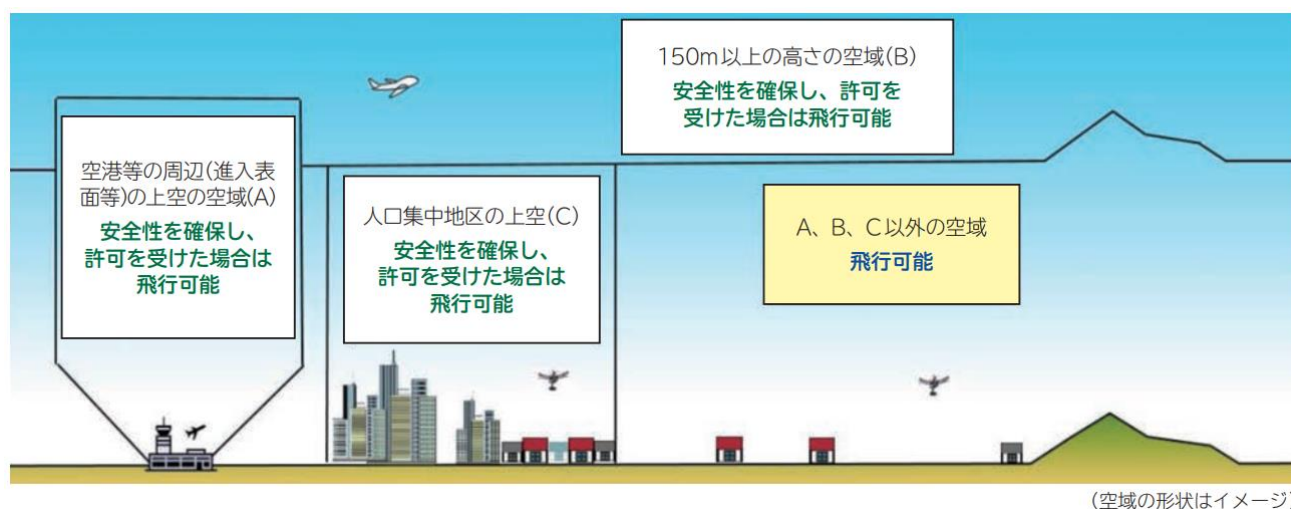


図 1.9：無人航空機の飛行許可が必要となる空域

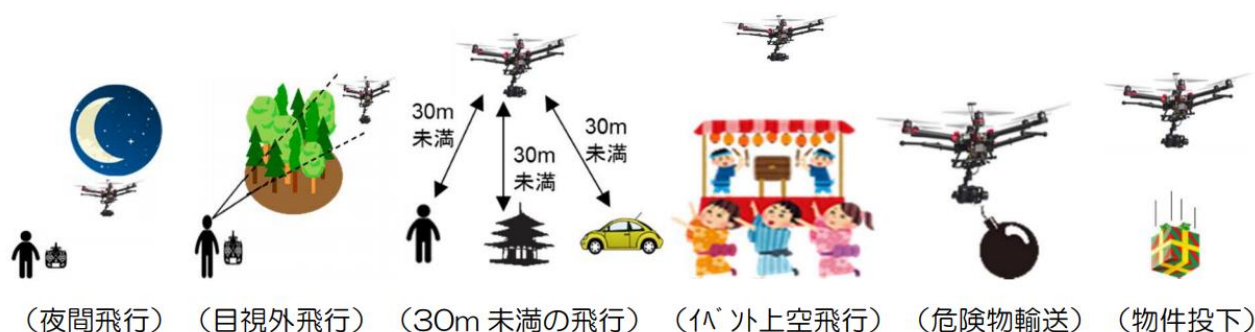


図 1.10：特別の許可を必要とする飛行方法

#### (4)航空管制・空域管理

ドローンの活用範囲は、物流に限らず災害対応やインフラ点検、警戒監視に至るまで非常に幅広い。将来的には、上空の至る所をドローンが飛び交うことになると予想される。そのため、ドローン同士が衝突しないように安全な航行が可能な空域管理が必須となる。以下に、我が国で進められているドローンの運行管理システムの代表的な例について紹介する。

1 つ目は、国の行政機関や各民間企業、官民の研究機関などの連携による「ロボット・ドローンが活躍する省エネルギー社会の実現プロジェクト」が挙げられる<sup>[34]</sup>。これは、経済産業省/新エネルギー・産業技術総合開発機構(NEDO)が 2017 年からスタートさせたプロジェクトであり、目視外の領域において数多くのドローンが飛び交う状況であっても相互に安全を確保しながらそれぞれのミッションを遂行できる運行管理システムの構築を目指している(図 1.11 参照)。また、事業者やドローンの運用者に対して必要な運航管理サービスや各種情報を提供することも目的としている。現在、福島ロボットテストフィールド(福島 RTF)にその開発拠点を設けて研究開発に取り組んでいる。将来的には、ここで培った機体同士の衝突回避技術などを国際標準企画にまで昇華させることを目指している。

2つ目は、「ドローンハイウェイ構想」である(図 1.12 参照)<sup>[35]</sup>。この構想は、東京電力ベンチャーズとゼンリンが共同で立ち上げたプロジェクトであり、街中に張り巡らされた送電線網を安全な航空路として利活用する取り組みである。ゼンリンは 3 次元地図を提供し、東京電力ベンチャーズは送電網の各鉄塔上に気象観測装置を設置して風向などの情報をリアルタイムで提供する。これらの情報を統合し活用することで、ドローンは現在地における風速情報を逐次取得することが可能となる。もし飛行中になんらかの危険を検知すれば、その場でホバリングするなどの危険回避行動を取る。このシステムは目視外におけるドローンの安全な自律飛行を実現することを目的としたものであり、首相官邸が主導する空の産業革命に向けたロードマップ 2018 にも示された構想である<sup>[36]</sup>。これらは、ドローンの安全な飛行を実現するための研究開発に関する一例に過ぎないが、このような安全で円滑な航空管制、空域管理の構築が今後必要になる。

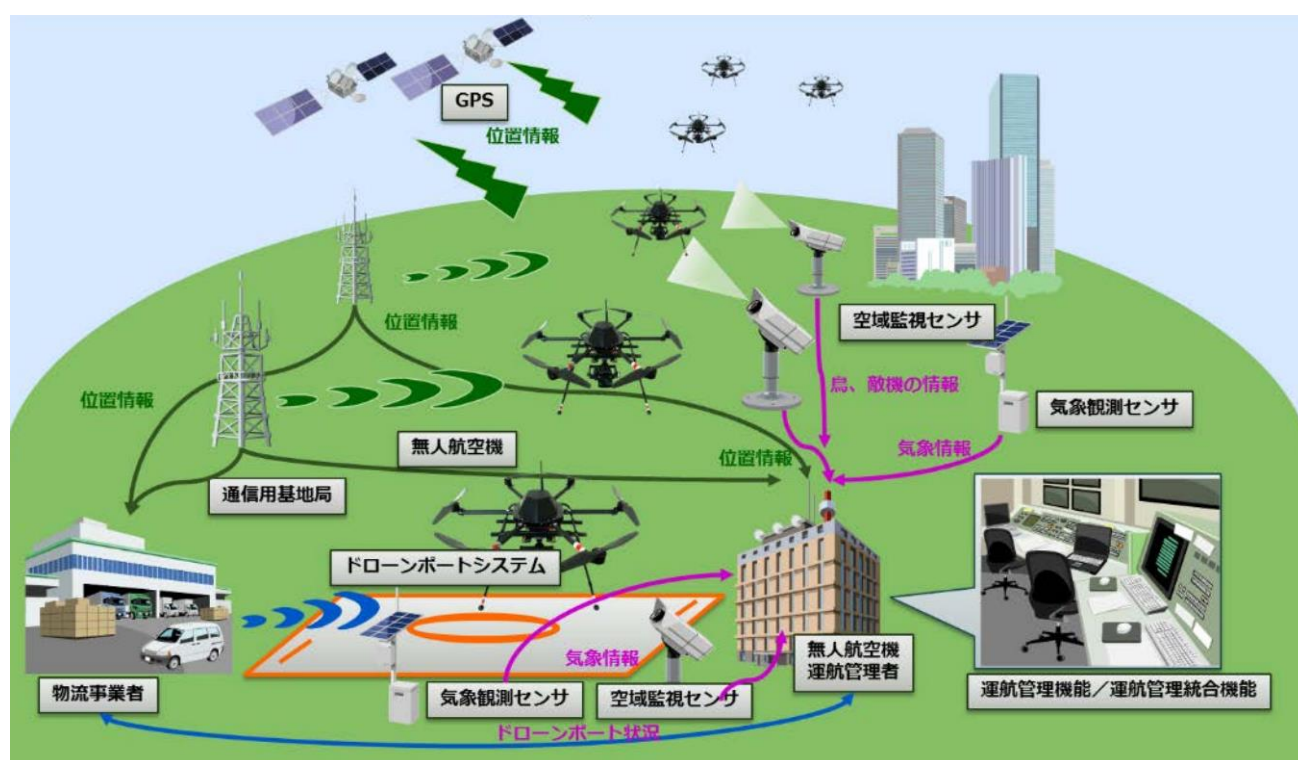


図 1.11：目視外領域におけるドローンの運行管理システムの構想図



これは、ポート自体に風向、風速を検知するセンサやドローンの安全な離着陸の可否判断、ポートまでの安全な誘導など、宅配ドローンの円滑なオペレーションをトータルで支援する機能が完備されている。これにより、ドローンはオペレータによる制御なしでも自律的に宅配物をポートに置いた後、帰投することが可能になる。このようなインフラを各家庭や集合住宅、高層ビルなどに整備していくことを、その実現可能性を含めて検討していく必要がある。

## (6)安全管理

ドローン宅配を実現する上で最も重要視されるのはその安全性である。これを担保するためには、機体の信頼性を向上させることはもちろん、万が一墜落事故が発生してもその被害を最小限に留めるセーフティー機能が求められる。特に、人的な被害が生じた場合は、大きな損害を被る上にその後のサービスの継続さえ厳しくなる事態にも発展しかねないと予想されることから、その対策には万全を期す必要がある。

## (7)世論

前述したように、ドローン宅配サービスには墜落の危険性や騒音被害、プライバシーの侵害、ハッキング等による誤作動から盗難に至るまで、様々な運用上のリスクが想定されるため、その1つ1つに確実な対策を講じなければならない。これらが不十分だと周辺住民の不満が高まる恐れがあり、ドローン宅配サービスを新たなビジネスチャンスとして展開していくことが困難になる。したがって、実証実験を積み重ねながら少しずつ関連する技術レベルを向上させ、その信頼の醸成に継続して取り組まなければならない。

## 1.4 ドローン宅配実現に向けた取り組み

### 1.4.1 国内事業

我が国では、人口の一極集中と地方の過疎化が社会問題の1つになっている。特に人口減少が著しい地方では宅配トラックの積載率が低下し、非効率的な宅配サービスを余儀なくされている。国土交通省による調査によれば、荷物1つを運ぶために走行するトラックの走行距離は、過疎地域において都市部より平均して約6倍に増加することが報告されている(表1.2参照)<sup>[38]</sup>。また、需要が低い地域には商業施設が立地しにくいいため、日常生活の必需品さえ購入が困難な住民が多数存在するといわれている。このような、いわゆる「買い物弱者」と呼ばれる住民の累計数は2015年時点で既に約700万人以上と推計されており、今後さらなる増加が危惧されている<sup>[39]</sup>。この事態に対応するため、過疎地を中心にドローン宅配サービスの実証実験等が産官学一体となって進められている。その活動の一環として、「過疎地域等におけるドローン物流ビジネスモデル検討会」が国土交通省の主導で2019年

3月に開始された<sup>[40]</sup>。これは、過疎地域におけるドローン宅配の商用展開を推進する事業の一環であり、2019年6月末時点で全国5地域(長野県白馬村, 福島県南相馬市・浪江町, 福岡県福岡市, 岡山県和気町, 埼玉県秩父市)における検証実験と計4回にわたる審議会が執り行われた。この他, 地方公共団体・民間事業者等における取組として, 長野県伊那市と大分県佐伯市におけるドローン物流プロジェクトも始まっている。

このように, 日本国内では都市部よりも過疎化が深刻する地方におけるドローン宅配サービスの実現に向けた取り組みが活発化しており, その一部には実証実験を完了させ, 一定の成果を上げているものもある。また, 法律的な側面から考慮しても, ドローンが自由に飛行できる余地が大きいのは人口密集地などの都市部ではなく, 田園風景が広がるような農村地帯である。これらの事情を勘案すれば, 都市部よりも地方におけるドローン宅配サービスの実現が費用対効果と実現可能性の両面で有利であると考えられる。

表 1.2 : 荷物 1 個あたりのトラックの走行距離の比較(物流事業者 A 社実績)

地域	トラック走行距離	荷物個数	荷物 1 個あたりの走行距離
過疎地域	約 37 万(km/月)	約 30 万(個/月)	約 1.2(km/個)
都市部	約 37 万(km/月)	約 160 万(個/月)	約 0.2(km/個)

#### 1.4.2 海外事業

海外では日本に先駆けて数多くのドローン宅配に関わるプロジェクトが進行しており, 中には既に商用化に漕ぎ着けた事例も存在する。代表的な事例をいくつか紹介する。

米大手アマゾンでは 2013 年にドローンによる宅配サービスの実現に向けた発表を行い, 2019 年 6 月には Machine Learning(機械学習), Automation(自動化), Robotics(ロボット工学), Space(宇宙)に関するイベント「re:MARS」において, 数か月以内にドローン宅配サービス「Amazon Prime Air」の商用化を開始すると発表した<sup>[41]</sup>。使用される機体の配送範囲は約 24km, 2.26kg までの荷物を運ぶことができ, 顧客からの注文に対しては 30 分以内での配送を実現するとしている。使用する機体には 6 つのローターが装備されており, ヘリコプターの垂直離陸能力と飛行機の空中旋回性能を兼ね備えたハイブリッドモデルになるという(図 1.14 参照)。内部にはコンピュータビジョンと機械学習機能が搭載された人工知能(AI)が組み込まれており, 鳥などの飛行物体や電線などの検知しにくい障害物も回避できる仕様となっている。



図 1.14 : Amazon Prime Air での使用が見込まれるドローン

同じく米国の UPS Flight Forward は、2019 年 7 月に米国連邦航空局 (FAA) に対し、UPS が提供するネットワーク内での商用目的のためのドローン飛行について認可申請を行ったと報じられた<sup>[30]</sup>。この申請が承認されれば、米国内では特例的に昼夜を問わずドローンの運用が可能となり、目視外での飛行も実現する見込みだという。これは他社の限定的な FAA 認可とは異なり、より包括的なドローンの活用を認可するという点において極めて画期的な事例になることが期待されている。

アイスランドのスタートアップ「Aha」は、世界最大手のドローンメーカー DJI 及びイスラエル企業と共同してドローンによる物流サービスを 2018 年から開始した<sup>[42]</sup>。このサービスは、世界で初めて商用のドローン宅配サービスを実現させた事例として注目され、アイスランドの首都レイキャビクにおいて、ドローンを使って料理、生鮮食品、日用品といった物資を各家庭へ届ける動画が投稿され、話題となっている。使用するドローンは 3kg 以内の荷物であれば 4~8km 圏内で飛行することができ、可能な限り産業地区や川、湖などの上空を飛行するようプログラムされている。ただし、障害物を検知したり回避したりする機能は一切搭載されておらず、予め設定された安全なルートのみを飛行するスタイルを採用している。報道された時点まででおよそ 500 件以上のオペレーションを完遂し、1 度の事故も起きていないという。

以上に挙げた事例はごく一部に過ぎないが、少なくとも海外では国の承認を完了させ、ドローン宅配サービスの商用利用を開始するフェーズにまで到達していることが報告されている。これらの現状と比較すれば、日本はこの分野において既に後進国の地位に甘んじていると言わざるを得ない。

## 1.5 本研究の目的

前述したように、我が国の物流業界は深刻なドライバー不足と宅配需要の増加に直面しており、その解決策の 1 つとしてドローン宅配サービスの実現に大きな期待が寄せられている。しかしながら、そのためには解決しなければならない数多くのハードルがいくつも存在することが明らかになった。本研究では、ドローン宅配サービスにおける宅配経路の生成方法について考察する。

宅配経路生成は、制約付き多目的最適化問題として定式化することができる。「制約付き」とは「制約条件」を表し、ドローン宅配サービスにおいては宅配トラックの燃料やドローンのバッテリー稼働時間、積載可能重量等がこれに該当する。「多目的最適化」とは、最適化の対象となる目的関数が複数存在することを意味し、宅配ドライバーの勤務時間やトラックの走行距離等がこれに該当する。この多目的最適化問題の解法についてはこれまで数多くの手法が提案されてきたが、いずれもその適用には何らかの制約や試行錯誤的な要素が必要になるという欠点を有しているため、これらをそのまま準用することは困難と考えられる。そこで本論文では、これらの欠点を回避し、意思決定者にとって最も適当な選好解を生成可能な新しい多目的最適化手法を提案する。また、この手法を用いてドローン宅配サービスによる宅配コスト削減効果を定量的に評価することにより、その便益を最大限享受するための知見を得ることが本研究の目的である。

## 1.6 論文構成

本論文の構成は次のとおりである。次章では、まず最適化問題に関連する一般的な概念や用語について整理する。続いて、最適化問題の近似解法に該当するいくつかの解探索アルゴリズムについて紹介する。また、これまで提案されてきた数多くの多目的最適化手法を 2 つのカテゴリーに分類し、その両方の利点と欠点について整理するとともに、新たな最適化手法の必要性について説明する。第 3 章では、「暫定理想点法」と名付けた遺伝的アルゴリズムをベースとした解探索アルゴリズムを提案する。この手法は、制約付き多目的最適化問題における実行可能解の生成と多目的最適化を実現する解探索手法である。この手法の適用方法について示すとともに、GO 問題及び DTLZ 問題と呼ばれるいくつかのベンチマーク問題を用いて本手法の有用性と妥当性を検証する。第 4 章では、ドローン宅配経路問題を制約付き多目的最適化問題として定式化するとともに、提案手法を用いてこの問題を解くための解表現方法とその遺伝的操作方法について示す。第 5 章では、提案手法を用いてドローン宅配経路を生成する際に留意すべき問題点について説明するとともに、これに対処するための改良型暫定理想点法について提案する。第 6 章では、第 5 章で提案した改良型暫定理想点法を用いて生成したドローン宅配経路が従来のトラックのみを用いた宅配経路と比較してどれくらい宅配コストを削減することができるのか定量的に評価する。さらに、それらの解

---

析結果を通じて得られたドローン宅配サービスのコスト削減効果に関する知見について示す。第7章では、本論文の結論と今後の課題について述べる。

## 第2章 最適化問題と解探索アルゴリズム

### 2.1 最適化問題

#### 2.1.1 一般式

最適化とは、ある目的関数を最小化、あるいは最大化する解を求めることである。仮に後者の場合、目的関数の逆数をとるかその符号を反転させることによって結局は最小化問題に変換することが可能なので、その意味において全ての最適化問題は最小化問題として扱って差し障りない。したがって、本論文では最適化問題とは全て最小化問題を意味するものとする。最適化問題には、解が満たすべき制約条件がある場合とそうでない場合が存在する。前者を「制約付き最適化問題」、後者を「制約なし最適化問題」という。ここで、制約付き多目的最適化問題は、一般的に式(2.1)に示すように定式化される。

$$\begin{cases} \min_{\mathbf{x}} \mathbf{F}(\mathbf{x}) = \min_{\mathbf{x}} [F_1(\mathbf{x}), F_2(\mathbf{x}), \dots, F_I(\mathbf{x}), \dots, F_{I_{max}}(\mathbf{x})] \\ \text{subject to } g_J(\mathbf{x}) \leq 0 \\ I = 1, 2, \dots, I_{max} \quad J = 1, 2, \dots, J_{max} \quad \mathbf{x} \in \mathbf{S} \end{cases} \quad (2.1)$$

ここで、 $\mathbf{F}(\mathbf{x})$ は最適化の対象となる全ての目的関数の評価値で構成されたベクトル関数であり、 $\mathbf{x}$ は設計変数、 $\mathbf{S}$ は実行可能領域を表す。 $\mathbf{x}$ は単なる実数で与えられる場合もあれば、ベクトル値や数列で与えられる場合など、問題に応じて様々な形態をとる。本論文のように宅配経路に関する最適化問題の場合であれば、 $\mathbf{x}$ は宅配経路を表す解となる。不等式制約条件 $g_J(\mathbf{x})$ についても扱う問題に応じて様々な条件設定が想定されるが、いずれも $g_J(\mathbf{x}) \leq 0$ の形に変形可能であるのでそのように表記する。上式を文章で表現すれば、「 $J_{max}$ 個の制約条件 $g_J(\mathbf{x})$ を満足し、かつ $I_{max}$ 個の目的関数を総合的に最適化する実行可能解 $\mathbf{x}$ を探索する」となる。ここで、 $I_{max} = 1$ の場合、上式は「制約付き単目的最適化問題」、 $I_{max} > 1$ の場合は「制約付き多目的最適化問題」となる。

#### 2.1.2 最適化問題の分類と宅配経路問題の特徴

最適化問題は、その特徴に応じて様々なカテゴリーに分類することが可能である。例えば、 $\mathbf{x}$ が連続値か離散値かによって「連続最適化問題」あるいは「離散最適化問題」に分類でき

る<sup>[43]</sup>. 後者は「組合せ最適化問題」ともいう. 一方, 連続値と離散値の両方を扱う最適化問題は, 「混合整数最適化問題」と呼ばれ, 同じく組合せ最適化問題の一種に該当する. さらに, 与えられた目的関数や制約条件は線形か非線形か, 解空間は凸か非凸か, 複雑性クラスはどのカテゴリーに分類されるか(詳細については後述する.)という観点からも区別可能である.

宅配経路問題における $x$ の解構造は宅配経路を表す離散的な解構造でなければならず, 単なる実数値などによって表現することはできない. 一方, 目的関数及び制約条件については, トラックの走行距離や宅配時間, トラックの台数等が考えられ, これらの値は実数値, もしくは整数値で与えられる. したがって, 宅配経路問題は, 離散構造の解をもつ混合整数最適化問題としての特徴を有すると解釈できる. また, 目的関数と制約条件を定式化する過程において2次式が使用されない限り, この問題は線形問題として扱われる. そうでない場合は, 非線形問題として扱われる. ただし, 凸性に関しては, 宅配経路問題が離散性を有していることから判定することはできない.

以上の考察より, 宅配経路問題は極めて複雑なクラスの組み合わせ最適化問題に分類されることがわかる. このように, 対象とする最適化問題の特徴を把握しなければ, それに対する適切な解法を選択することもできない. もちろん, 前述したように判定できない要素が含まれる場合もあるので, その解法としてはどのようなクラスに分類される最適化問題であっても適用可能な汎用的手法が望ましい.

### 2.1.3 大局的最適解と局所的最適解

最適化問題における最適解は, 「大局的最適解」と「局所的最適解」に大別できる<sup>[44]</sup>. 例えば,  $I_{max} = 1$ となる単目的最適化問題において, 全ての実行可能解 $x_F$ に対して式(2.2)となる $x_G$ が与えられたとき, これを大局的最適解という.

$$F_1(x_G) \leq F_1(x_F) \quad (x_G \in S, \forall x_F \in S) \quad (2.2)$$

これに対し,  $x_F$ の近傍, すなわち $S' \subset S$ となる一部の実行可能領域において, 式(2.3)となる $x_L$ が与えられたとき, これを局所的最適解という.

$$F_1(x_L) \leq F_1(x_F) \quad (x_L \in S', \forall x_F \in S') \quad (2.3)$$

この2つの最適解の概念図を図2.1に示す.

一般的に, 最適化問題は, 目的関数や制約条件, 設計変数の数が増加するほど探索すべき解空間も広大になるため, その全てを網羅的に探索して大域的最適解を求めることは現実的ではない. そこで, はじめから探索する領域を限定して, 現実的な時間内で効率よく優れた局所的最適解を求めるアプローチが主流となる. このような解法は, 2.2節で後述する「近似解法」に該当する. 宅配経路問題における解法でも近似解法が主流となる.

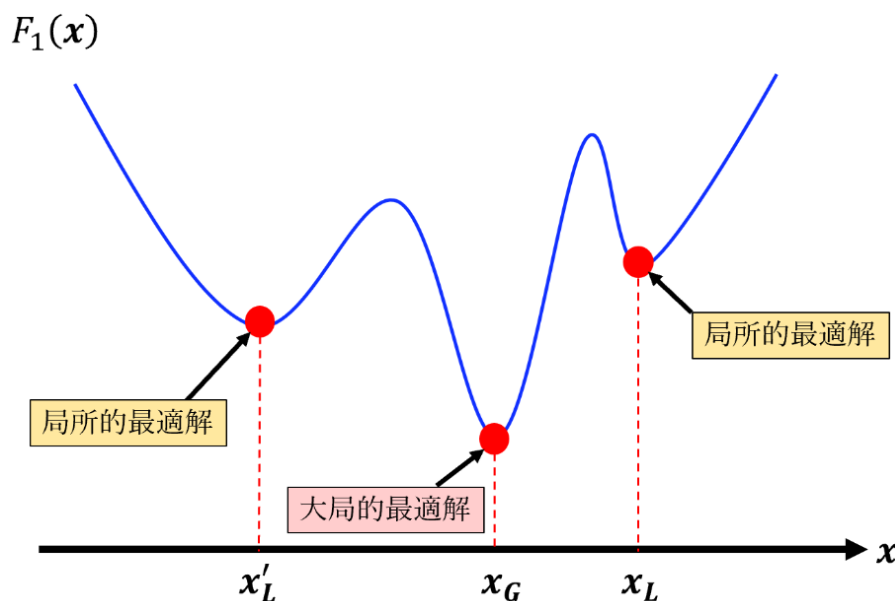


図 2.1：大局的最適解と局所的最適解の概念図

#### 2.1.4 多目的最適化問題の特徴

多目的最適化問題では、一般的に目的関数同士が互いにトレードオフとなる<sup>[45]</sup>。例えば、車を購入するとき、高額な車体ほど優れた性能や乗り心地、快適性を期待できるが、逆に安価なものを選択すれば、その分だけこれらの要素を犠牲にすることになる。購入予算は、制約条件に該当する。このように、目的関数が2つ以上になると、その全てを最適化できる解が存在することはほぼないので、意思決定者は適当な妥協点を探らなければならない。したがって、意思決定者次第で選択される最適解も変わるということである。ここで、トレードオフの関係を視覚化するため、2目的最適化問題の解空間を表す概念図を図 2.2 に示す。

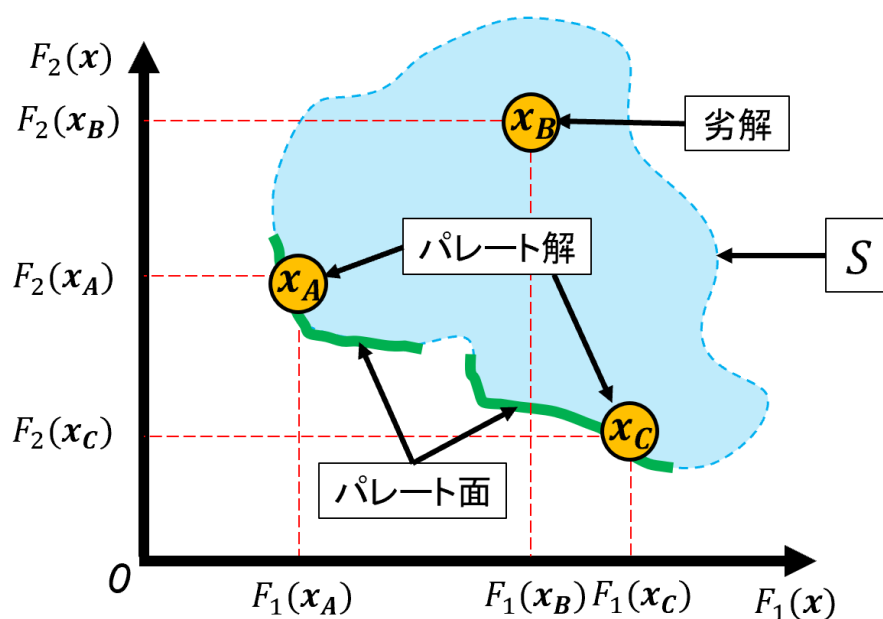


図 2.2：2 目的最適化問題の解空間を表す概念図

上図における実行可能領域 $S$ には $x_A, x_B, x_C$ の3つの解が描かれている。これらの解の大小関係は次のとおりである。

$$\{F_1(x_A) < F_1(x_B)\} \cap \{F_2(x_A) < F_2(x_B)\} \quad (2.4)$$

$$\{F_1(x_A) < F_1(x_C)\} \cap \{F_2(x_A) > F_2(x_C)\} \quad (2.5)$$

式(2.4)において、 $x_A$ は $x_B$ のいずれの目的関数値よりも小さいため、この場合は明らかに $x_A$ の方が優れた解であると判定できる。このとき、 $x_B$ を「劣解」といい、「 $x_A$ に支配される」とも表現する。一方、式(2.5)においては $x_A$ と $x_C$ の優劣関係を一概に判定することができない。このように、他のどの解と比較しても劣解とはならず、少なくとも1つ以上の目的関数値については他方より優れている解のことを「パレート解」もしくは「非劣解」と呼ぶ。図2.2において、 $x_A$ 及び $x_C$ はこのパレート解に該当する。緑の曲線で表される部分はパレート解の集合を表し、「パレート面」もしくは「パレートフロンティア」と呼ばれる。なお、 $I_{max} \geq 4$ 以上の場合、パレートフロンティアは超平面となるので描画することはできない。意思決定者にとっての最適解は、このパレート面に存在するパレート解のいずれかと考えられる。数あるパレート解の中から意思決定者の選好情報にもっとも近いものを「選好解」と呼ぶ。多目的最適化問題における解法の目標は、この選好解を生成することである。

宅配経路問題では、宅配時間や走行距離、トラックの台数など、最適化したい対象や考慮すべき制約条件が多数存在すると考えられる。また、この問題がドローン宅配サービスを想定した最適化問題であると仮定するならば、ドローンの飛行可能時間やペイロードなどといったより多くの制約条件などについても同時に満足しなければならない。したがって、この種の問題を解く際には、広大な解空間の中から実行可能解を見つけ出すことと、その中でもより優れた実行可能解を効率よく探索する手法が必要と考えられる。

### 2.1.5 複雑性クラス

最適化問題は $\mathcal{P}$ ,  $\mathcal{NP}$ ,  $\mathcal{NP}$ 完全,  $\mathcal{NP}$ 困難と呼ばれる計4つのクラスに分類することができ、この列挙順に難易度が上がる<sup>[43]</sup>。これらは総称して「複雑性クラス」と呼ばれる。 $\mathcal{P}$ は多項式時間の英訳「polynomial time」の頭文字に由来する。

ある最適化問題がどのクラスに属する問題であるかを判別するためには、その解を得るまでの計算量を見積もらなければならない。ここで、 $n$ 個の都市を全て訪問する最短巡回経路を求める巡回セールスマン問題(Travelling Salesman Problem (TSP))を例にすると、その解候補は全部で $(n-1)/2!$ とおりあることがわかる<sup>[46]</sup>。ここで、 $n!$ はスターリングの公式により以下の式で近似できる。

$$n! \sim \sqrt{2\pi n} \left(\frac{n}{e}\right)^n \quad (2.6)$$

つまり、この問題の計算量をオーダー記法で表記すれば $O(N^N)$ となり、 $n$ の増加に伴って指数関数的に計算時間が増大することがわかる。これに対し、 $O(\log N)$ や $O(N^2)$ といった計算量が見積もられる問題は「多項式時間」で解くことができる問題として区別される。ある最適化問題に対して多項式時間で厳密解が得られるアルゴリズムが存在するとすれば、これはクラス $\mathcal{P}$ に属す問題といえる。逆に、多項式時間で厳密解が得られるアルゴリズムが存在しないことを証明できれば、その問題はクラス $\mathcal{NP}$ かそれ以上の複雑性を有しているということになる。しかし、それを証明することは容易ではない。一般的には $\mathcal{P} \neq \mathcal{NP}$ と予想されているが、その証明は未だになされていない。クラス $\mathcal{P}$ の厳密な定義は、「判定問題のうち、ある決定性チューリング機械(「チューリング」は考案者の名前)によって多項式時間で解くことができる問題」とされている<sup>[47]</sup>。ここで、「判定問題」とは、0か1、あるいはYesかNoなどを出力する二値分類問題のことを指す。「決定性チューリング機械」とは、 $x$ という入力に対して $y$ という出力がなされるまでのプロセスが一意的に定まっている計算機を意味する。その対義語として「非決定性チューリング機械」という用語が存在し、これは途中の計算過程が複数に分岐する場合、これらを同時並行で実行できるという仮想的かつ理想的な仮想計算機を表す。このような計算機が存在すれば、これまで多項式時間で解けなかった問題がその時間内で解けるようになると予想されている。一般的なPCが決定性チューリング機械に該当するのに対し、量子計算機などは非決定性チューリング機械に該当する。

クラス $\mathcal{NP}$ の厳密な定義は「判定問題のうち、ある非決定性チューリング機械によって多項式時間で解くことができる問題」である。つまり、イメージ的には量子コンピュータで現実的、効率的に解けるような問題はクラス $\mathcal{NP}$ に属すといえる。また、クラス $\mathcal{NP}$ に属する問題のどれと比較しても、その難易度や複雑性が同等かそれ以上になる問題は $\mathcal{NP}$ 困難のクラスに分類される。 $\mathcal{NP}$ 完全はクラス $\mathcal{NP}$ に属する $\mathcal{NP}$ 困難のことを指す。これらの集合の関係を図 2.3 に示す。

宅配経路問題は、TSP を拡張した最適化問題ということができるので、TSP と同等かそれ以上の複雑性を持つと考えられる。また、TSP が $\mathcal{NP}$ 困難であることは先ほどの例からも明らかであるから、宅配経路問題もそれかそれ以上のクラスに属していると考えられる。つまり、宅配経路問題は解候補を全て列挙して大局的最適解を求めることは困難であり、TSP と同様にある程度探索範囲を絞ったうえで効率的に妥当な解を探索するのが現実的なアプローチと考えられる。

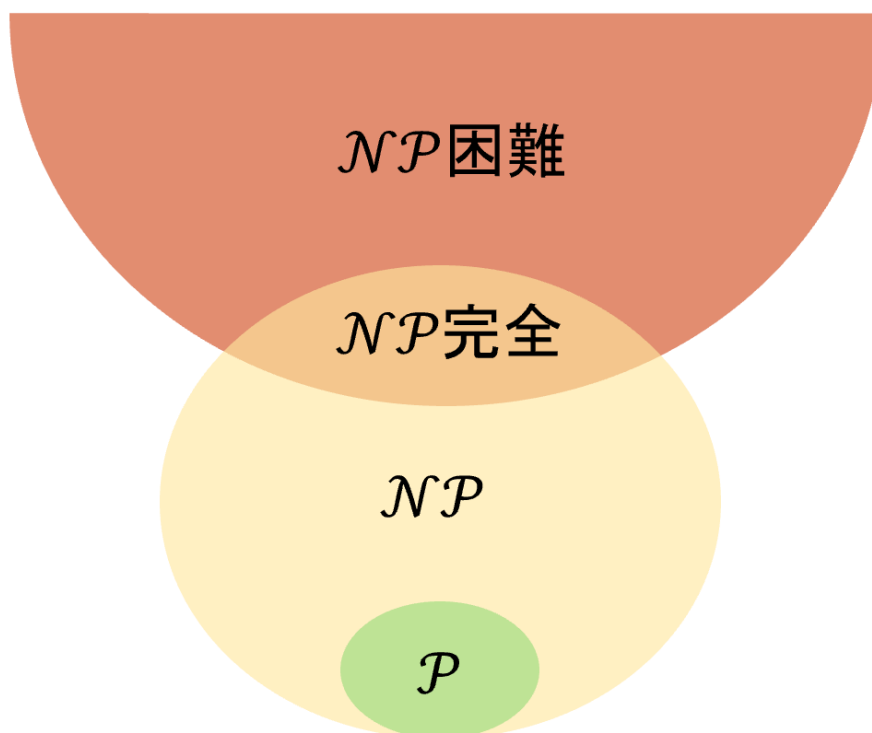


図 2.3：複雑性クラスの関係図

## 2.2 解探索アルゴリズム

最適化問題の解法には「厳密解法」と「近似解法」が存在する。厳密解法が大局的最適解の生成を目的とする一方、近似解法は前述したように限られた時間の範囲内で大局的最適解に近い解を生成することを目的とする。基本的には厳密解法による大局的最適解の生成が望ましいが、 $NP$ 困難のクラスに属するような最適化問題にはその適用は困難である。したがって、この種の問題に関してはほとんどの場合、近似解法によるアプローチが適用される。本論文では、「メタヒューリスティクス」と呼ばれる近似解法に焦点を当てる<sup>[37]</sup>。「メタ」とは、「高次の」あるいは「超」などと訳される。一方、「ヒューリスティクス」とは、「経験」という意味がある。つまり近似解法とは、「最適解を効率的に導出する上で有利となる経験則を解探索の過程に導入する手法」と解釈することができる。さらに、「メタ」という言葉を冠することで、その経験則を特定の問題に限定することなく、汎用的に応用可能にするという意味が与えられる。以下にメタヒューリスティクスの基本的原理と既存手法の代表的な例をいくつか示す。

### 2.2.1 メタヒューリスティクスの基本戦略

メタヒューリスティクスの基本戦略は、解の近傍にはより優れた解が存在するという「近接最適性原理(Proximate Optimality Principle: POP)」に基づいている<sup>[48],[49]</sup>。この原理を繰り返し利用することによって、全ての解候補を総当たりせずともある程度妥当な解を短時

間で発見することが可能となる。POP は、大局的最適性を保証するものではないが、短時間にある程度妥当な解を得られるというメリットがある。

POP を利用する上で注意すべきことは、「解構造の連続性」が確保されていることである。これは、解構造の一部を改変した際に親の形質が子にある程度継承される必要があることを示唆している。仮に、解構造の一部改変により生まれた新しい個体が親の形質を全く引き継がないものであったならば、それは単なるランダム探索になってしまう。メタヒューリスティクスを用いた解探索は、最適化問題の特徴に応じてその解表現方法を独自に定義しなければならないが、その際には必ずこの連続性について留意した解構造としなければならない。

## 2.2.2 解探索の集中化と多様化

メタヒューリスティクスを用いて解を探索する際には、POP に基づく解の集中的探索に加えて、解集団の多様性を維持することが重要である。なぜなら、POP のみに依存した解探索は、局所的最適解への収束を極端に早めてしまい、結果的にごくありふれた平凡な解しか得られない結果を招くからである。このように、他に優れた解が存在するにもかかわらず、それを発見できないまま解探索が早期に収束してしまう事態を避けるため、遺伝的アルゴリズムにおける突然変異や、解探索を敢えて改悪させる方向へ進めたりすることでそこから脱出するタブーサーチなどが提案されている。この他にも解の多様性を維持するための様々なアプローチが提案されているが、いずれも最適化問題の性質によって様々な創意工夫が必要であり、体系化された理論は存在しない。いずれにせよ、メタヒューリスティクスに基づく解探索は、近傍解の集中的探索と多様性を維持することが最も重要である。

## 2.2.3 代表的なメタヒューリスティクス

メタヒューリスティクスの研究は既に半世紀以上にわたって行われており、その間に数々の優れた解探索アルゴリズムが提案された。以下に代表的なメタヒューリスティクスを示す。

### (1) 遺伝的アルゴリズム

遺伝的アルゴリズム(Genetic Algorithm, 以下「GA」という。)は、1858年に Darwin と Wallace によって体系化された「自然選択説」という学説にインスピレーションを受けて生まれた近似解法で、1975年に Holland が発表した論文において提唱された<sup>[50],[51]</sup>。自然選択説とは、生物が世代を経るごとに周囲の環境へ適合するよう徐々に進化していく様子を述べた学説のことである。Holland は、生物が進化する様子を最適化問題の解法に応用できることを述べた。同様の試みは1960年代から既に始まっており、GA以外にも進化プログラミ

ング(Evolutionary Programming)や進化戦略(Evolution Strategy)といった解探索手法に関する研究がそれぞれ独立した立場から進められていた<sup>[52]~[55]</sup>. これらは現在、「進化型計算(Evolutionary Computation : EC)」に関する学術領域として体系化されている<sup>[56]~[61]</sup>. GA が発表された当時は、その僅か数年後には大規模な国際会議が世界各地で開催されるほど注目された計算手法である<sup>[62]</sup>.

親の形態的特徴が遺伝情報として子に継承されるプロセスには、親同士の組み合わせによって生じる「交叉・交配」によるものと、親の遺伝子の複写ミスによって偶然生まれてしまう突然変異によるものが存在する. この2つの因子により生まれる個体は、親よりもより環境に適合する場合もあれば、そうでない場合もある. 前者は厳しい生存競争を勝ち抜いて更に次の世代へと優秀な子孫を残すことができるが、逆に後者は生き残ることができずに淘汰されてしまう. 長い年月をかけてこの厳しい生存競争が繰り返されることで、最終的には初期個体よりもはるかに進化した個体を得られる. このコンセプトを最適化に応用したのが GA である<sup>[63],[64]</sup>.

つまり GA は、ランダムに生成された初期解に対して、交叉・交配及び突然変異による演算子を適用して新しい解を生成するとともに、より優れた評価値を持つ個体からさらに新しい個体を作り出すという過程を繰り返すことによって、最終的に比較的良好な解を得るメタヒューリスティクスであるといえる. この際、最適化問題ごとに固有の解表現や遺伝的操作方法、生成解の評価方法が必要となるが、逆にその枠組みさえ構築できれば様々な問題への解法として応用可能であるといえる. 図 2.4 に GA の基本的な解探索フローチャートを示す.

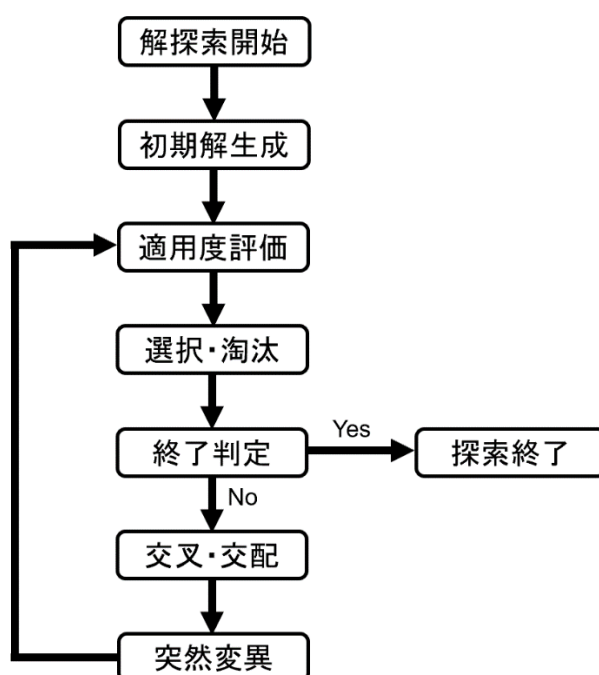


図 2.4 : GA の基本的な解探索フローチャート

なお、新たな解の生成は、交叉・交配によるものが大部分を占め、突然変異を適用することで生成する個体はごく少数とするのが一般的である。

GA は、初めて提唱された年から既に 40 年以上が経過しているものの、未だによく明らかにされていないメカニズムが数多く残されている。その意味において、GA は発展途上にある学問分野であるといえる。注意すべきことは、GA は解の探索過程を生物の進化に見立てただけに過ぎず、両者に実質的な関係は存在しないということである。つまり、生物学に関わる予備知識などは必要とせず、またそれらの知識に縛られる必要もないということである。ただし、確率的アプローチであるが故に、全ての計算条件が同一であったとしても最終的に得られる解は決定的でないという GA 特有の不安定さもあるのが欠点である。

## (2) 粒子群最適化

粒子群最適化(Particle Swarm Optimization, 以下「PSO」という.)は、1995 年に James らによって開発された解探索アルゴリズムであり、その歴史はまだ浅い<sup>[65],[66]</sup>。この探索手法は、鳥の群れが餌を探す際に群れ全体で位置情報を共有するという仮定に基づいたアイデアであることから、「Particle(微小粒子)」と「Swarm(群れ)」の 2 つの言葉を合わせた名が与えられた。解探索の基本的な流れは、まず各粒子にランダムな初期位置ベクトルと初速度ベクトルを与え、位置が更新されるごとに各粒子の現在地における評価値を計算する。このとき、それまでに探索した中で最も評価値の優れた解が存在した位置を粒子ごとに記憶する。また、全ての粒子の中で最も優れた解が存在していた位置についても記憶する。この 2 種類の最良解の位置情報に基づいて次に移動する際の速度ベクトルが粒子ごとに決定されるので、必然的に次の位置ベクトルも定まる。2 種類の最良解の位置は、次ステップ以降もこれらを上回る解が発見される度に更新され続ける。このように、PSO では各粒子がこれまで辿った道程で発見した最良解と粒子全体での最良解の位置ベクトルを用いることで、探索の集中化と多様化を実現している。一方で、PSO の適用にはいくつかの課題も残されている。例えば TSP の解法に PSO を用いる場合、解となる巡回経路を位置ベクトルとしてどう表現するかといった問題がある。既にこの問題についてはいくつかの研究が報告されているものの、TSP 以外の最適化問題でも同様の問題が提起させるため、その汎用性という面において課題が多い手法といえる。

## (3) タブーサーチ

タブーサーチ(Tabu Search)は、1986 年に Fred Glover らによって提案された手法で、人間が持つ記憶能力にその着想を得ている<sup>[67]-[70]</sup>。タブーサーチの基本的原理は、「戦略的に悪い選択をすることは、行き当たりばったりでよい選択をするよりも多くの情報をもたらす可能性がある」というものである。具体的には、解 $x$ とその近傍解 $y$ の関係が $F(x) \geq F(y)$ と

なるとき、解 $x$ を近傍解 $y$ へと遷移させるとともに、その過程を記憶として保存する。もし生成した近傍解に $F(x) < F(y)$ の関係となる解しか存在しない場合は、改悪幅が最小となる近傍解 $y$ へと遷移させ、同じく記憶する。記憶された領域は「タブーリスト」と呼ばれ、もし解探索の過程で近傍解 $y$ の方が解 $x$ より優れている場合でも、タブーリストに掲載されている $y$ に解 $x$ が更新されることはない。このように、タブーサーチ法はタブーリストを適切に活用することで近傍解の集中的探索と多様性の維持を実現している。一方で、タブーサーチには「サイクリング」と呼ばれる固有の課題が内在している。これは、局所解からの脱出が成功したとしても、いくつかの解を経由して再び同じ解に辿り着いてしまうループ状態に陥ることを指す用語である。これは、タブーリストの中身が徐々に更新されるうちに古い記憶が消去されることに起因している。したがって、一度生成した局所解は一貫して解集合から除外するなどの処置が必要になると考えられる。

#### (4) 焼きなまし法

焼きなまし法(Simulated Annealing, 以下「SA」と省略)は、1983年に Scott らによって提案された近似解法である<sup>[71],[72]</sup>。彼らは、金属が高温状態から低温状態へゆっくりと冷却加工されると、その構造が破壊されにくい強度を持つという特性に着目し、これを解探索に応用できないかを検討した。SAでは、解 $x$ とその近傍解 $y$ の関係が $F(x) \geq F(y)$ となるとき、解 $x$ を近傍解 $y$ へと遷移させる。一方、それ以外の場合には「受理基準」と呼ばれる確率に基づいて近傍解 $y$ への遷移を決定する。換言すれば、場合によっては現状を改悪する方向へと解探索が進むことも許容するということである。こうすることで、POPに基づく集中的探索と多様性の維持を目指す。

SAでは「温度」と呼ばれるパラメータ値が解探索のあり様を決定づける重要な指標として機能する。解探索初期では温度が高温状態にあると仮定するため、場合によっては大幅な状態の改悪を許容することもあるが、解探索が進むにつれて低温状態へと遷移していくので、その改悪の許容幅も徐々に縮小していき、最終的には一切の改悪を認めない状態へと収束する。この状態に移行するまでに発見した最良解を最適解として採用する。この解探索アルゴリズムを設計する際には、初期状態、温度の冷却基準、解探索の反復回数及び終了条件という4つのパラメータを事前に決定しなければならず、その設計基準も問題ごとに一様ではないので、妥当な解を得られるようなパラメータ調整が完了するまでにかなりの試行錯誤が必要になると予想される。

#### (5) アントコロニー最適化法

アントコロニー最適化法(Ant Colony Optimization, 以下「ACO」と省略)は、1996年に Marco らが提案した解探索アルゴリズムであり、その着想はアリの巣から餌場までの経路

が徐々に短縮化されていく様子から得られたとされる<sup>[73],[74]</sup>。解探索のキーとなるのは「フェロモン」と呼ばれる匂い物質である。アリは自ら言葉を発することで仲間に情報を伝達することができない。また、極度の近視であることから、視力に頼った情報の獲得にも不自由である。しかしながら、彼らは自分が経由した道にフェロモンを残すことによって仲間同士と協力して餌を探ることができる。フェロモンは、時間の経過とともに蒸発してやがては消えてしまう特徴を持ち、アリはこのフェロモンが強い経路ほど好んで選択する傾向がある。この 2 つの特性を組み合わせることで、巣から餌場までの最短距離を生成することが可能である。例えば、巣から餌場までの距離が長いほど移動に時間を要するので、その間に蒸発するフェロモンの量も増える。一方、距離が短いほどアリが往来する頻度も高くなるため、蒸発するフェロモンの量は少なくなるばかりか、その経路を往来するアリの数も増加するので、逆にその経路上のフェロモンは強化されていくと予想される。結果的にアリは距離が短い方の経路を好んで選択するようになる。ACO は、他の近似解法と同様フェロモンを数値化する過程においていくつかのパラメータ設定を事前に行う必要があるものの、条件さえ整えば TSP の解法として十分活用できることが知られている。

#### 2.2.4 近似解法を選択

メタヒューリスティクスを用いた解法は、特定の問題に対しては優れた解探索能力を示すが、全ての複雑性クラスに属する最適化問題に対しては平均的な性能しか発揮できないことが知られている。換言すれば、汎用的に優れた性能を発揮する万能な近似解法は存在しないということである。これを「ノーフリーランチ定理」という<sup>[75]</sup>。これは、最適化の対象とする問題がそれぞれ固有の特性をもち、その全てに共通する経験則や戦略は存在しないことを示唆している。したがって、上記で示したどのメタヒューリスティクスを選択しようとも、対象とする最適化問題に対して解の集中的探索と多様性維持が担保されれば、ある程度妥当な解を得ることが期待できるといえる。そこで、本論文では上記で挙げたメタヒューリスティクスの中では最も歴史が古く、かつ様々な問題への適用が可能とされる GA を制約付き多目的最適化問題の近似解法として選択する。

### 2.3 多目的最適化手法

多目的最適化手法には大きく分けて 2 つのアプローチが存在する。1 つは、多目的最適化問題を単目的最適化問題に変換することで、選好解をただ一つだけ求めるアプローチである。このような手法を「単一目的化手法」、もしくは「スカラー化手法」と呼ぶ。もう一方のアプローチは、パレートフロンティアそのものを求める手法である。このような手法を本論文では「パレート最適化手法」と呼ぶこととする。前者に関しては、重み係数法や $\epsilon$ 制約法、辞書式配列法などがその例として挙げられる<sup>[76]-[81]</sup>。後者については VEGA(Vector

Evaluated Genetic Algorithm)や MOGA(Multi-Objective Genetic Algorithm), NSGA- II (Non-dominated Sorting Genetic Algorithm- II ), SPEA2(Strength Pareto Evolutionary Algorithm2)などの手法が挙げられる<sup>[82]-[91]</sup>. これら 2 種類のアプローチの利点と欠点を整理したものを表 2.1 に示す.

表 2.1 : 2 つの多目的最適化手法の利点と欠点

	単一目的(スカラー)化手法	パレート最適化手法
利点	<ul style="list-style-type: none"> <li>● アルゴリズムの実装がシンプルである.</li> <li>● 選好解がただ 1 つに定まるので, 数ある解候補から選好解を選択する必要がない.</li> <li>● 目的関数の数が増加しても計算負荷が増大しにくい.</li> </ul>	<ul style="list-style-type: none"> <li>● 1 回の解探索で多数のパレート解を生成できる</li> <li>● 複数の選択枝から意思決定者によって適当な選好解を選択できる.</li> </ul>
欠点	<ul style="list-style-type: none"> <li>● 1 回の解探索で 1 つの最適解しか得られない.</li> <li>● パレートフロンティアの形状次第では, 意思決定者が所望する選好解を得られない可能性がある.</li> <li>● 重み係数などのパラメータ調整が必要である.</li> </ul>	<ul style="list-style-type: none"> <li>● 解探索アルゴリズムがスカラー化手法より複雑である.</li> <li>● 次世代へ継承する解の選別に多くの計算資源を必要とする.</li> <li>● 目的関数が 4 個以上になると最適化計算が困難になる.</li> </ul>

この表に示したように, 2 つのアプローチは共に利点と欠点を併せ持っており, そのどちらも宅配経路問題を解く手法として完全に適しているとはいえない. また, これまで提案されてきた GA を用いた多目的最適化手法のうち, 実行可能解の生成と多目的最適化を同時に実現する解法に関する例はほとんど報告されていない.

以上の議論を踏まえ, 次章以降では実行可能解の生成と多目的最適化を同時に実現する GA ベースの新しい多目的最適化手法について提案する.

## 第3章 暫定理想点(PIP)法の提案

### 3.1 はじめに

本章では、式(2.1)で示した制約付き多目的最適化問題の解法として GA をベースとした「暫定理想点(Provisional-Ideal-Point(以下、「PIP」と表記する。))法」を提案する<sup>[92],[93]</sup>。以下にその詳細を示す。

### 3.2 PIP法の解探索プロセス

PIP法の解探索プロセスは2段階に分けられる。第1段階では実行可能解の生成を中心に実行し、第2段階では多目的最適化のための解探索を実行する。以下にその詳細を示す。

#### <第1段階> 実行可能解の生成

制約付き多目的最適化問題の選好解として満たすべき最低条件は、与えられた全ての制約条件を満足する実行可能解であることである。図3.1(a)及び図3.1(b)は与えられた制約条件 $g_j(x)$ を棒グラフで可視化した概念図である。この2つの図において、 $g_j(x) > 0$ となる領域は赤色、 $g_j(x) \leq 0$ となる領域は青色で色分けをしている。実行可能解は、全ての制約条件が $g_j(x) \leq 0$ となる解であるから、 $g_j(x)$ の棒グラフは図3.1(a)に示したように全て青色の領域に含まれるはずである。一方、図3.1(b)に示したように $g_j(x)$ の棒グラフが1つでも赤の領域に存在する場合は実行不可能解とみなされる。

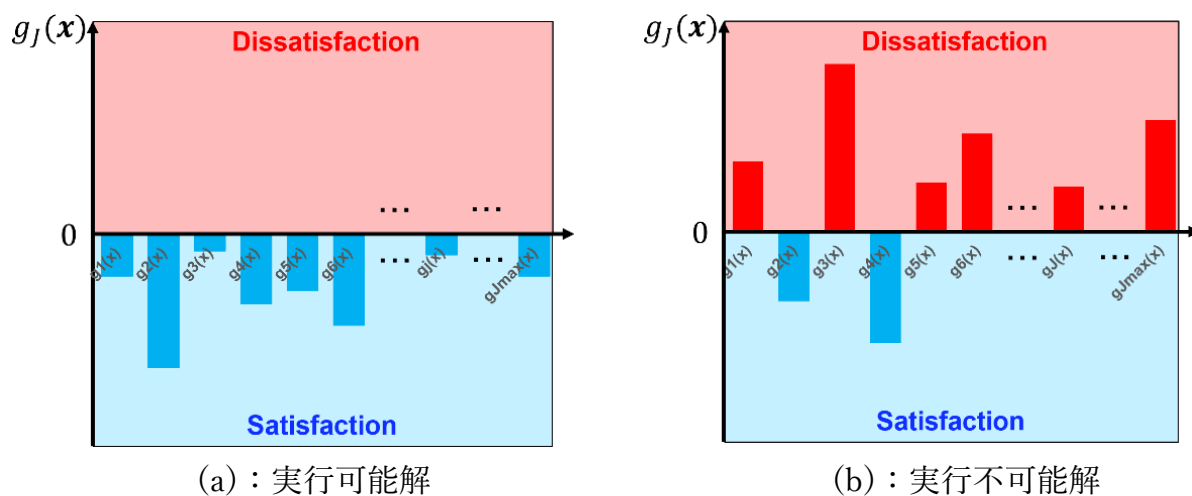


図 3.1 : 実行可能解と実行不可能解の概念図

PIP法の第1段階では、上記で言及した最低条件を満足するために全ての計算資源を実行可能解の探索のみに投入する。ここで、赤の領域に含まれる $g_j(\mathbf{x})$ の値を合算することで、「ペナルティ関数」を式(3.1)のように定義する。

$$P(\mathbf{x}) = \sum_{j=1}^m P_j(\mathbf{x}) \quad \text{s.t.} \quad \begin{cases} P_j(\mathbf{x}) = g_j(\mathbf{x}), & g_j(\mathbf{x}) > 0 \\ P_j(\mathbf{x}) = 0, & g_j(\mathbf{x}) \leq 0 \end{cases} \quad (3.1)$$

$P(\mathbf{x})$ は制約条件を逸脱した偏差量の総和であるから、実行可能解の場合は $P(\mathbf{x}) = 0$ となるはずである。この関数の特徴は、 $P_j(\mathbf{x})$ 固有の単位を無視し、全てスカラー値として扱う点である。一般的には単位の異なる値同士を四則演算の対象とすることはできないが、第1段階における解探索の最終目標は $P(\mathbf{x}) = 0$ となる実行可能解の生成であることから問題は生じない。その上、異なる単位同士を一律に扱えるため、実問題を含めてその適用が容易である利点がある。 $P(\mathbf{x})$ を最小化させるように解探索を進め、最終的に実行可能解が1つでも得られた段階で第2段階のプロセスへと移行する。もし、一定世代数を経過しても実行可能解が得られない場合は、制約条件の緩和などを含め、問題設定の見直しを検討することが必要と判断できる。

<第2段階> 多目的最適化

得られた実行可能解の集合について、式(3.2)で表現される実行可能解 $\mathbf{x}_F$ 及び $\mathbf{x}_{F'}$ をそれぞれ定義する。

$$0 \leq F_l(\mathbf{x}_{F'}) \leq F_l(\mathbf{x}_F) \quad (\mathbf{x}_{F'} \in \mathcal{S}, \forall \mathbf{x}_F \in \mathcal{S}) \quad (3.2)$$

このとき、 $\mathbf{x}_{F'}$ を $F_l(\mathbf{x})$ の「暫定最適解」と呼び、 $F_l(\mathbf{x}_{F'})$ を $F_l(\mathbf{x})$ の「暫定最適値」と呼ぶ。特に、 $F_l(\mathbf{x}_{F'})$ に関しては式(3.3)のように表記する。

$$F_{l_{pro}} = F_l(\mathbf{x}_{F'}) \quad (3.3)$$

$F_{l_{pro}}$ は解探索の過程でより優れた解が生成される度に更新される。ここで、各目的関数の評価値がそれぞれ $F_{l_{pro}}$ に最適化される仮想的な解を「暫定最適解」と呼ぶ。また、暫定最適解を $F_{l_{pro}}$ で正規化することで得られる座標点を「暫定理想点」と呼ぶ。これに対し、生成した解を $F_{l_{pro}}$ で正規化することで得られる座標点を「解点」と呼ぶ。これら2つの座標点はそれぞれ式(3.4)及び式(3.5)で定義される。

$$C_{ideal} = \left[ \frac{F_{1_{pro}}}{F_{1_{pro}}}, \dots, \frac{F_{l_{pro}}}{F_{l_{pro}}}, \dots, \frac{F_{M_{pro}}}{F_{M_{pro}}} \right] \\ = [1, \dots, 1, \dots, 1] \quad (3.4)$$

$$C_{sol}(\mathbf{x}_F) = \left[ \frac{F_1(\mathbf{x}_F)}{F_{1_{pro}}}, \dots, \frac{F_l(\mathbf{x}_F)}{F_{l_{pro}}}, \dots, \frac{F_M(\mathbf{x}_F)}{F_{M_{pro}}} \right] \\ = [F'_1(\mathbf{x}_F), \dots, F'_l(\mathbf{x}_F), \dots, F'_M(\mathbf{x}_F)] \quad (3.5)$$

この2点間のユークリッド距離を $D_{is}(\mathbf{x})$ と表記する．この $D_{is}(\mathbf{x})$ を式(3.6)で示すように最小化する解を探索する．その様子を示した概念図が図3.2である．

$$\begin{cases} \min_{\mathbf{x}} F(\mathbf{x}_F) = \min_{\mathbf{x}_F} D_{is}(\mathbf{x}_F) = \min_{\mathbf{x}_F} \|C_{ideal} - C_{sol}(\mathbf{x}_F)\| = \min_{\mathbf{x}_F} \left( \sum_{l=1}^M (1 - F'_l(\mathbf{x}_F))^2 \right)^{\frac{1}{2}} \\ \text{subject to } P(\mathbf{x}_F) = 0 \end{cases} \quad (3.6)$$

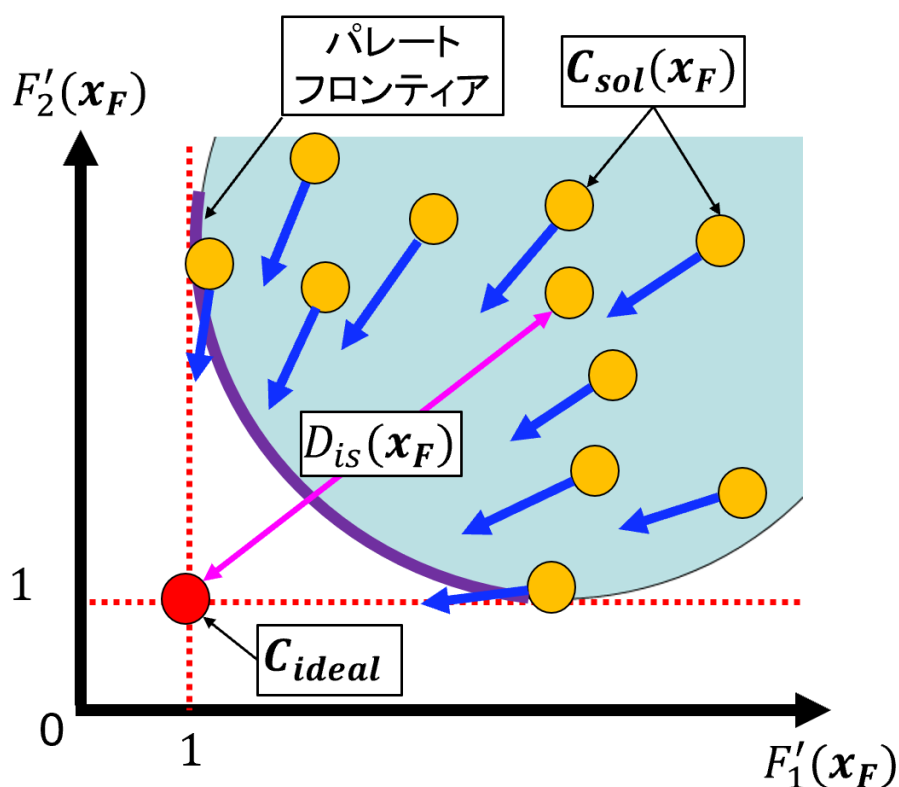


図 3.2 : 解探索第 2 段階の概念図

最終的に，図 3.3 で示すようにパレートフロンティア上，もしくはその近傍に実行可能解を生成することが期待できる．

このように，全ての目的関数を最適化する仮想的な解を理想点として定義し，それに最も近い実行可能解を探索することでただ1つの選好解を得る手法は，「希求点を用いた多目的最適化手法」で言及した解探索手法を準用した概念である<sup>[94]</sup>．

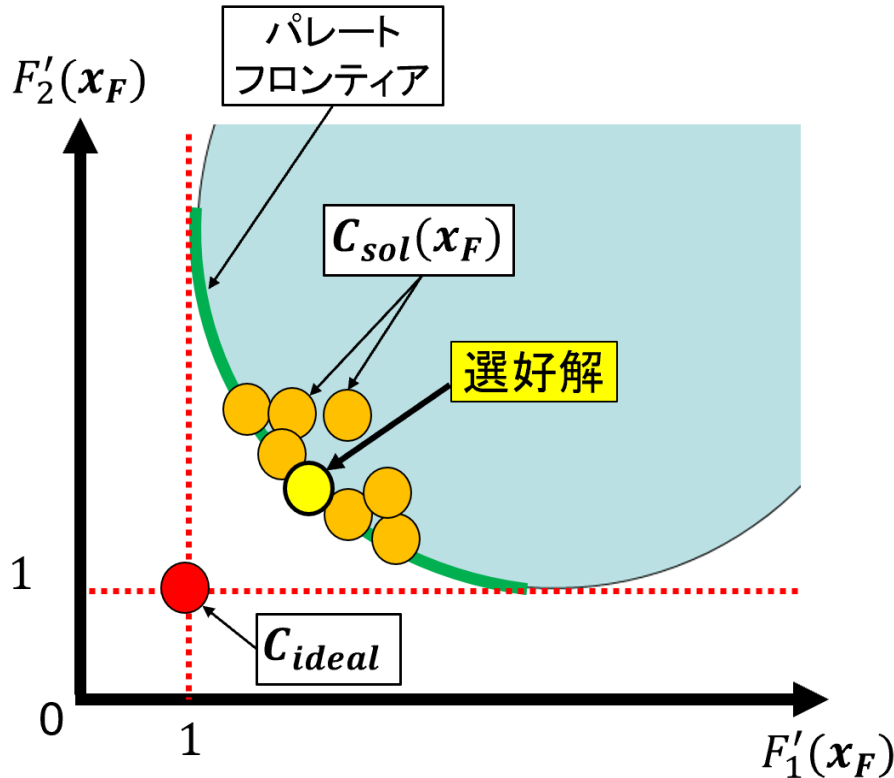


図 3.3：パレートフロント上に生成される選好解の概念図

PIP 法では、 $C_{ideal}$ に最も近い $C_{sol}(x_F)$ を意思決定者の選好解とみなすことに特徴がある。このアプローチは、パレートフロンティア上に存在する数多くの解を同時に生成しようとする従来の方法とは異なる考え方である。仮に、数多くの目的関数を考慮した多目的最適化問題を解く場合には、明らかに PIP 法の方が有利と考えられる。なぜなら、先行研究で示したアプローチは、目的関数の数が増えるほどパレート解の候補も莫大な数となり、事実上パレートフロンティア上の解を全てカバーすることは不可能だからである。一方、PIP 法であれば、最初からただ一つの選好解の生成のみに計算資源を集中させることができるので、例えば目的関数の数が多くなろうとも計算負荷の増加は限定的である。

上記で示した実行可能解の生成と多目的最適化に関する解探索の流れをステップごとに整理したものを以下に示す。

#### <Procedure 1>(実行可能解の生成)

- <Step 1> 世代 $g \leftarrow 0$ とする。
- <Step 2> ランダムに生成した $s$ 個の初期個体 $x$ の集合を $p$ とし、世代 $g \leftarrow 1$ とする。
- <Step 3>  $P(x)$ の値が小さい順に上位 $E_n$ 個の $x$ を選別する。選別されなかったそれ以外の個体は $p$ より除外する。
- <Step 4>  $E_n$ 個のエリート個体とその遺伝的操作により生成した $(s - E_n)$ 個の個体集合を $p$ として、 $g \leftarrow g + 1$ とする。

- <Step 5>  $P(\mathbf{x}) = 0$ となる $\mathbf{x}$ が $p$ 内に1つ以上存在する場合, それらの個体を  $\mathbf{x}_F = \mathbf{x}$ として <Procedure 2>へ進む.
- <Step 6>  $g > G_{max}$ ならば<Step 7>に進む. そうでなければ<Step 3>へ戻る.
- <Step 7> 実行可能解の生成は困難とみなし, <Procedure 1>を終了して問題設定を再考する.

### <Procedure 2>(多目的最適化)

- <Step 8>  $F_l(\mathbf{x}_F') \leq F_l(\mathbf{x}_F)$  ( $\mathbf{x}_F' \in p, \forall \mathbf{x}_F \in p$ )となる $\mathbf{x}_F'$ を選択し,  $F_{l_{pro}} \leftarrow F_l(\mathbf{x}_F')$ とする.
- <Step 9>  $D_{is}(\mathbf{x}_F)$ の値が小さい順に上位 $E_n$ 個の $\mathbf{x}_F$ を $p$ より選別する.  $E_n$ 個の $\mathbf{x}_F$ が存在しない場合は, 不足分を他の $\mathbf{x} \{ \mathbf{x} \in p | P(\mathbf{x}) > 0 \}$ より $P(\mathbf{x})$ の値が小さい順に充填する. その後, 選別された以外の個体を $p$ より除外する.
- <Step 10>  $E_n$ 個のエリート個体とその遺伝的操作により生成した $(s - E_n)$ 個の個体の集合を $p$ として,  $g \leftarrow g + 1$ とする.
- <Step 11>  $g > G_{max}$ であれば<Step 12>に進む. そうでなければ<Step 8>に戻る.
- <Step 12>  $D_{is}(\mathbf{x}_F^*) \leq D_{is}(\mathbf{x}_F)$  ( $\mathbf{x}_F^* \in p, \forall \mathbf{x}_F \in p$ )となる $\mathbf{x}_F^*$ を選好解とする.

## 3.3 PIP 法の評価方法

PIP法は制約付き多目的最適化問題を解くための手法であるが, このアルゴリズムの妥当性を単独で評価することが可能なベンチマーク問題は存在しない. そこで, 制約付き単目的最適化問題として知られる Global Optimization Test Problems(「GO問題」という.)と, 制約なし多目的最適化問題として知られる DTLZ問題を用いる<sup>[95]~[97]</sup>. GO問題では, 実行可能解生成後, その状態を保持したままその問題の最適解を探索できるかどうかを検証する. DTLZ問題については, 適当な制約条件を独自に付与してからPIP法を適用し, 制約条件内でパレート解を生成できるかどうかを検証する. これら2つの結果から PIP法の有用性を評価する. なお, ベンチマーク問題は, 既存手法と新手法の優位性を比較するために用いられるのが一般的ではあるが, 今回は PIP法による実行可能解の生成と多目的最適化が同時に実現可能かどうかを検証するために使用するので, 既存手法との比較は行わない.

GO問題には複数のベンチマーク問題が用意されているが, 過去の研究ではほぼ全ての種類の問題を用いて提案手法の妥当性を評価しているため, 本論文でもその慣例に従い計10種類のGO問題(「go01~go10」と表記する.)を使用する. 一方, DTLZ問題にも複数のベンチマーク問題が用意されているが, 本論文ではパレートフロンティアの形状がそれぞれ直線状, 円弧状及び不連続形状となる「DTLZ1」, 「DTLZ3」及び「DTLZ7」のみを用いるものとし, 定義式は異なるが同じ形状を持つその他のDTLZ問題については使用しない.

以下に, PIP法を各ベンチマーク問題に適用するための方法について示す.

### 3.3.1 解表現方法

GO 問題及び DTLZ 問題の設計変数  $x$  は全て実数であるから、その解は図 3.4 で示すようにバイナリ形式(配列の右端を 1 桁目とする)で表現することができる。

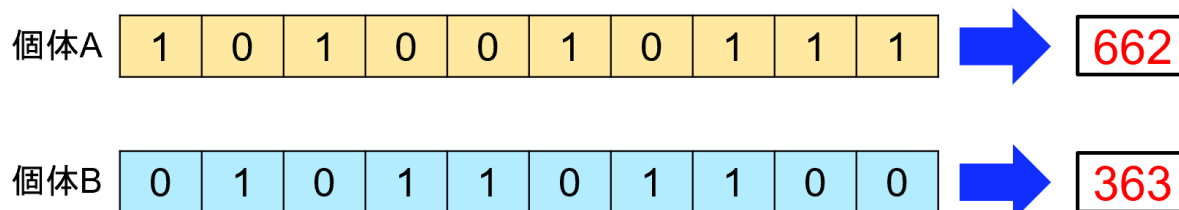
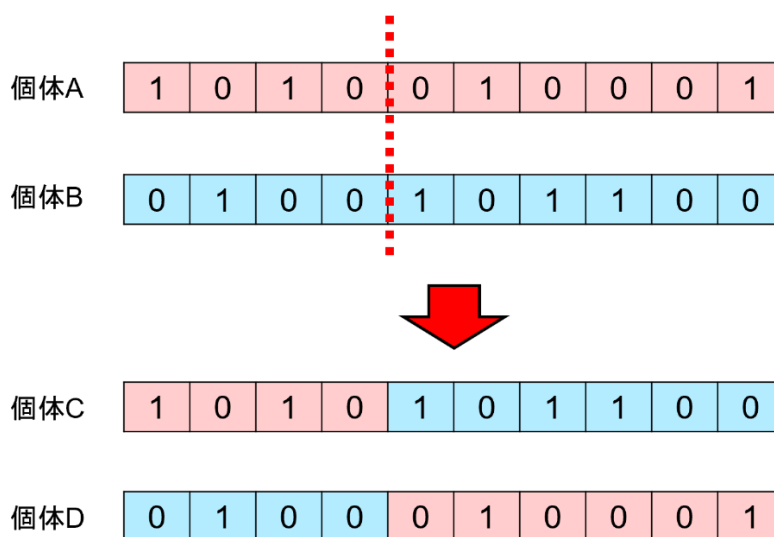


図 3.4: バイナリ形式による実数の表現

上図は、10 列のバイナリ数列を用いて 0~1023 までの間の数字の一部を表現した例を表している。もし、 $n$ 列のバイナリ数列を用いれば、0から $2^n - 1$ までの数値を表現することができる。ここで、 $0 \leq x \leq 1$ の定義域が与えられた $x$ を仮定すると、図 3.4 で示した解表現に関しては、0から $1/1023$ 刻みで1までの任意の実数を表現することができる。

### 3.3.2 交叉・交配

交叉・交配は、図 3.5(a)~(c)に示すように 2つの個体の解構造のそれぞれ一部ずつ組み合わせることによって新しい解を生成する。図 3.5(a)は 1 か所、図 3.5(b)は複数か所に解構造を分割して互いに掛け合わせる。図 3.5(c)はマスクと呼ばれる 0 と 1 とで表現される配列を新たに生成し、その配列に基づいて 2つの解構造を組み合わせるものである。



(a) : 1 点交叉

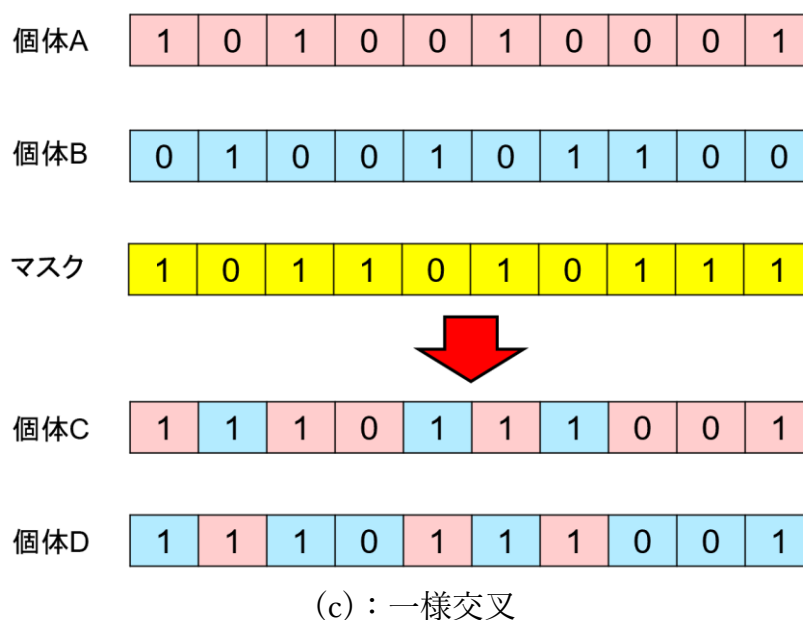
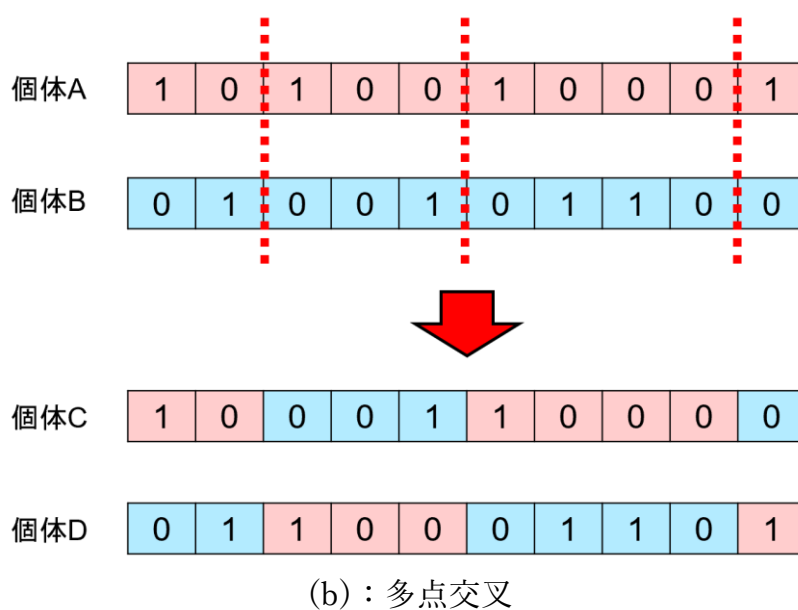


図 3.5 : 交叉・交配による遺伝的操作

### 3.3.3 突然変異

突然変異は、解構造を一部変化させることによって新しい解を生成するものである。図 3.6(a)は、解構造の 1 部を反転することで新たな個体を生成する一例を表している。図 3.6(b)は、解構造の 1 部を抽出し、任意の箇所へ倒置する解操作を表す。図 3.6(c)は、解構造内の配列の 1 つを指定し、その配列が 0 ならば 1, 1 ならば 0 に入れ替えることで新たな個体を生成する遺伝的操作を表す。

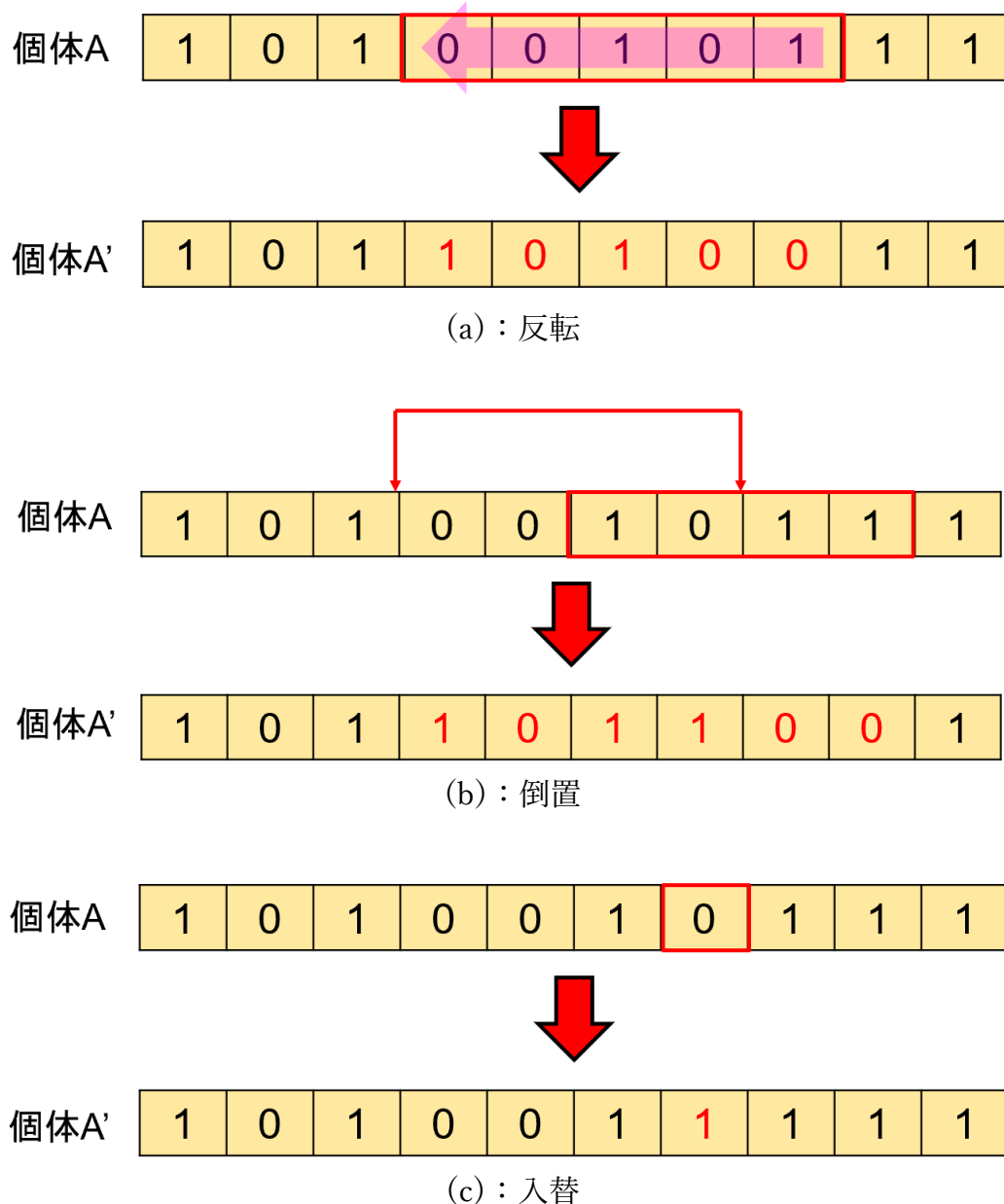


図 3.6 : 突然変異による遺伝的操作

### 3.3.4 選択・淘汰

母集団内に存在する個体のうち、優れた評価値をもつ個体から順に上位 $E_n$ 個の個体を次世代に継承する操作を繰り返すことによって 2.2.1 節で言及した POP(近接最適性原理)に基づく最適解の生成を実現する。 $E_n$ の値の設定の仕方に関りはないが、一般的には解集合全体の1割から2割程度の数を設定するのが通例である。

### 3.3.5 解探索フローチャート

図 3.7 は、PIP 法の解探索プロセスをフローチャート化したものである。

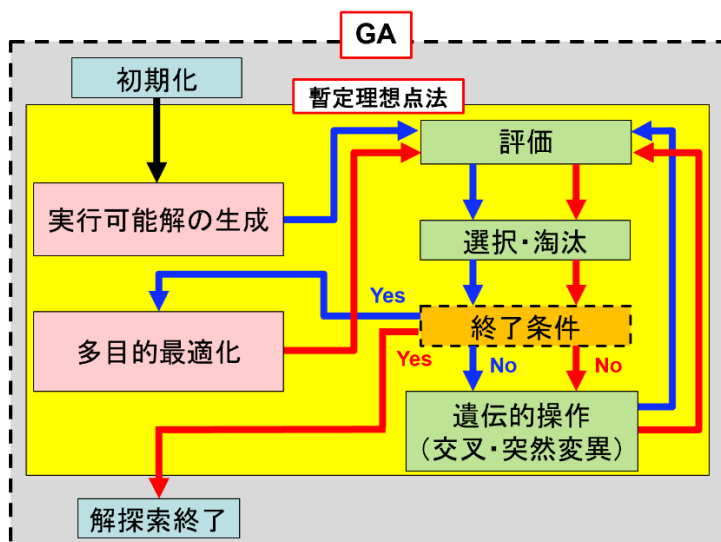


図 3.7 : PIP 法による解探索フローチャート

この図において、「遺伝的操作」の箇所では、淘汰された解の数だけ新しい解が生成されるプロセスが行われる。この工程で生まれる新しい解は、保存されるエリート個体のコピーに 95%の確率で交叉交配を適用し、残り 5%の確率で突然変異を適用することで生成するものとする。この 2つの確率は、第 2 章で示した GA の通例に基づいて設定した。

### 3.4 GO 問題を用いた PIP 法の妥当性評価

GO 問題には、その定義式が線形あるいは非線形の多項式、三角関数やべき乗を含む式などいくつかの種類が存在するが(Appendix A 参照)、ここでは go01 から go10 までの計 10 種類を取り上げる。表 3.1 は、PIP 法を GO 問題に適用した際の計算結果を示す。

表 3.1 : PIP 法を GO 問題に適用した計算結果

問題	既知の最適解	生成解	誤差率
go01	-15.0000	-14.8977	0.60%
go02	0.8036	0.6975	13.20%
go03	1.0000	0.9978	0.20%
go04	-30665	-30636	0.09%
go05	5126.5	5126.5	0.00%
go06	-6961.8	-6925.7	0.52%
go07	24.3062	16.7100	31.25%
go08	0.0958	0.0958	0.00%
go09	680.63	681.30	0.10%
go10	7049.3	7127.1	1.10%

図 3.8～図 3.17 は、PIP 法を go01～go10 に適用した場合における  $F(\mathbf{x})$  及び  $P(\mathbf{x})$  の変移を表すグラフである。各図で示したグラフは、それぞれ  $P(\mathbf{x}) > 0$  の範囲に関してはマゼンタ色、 $P(\mathbf{x}) = 0$  の範囲に関しては青色で表示しており、 $F(\mathbf{x})$  のグラフに併記された赤色の直線は、各ベンチマーク問題の最適値を表す。以下にそのグラフと考察結果を示す。

(1) go01

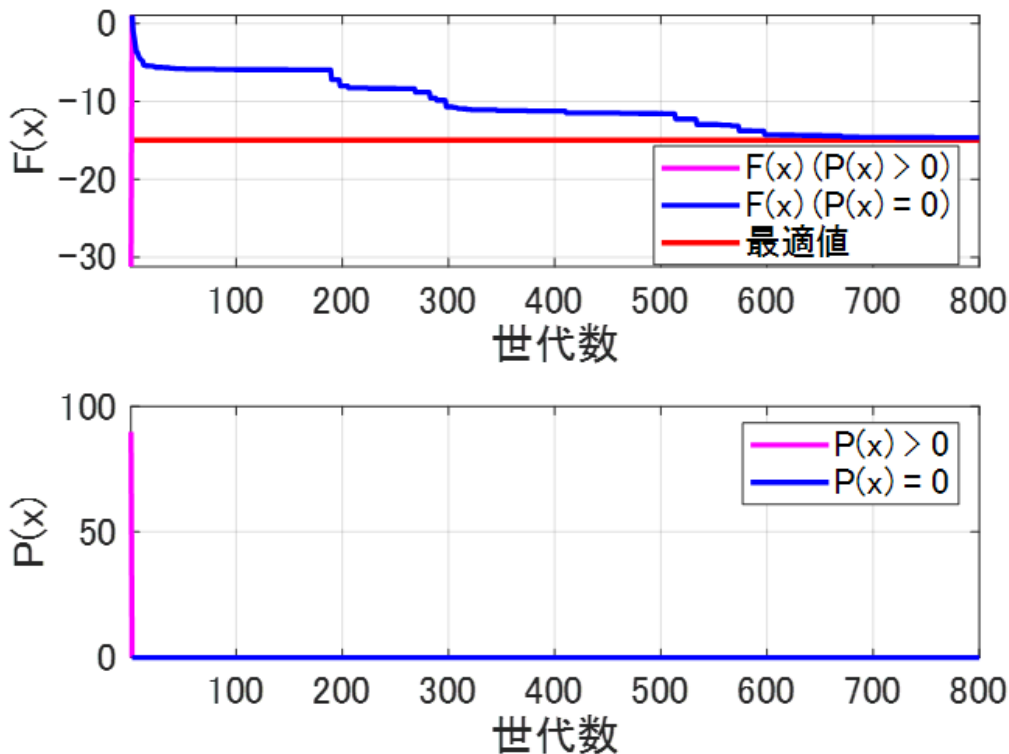


図 3.8 : go01 への適用結果

上図は、PIP 法を用いて go01 を解いた計算結果を表している。  $P(\mathbf{x})$  のグラフのマゼンタで表された部分が解探索直後に青色に変化していることから、初期の世代数で  $P(\mathbf{x}) = 0$  となる実行可能解の生成に成功していることがわかる。一方、  $F(\mathbf{x})$  の解探索直後のグラフの様子に着目すると、go01 が最小化問題であるにもかかわらず実行可能解が生成されるまでの間にその評価値が  $-30$  付近から  $0$  以上まで急増している様子を確認できる。これは、PIP 法の第 1 段階における解探索が、  $F(\mathbf{x})$  の評価値の変化を度外視して実行可能解の探索のみを優先した結果だと考えられる。  $P(\mathbf{x})$  の青色で示された部分のグラフは、その後解探索が終了するまでの間  $P(\mathbf{x}) = 0$  の状態を維持したまま遷移している一方、  $F(\mathbf{x})$  のグラフについては、青色に変化してからは一度も増加することなく赤色の直線で示された最適値に漸近していき、最終的に両者はほぼ重なり合っている様子を確認できる。表 3.1 に示された最適解と生成解の誤差率は  $0.60\%$  であることと、上記で示した解探索の経過から、PIP 法が想定されたとおりの解探索を実現していることが示された。

(2) go02

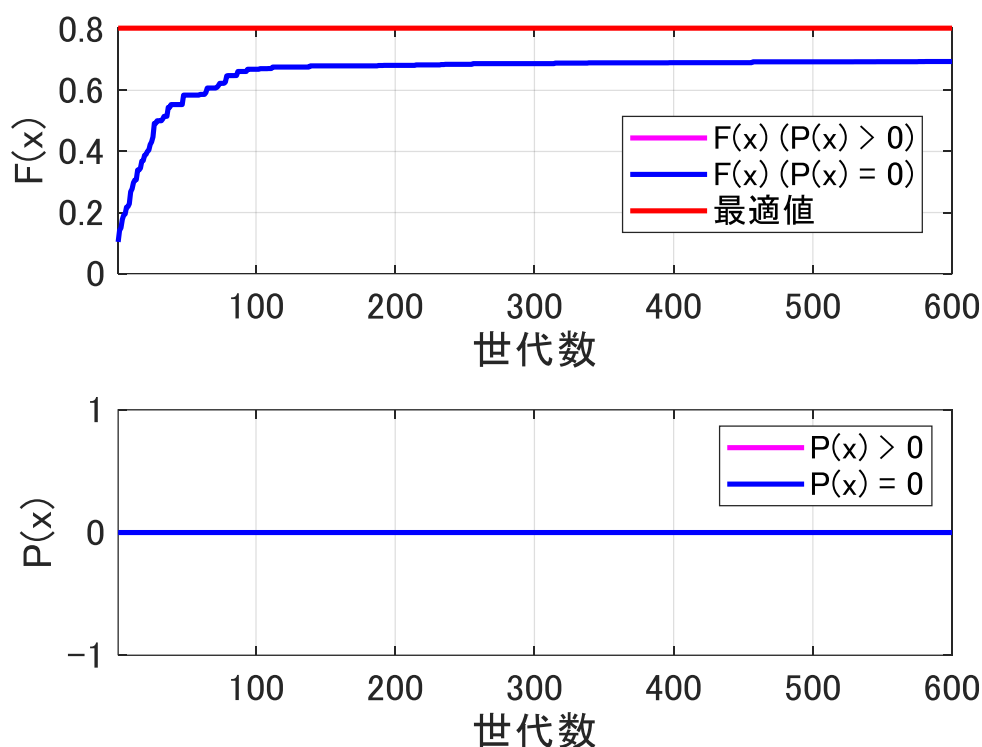


図 3.9 : go02 への適用結果

上図は、PIP 法を用いて go02 を解いた計算結果を表している。  $P(\mathbf{x})$  のグラフのマゼンタで表された部分がないのは、初期解を生成した段階で実行可能解が得られたためである。つまり、go02 に関してはその実行可能解を得ること自体は容易であることが推測される。またこの問題は最大化問題であるから、  $F(\mathbf{x})$  のグラフが徐々に増加していき、世代数が 100 を超えた辺りで解探索が停滞している様子が見られる。この間、  $P(\mathbf{x})$  の値は 0 の状態を維持しているので、PIP 法が実行可能解の中から go02 の評価関数を最大化させる解を探索しているといえる。表 3.1 より、最終的に得られる最良解と最適解の誤差率は 13.20% となっており、go01 の場合と比較すればあまり良い精度は得られなかったものの、解の探索過程を確認する限り PIP 法に期待される解探索を実現しているとみなすことは可能である。

(3) go03

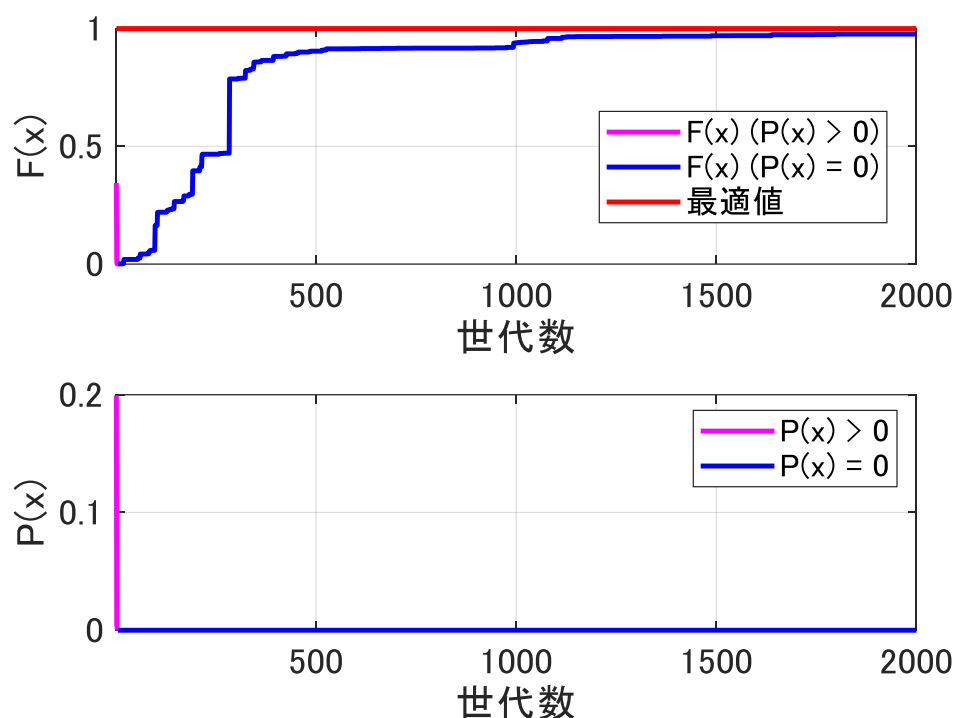


図 3.10 : go03 への適用結果

上図は、PIP 法を用いて go03 を解いた計算結果を表している。  $P(x)$  のグラフのマゼンタで表された部分が解探索直後に青色に変化していることから、初期の世代数で  $P(x) = 0$  となる実行可能解の生成に成功していることがわかる。一方、解探索直後の初期世代における  $F(x)$  のグラフに着目すると、go03 が最大化問題であるにもかかわらず、実行可能解が生成されるまでの間は 0.3 から 0 まで急減していることがわかる。これは、go01 の場合の同様に、PIP 法の第 1 段階における解探索が、 $F(x)$  の評価値の変化を度外視して実行可能解の探索のみを優先した結果だと考えられる。 $P(x)$  の青色で示された部分のグラフは、その後 2000 世代に到達するまでの間、 $P(x) = 0$  の状態を維持したまま遷移している。一方の  $F(x)$  のグラフについては、青色に変化してからは一度も減少することなく一貫して赤色の直線で示された最適値に漸近していき、最終的に両者が肉薄している様子を確認できる。表 3.1 に示された最適解と最終的に得られた生成解の誤差率は 0.20% であることと、上記で示された解探索の経過より、この問題についても PIP 法が想定されたとおりの解探索が実現されているとみなしてよいといえる。

(4) go04

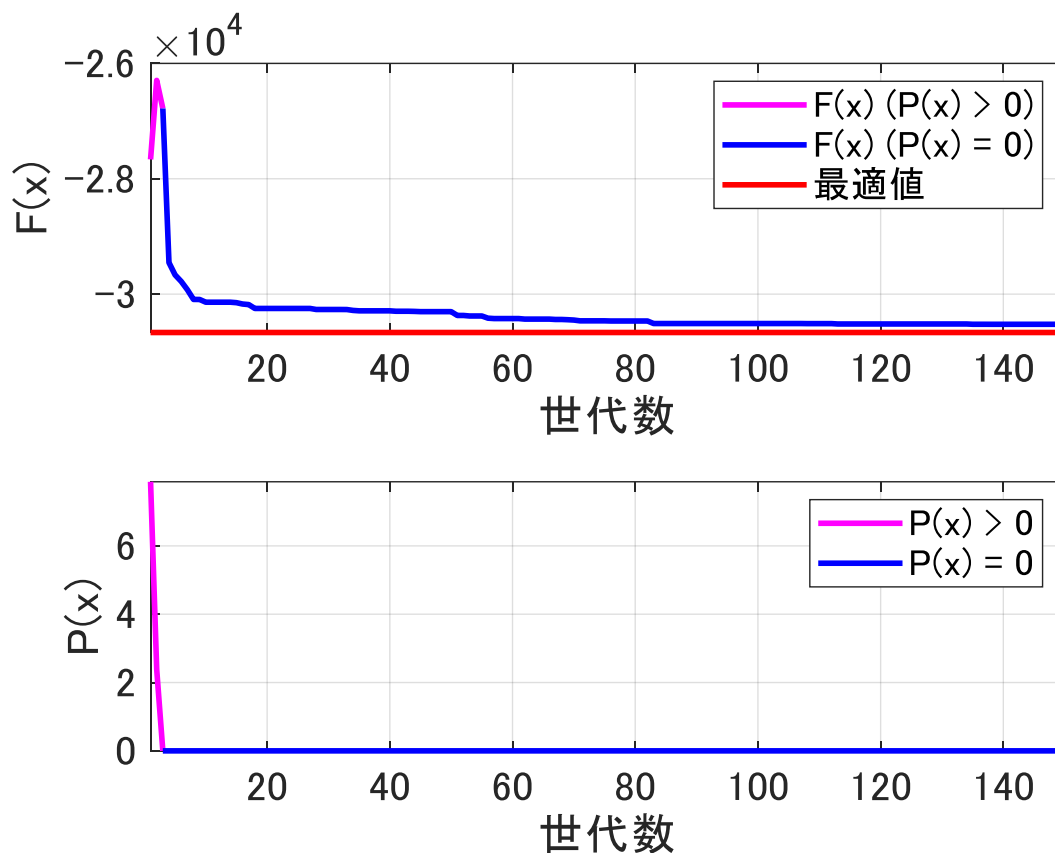


図 3.11 : go04 への適用結果

上図は、PIP 法を用いて go04 を解いた計算結果を表している。このグラフについても解探索が開始されてから早期に  $P(x) = 0$  となる実行可能解の生成に成功していることがわかる。一方、 $F(x)$  のグラフについては、go04 が最小化問題であるにもかかわらず一旦評価値が増加した直後に減少傾向に転じている様子を確認できる。この理由も go01 や go03 と同様に実行可能解が生成されるまでの間は  $F(x)$  の評価値の変化を度外視して実行可能解の探索のみを優先した結果だと考えられる。 $P(x)$  の青色で示された部分のグラフは、その後  $P(x) = 0$  の状態を維持したまま推移している一方、 $F(x)$  のグラフについては、青色に変化してからは一貫してその値が減少していき、赤色の直線で示された最適値にかなり近づいている様子を確認できる。表 3.1 に示された最適解と生成解の誤差率は 0.60% であることと、上記で示した解探索の経過から、この問題についても PIP 法が想定されたとおりの解探索が実現されているとみなしてよいといえる。

## (5) go05~go10

図 3.12~図 3.17 に示す 6 つのグラフは、それぞれ PIP 法を用いて go05~go10 を解いた計算結果を表している。

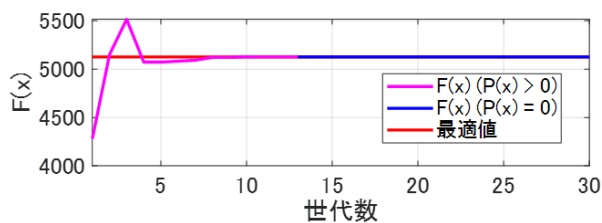


図 3.12 : go05 への適用結果

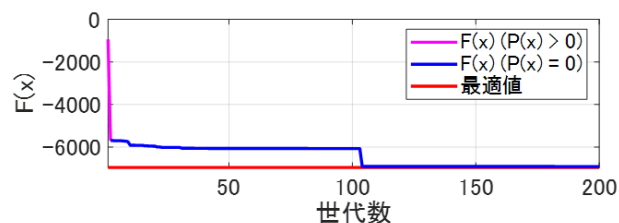


図 3.13 : go06 への適用結果

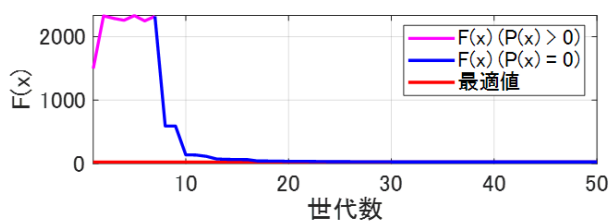


図 3.14 : go07 への適用結果

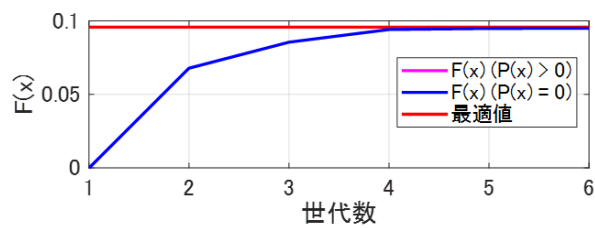


図 3.15 : go08 への適用結果

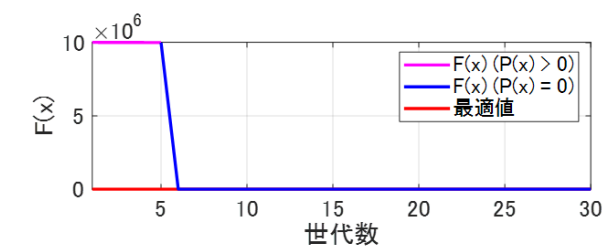


図 3.16 : go09 への適用結果

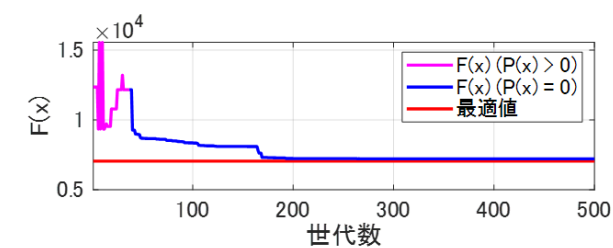


図 3.17 : go10 への適用結果

上に示した6つの計算結果より, go01~go04の場合と同様にその全てにおいて実行可能解の生成に成功している様子がわかる. 実行可能解が生成されるまでの間に $F(\mathbf{x})$ の値が上下する理由についても, これまでと同様に PIP 法の第1段階における解探索が $F(\mathbf{x})$ の評価値の変化を度外視して実行可能解の探索のみを優先した結果だと考えられる. また, 青色で示された $P(\mathbf{x})$ のグラフが $P(\mathbf{x}) = 0$ の状態を維持している状態で $F(\mathbf{x})$ のグラフが赤色の直線で示された最適値に漸近していく様子が確認できることから, 実行可能解の単目的最適化についても想定通りに機能しているといえる(ただし, go05 に関しては実行可能解が生成された段階で既に最適値が得られている). これは, 表 3.1 に示された最適解と生成解の誤差率からも確認することが可能である. なお, go02 及び go07 の誤差率は他と比べると比較的大きいことがわかる. これは, PIP 法の解探索プロセスというよりはむしろ GA の根幹である交叉・交配及び突然変異演算子の欠陥によるものか計算世代数が十分でなかったことに起因していると推測する. なぜなら, いずれの解探索過程も途中までは想定通りに機能していることが確認できるためである.

### 3.5 DTLZ 問題を用いた PIP 法の妥当性評価

「DTLZ」という用語は, このベンチマーク問題を考案した4人の執筆者の名前の頭文字を繋ぎ合わせたことに由来している. DTLZ 問題の特徴は, 目的関数の個数と設計変数の個数をユーザーが自由に設定できることであり, パレートフロンティアの形状も既知であるため, 生成した解がその近傍に存在するかどうか視覚的に判別することが可能である. そのため, これまで提案されてきた数多くの最適化手法の妥当性を評価するベンチマーク問題として頻繁に用いられてきた. 以下に DTLZ 問題に対して PIP 法を適用した結果を示す. なお, 使用する各 DTLZ 問題の定義式については Appendix B に示す.

#### (1) DTLZ1

前述したように, DTLZ 問題には制約条件が与えられていないため, 独自に与える必要がある. ここでは目的関数の数を $M = 2$ , 設計変数の数を $n = 10$ に設定し, 式(3.7)に示す制約条件を DTLZ1 に与える.

$$\begin{cases} 0 \leq f_1(\mathbf{x}) \leq 0.3 \\ 0 \leq f_2(\mathbf{x}) \leq 1.5 \end{cases} \quad (3.7)$$

以上の条件下で DTLZ1 に PIP 法を適用した場合の計算結果を図 3.18(a)~(d)にそれぞれ示す. なお, DTLZ 問題も GO 問題と同様全て単位を持たない実数値であるため, 解表現や遺伝的操作に関しても同様の方法を踏襲する. また, GA のパラメータとして, 解集合の個体数を $P_n = 200$ , エリート数を $E_n = 10$ , 最大世代数を $G_{max} = 1000$ に設定し, 最大世代数に到達した時点で解探索を終了とした.

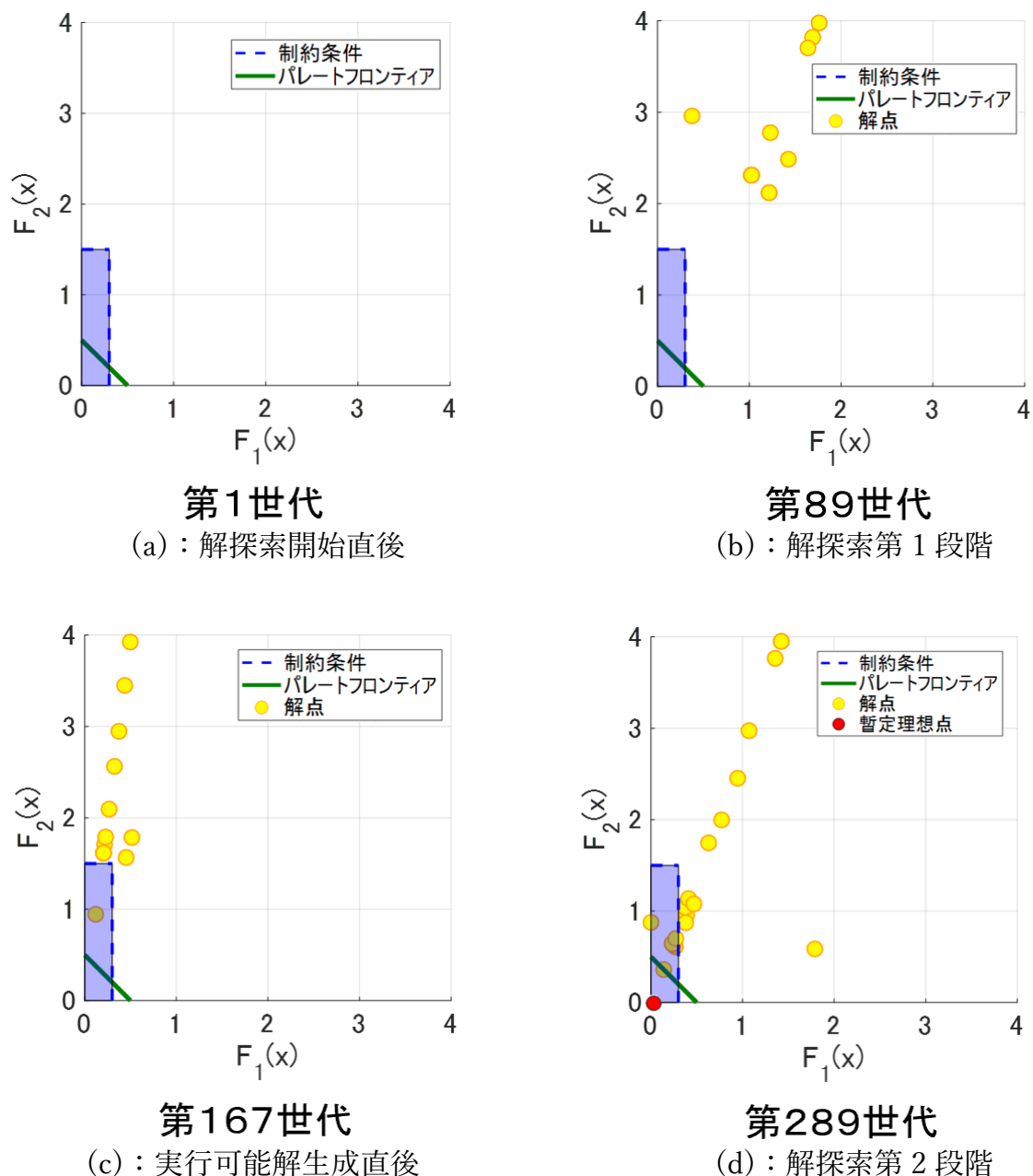


図 3.18 : DTLZ1 の解探索結果

図 3.18(a)は, DTLZ1 のパレートフロンティア近傍の解空間を示している. 原点近傍に描かれた緑色の直線は DTLZ1 のパレートフロンティアであり, 青色の領域は式(3.7)で示した制約条件を視覚化したエリアである. 図 3.18(b)は, PIP 法における解探索第1段階の様子の一部を表しており, 第 89 世代における原点付近の解空間の様子を表している. この様に, 世代数が進むにつれて青色のエリア付近に生成される黄色の丸印で示された解点の数が増加していく. その後, 図 3.18(c)に示すように第 167 世代で青色のエリア内に解点が生じることがわかる. この時点で与えられた制約条件を全て満たす実行可能解が生成されたと判断することができ, 解探索は第 2 段階へ移行する. また, この際赤色の丸印で示さ

れた暫定理想点も同時に生成される．第 2 段階では生成した暫定理想点と解点のユークリッド距離が最小となるように解探索を実行する．図 3.18(d)は，第 289 世代においてパレートフロンティア上に解点を生成できた様子を表している．この計算結果から，PIP 法によって所望の選好解を得ることに成功したといえる．

## (2) DTLZ3

図 3.19(a)～(d)に DTLZ3 に PIP 法を適用した計算結果を示す．計算条件は，制約条件も含め DTLZ1 を解いた時と同様の設定とした．

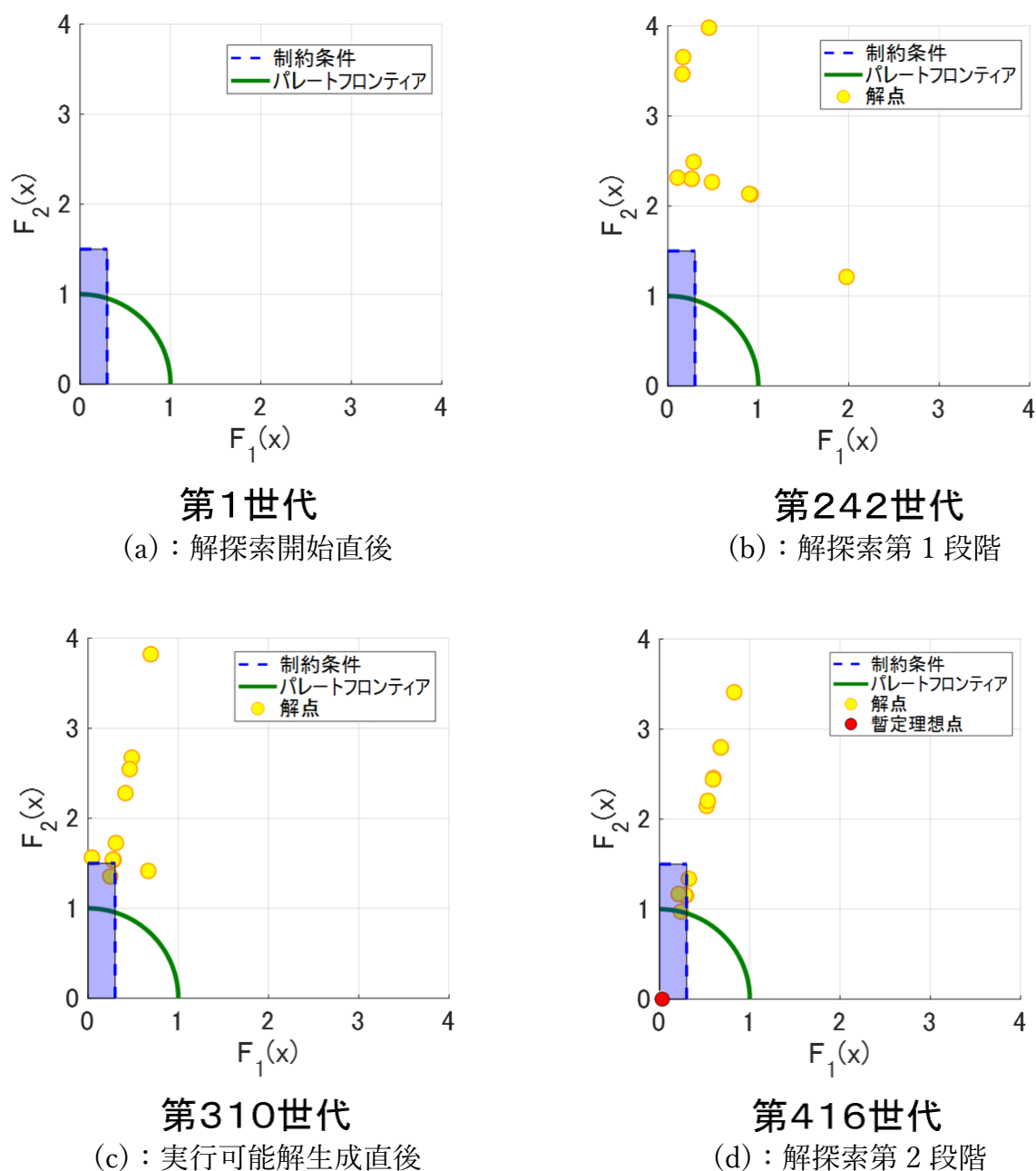


図 3.19 : DTLZ3 の解探索結果

DTLZ3 は、上図に示したように原点に対して非凸となるパレートフロンティアを持つため、従来の多目的最適化手法では意思決定者の意図通りに探索が行われないケースが報告されている<sup>[92]</sup>。このため、PIP 法がパレートフロンティア形状にかかわらず安定した解探索が実行可能かどうか評価する上で最適なベンチマーク問題とすることができる。

図 3.19(a)は、DTLZ3 のパレートフロンティア近傍の解空間を示している。図 3.19(b)は、PIP 法の解探索第 1 段階において世代数が 240 を超えたあたりで解点が徐々に制約条件を示す青色の領域付近に生成され始めたことを示している。その後、図 3.19(c)に示すように、第 310 世代で青色の領域内に解点が生成されている。つまり、この時点で実行可能解の生成に成功したと判断できる。その後第 2 段階の解探索へ移行し、最終的に図 3.19(d)に示すように第 416 世代でパレートフロンティア上に解点を生成していることがわかる。この計算結果より、DTLZ3 においても PIP 法を用いることで所望の選好解を得ることができるといえる。

### (3) DTLZ7

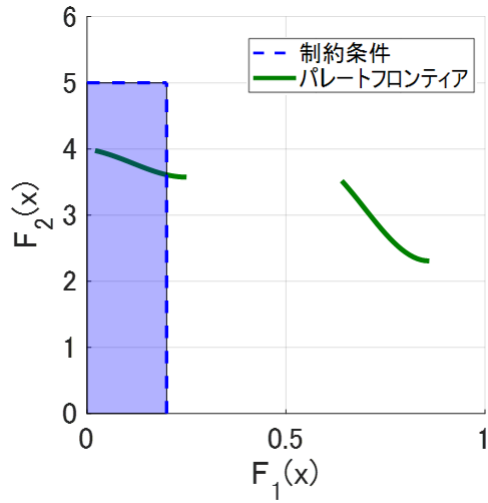
DTLZ7 のパレートフロンティアは図 3.20(a)に示すように不連続形状となるのが特徴である。また、Appendix B に示した定義式から分かるように、前述した 2 つの DTLZ 問題とは異なり、 $M$  番目の目的関数以外は該当する設計変数の定義域がそのまま目的関数の値域となっていることも大きな特徴である。DTLZ7 に対し式(3.8)で示すように独自の制約条件を付与することによって PIP 法の妥当性を評価する。

$$\begin{cases} 0 \leq f_1(\mathbf{x}) \leq 0.2 \\ 0 \leq f_2(\mathbf{x}) \leq 5.0 \end{cases} \quad (3.8)$$

上記で示した制約条件以外の計算条件は、DTLZ1 及び DTLZ3 と同様とする。

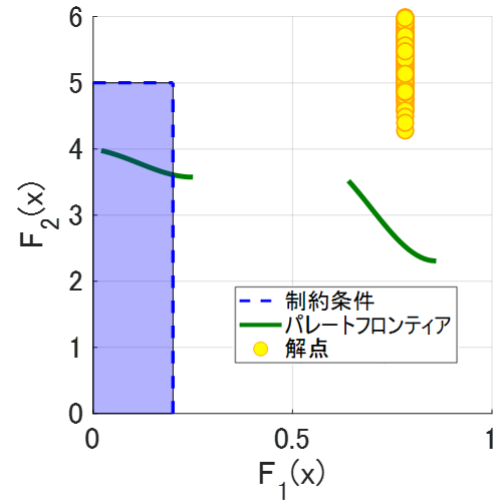
図 3.20(b)は、PIP 法の第 1 段階を適用して 101 世代が経過したときのパレートフロンティア近傍の様子を表している。DTLZ1 や DTLZ3 とは異なり、解点が縦方向に密集しているのが特徴的である。これは、DTLZ7 の定義式の特徴に起因していると考えられる。図 3.20(c)では、その後まもなく実行可能解が得られる様子が確認できる。そして最終的に図 3.20(d)で示すように第 135 世代において制約条件内のパレートフロンティア上に選好解を生成できている様子が確認できる。

以上の結果から、DTLZ7 においても PIP 法を用いて所望の選好解を得られることが示された。



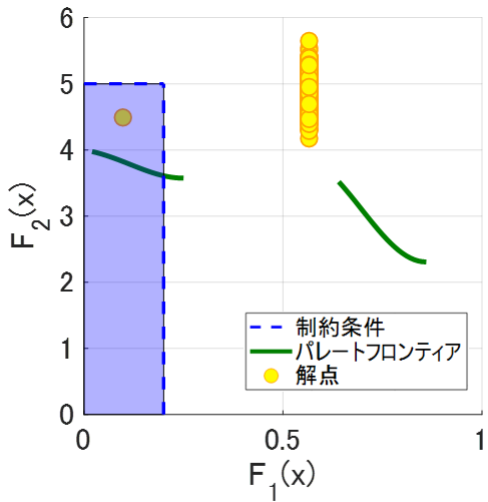
第1世代

(a) : 解探索開始直後



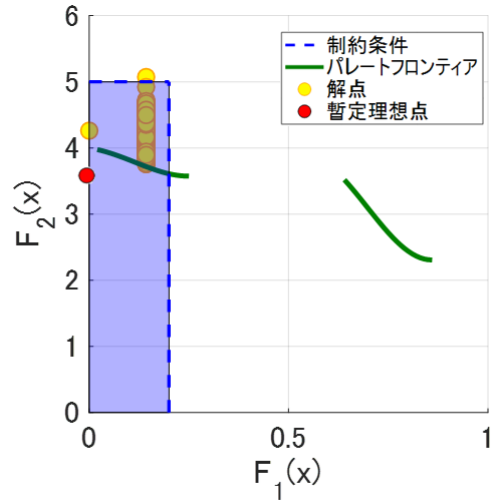
第101世代

(b) : 解探索第1段階



第118世代

(c) : 実行可能解生成直後



第135世代

(d) : 解探索第2段階

図 3.20 : DTLZ7 の解探索結果

## 第4章 宅配経路問題の定式化

### 4.1 はじめに

この章では、ドローン宅配サービスの経路生成問題(以下、「DDP(Drone Delivery Problem)」と呼称する)を制約付き多目的最適化問題として定式化することについて説明する。はじめに、いくつかの宅配スタイルに対して固有の名称を与える。

まず、トラックのみを用いた宅配スタイルについて、本論文では「Truck 宅配」と呼ぶものとする。その概念図を図4.1に示す。

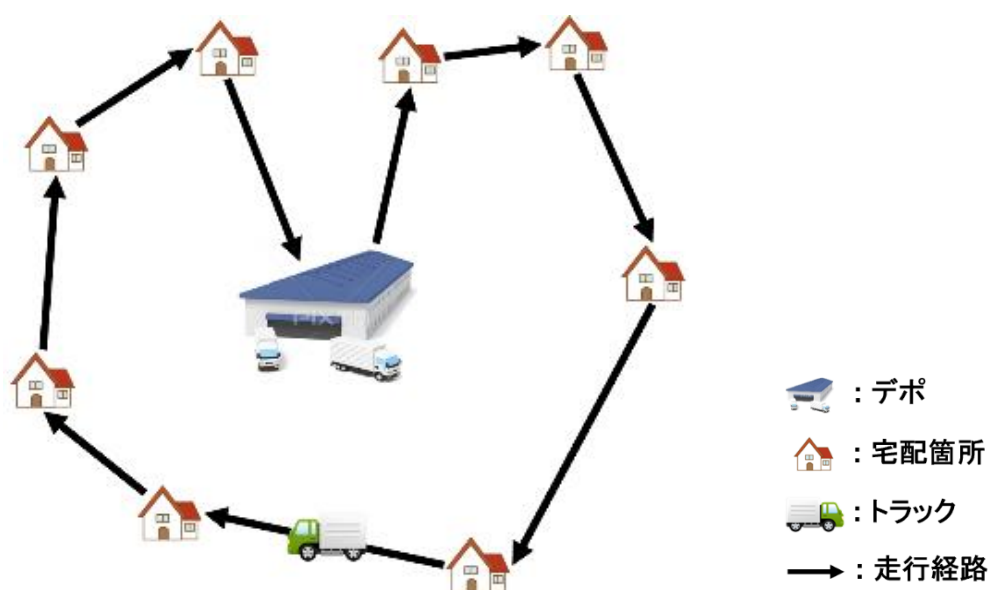


図4.1：Truck 宅配の概念図

上図が示すように、この宅配サービスは宅配ドライバーが宅配場所の玄関先まで直接赴き、住人に荷物を届けるという伝統的なスタイルであり、現在でも主流のサービスとなっている。

これに対し、図4.2は、ドローンのみを用いてデポから直接各家庭へ荷物を運ぶスタイルを表している。この宅配スタイルについて、本論文では「Drone 宅配」と呼ぶものとする。この宅配スタイルは、全ての宅配作業をドローンに担わせることを前提としているので、ここ

では宅配ドライバー等に支払う人件費が一切掛からないと仮定する(本来であれば, ドローンの運用に必要な人的コストを考慮すべきであるが, 具体的な指標となる資料が存在しないため, 宅配コストに包含しない). 一方, デポから遠方に位置する宅配場所については, ドローンの航続距離が不足する場合も想定される. その際には何らかの形で別途配送作業を補完しない限りこの宅配スタイルの経路生成は不可能となることに留意しなければならない.

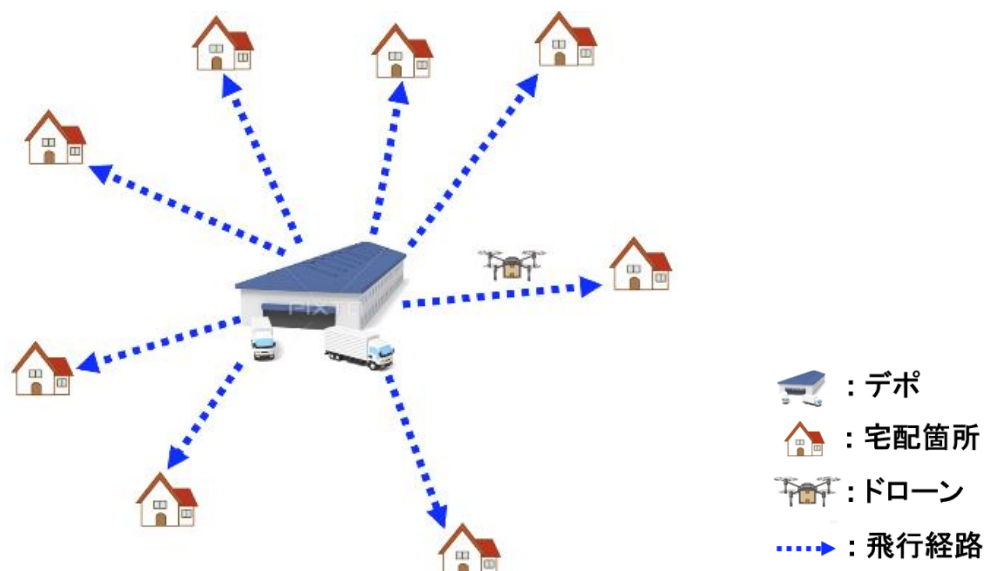


図 4.2 : Drone 宅配の概念図

図 4.3 は, トラック上にドローンを搭載し, 異なる宅配場所に同時並行で宅配サービスを展開することが可能なハイブリッド型の宅配スタイルを表しており, 本論文ではこれを「Hybrid 宅配」と呼ぶ.

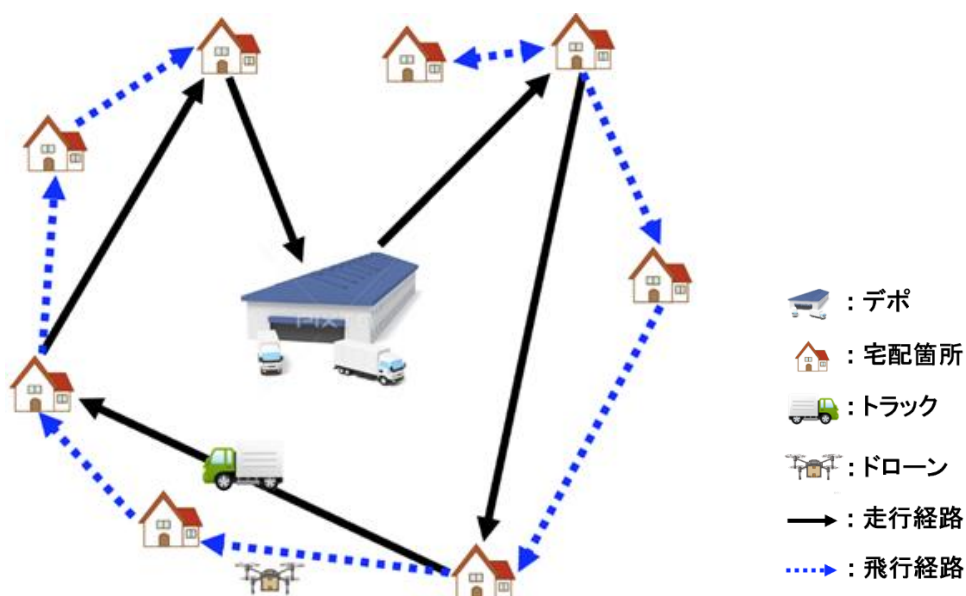


図 4.3 : Hybrid 宅配の概念図

この宅配スタイルは、1.3 節の図 1.8 で示した Tandem Delivery と同様のものであり、トラックとドローンが同時に別々の場所へ宅配することによって作業時間を短縮することが可能である。

この Hybrid 宅配を発展させたものとして、図 4.4 に示すような 1 台のトラックに複数のドローンを搭載して同時に運用する宅配スタイルも考えられる。本論文ではこれを「Multiple」の頭文字をとって「M-Hybrid 宅配」と呼ぶ。

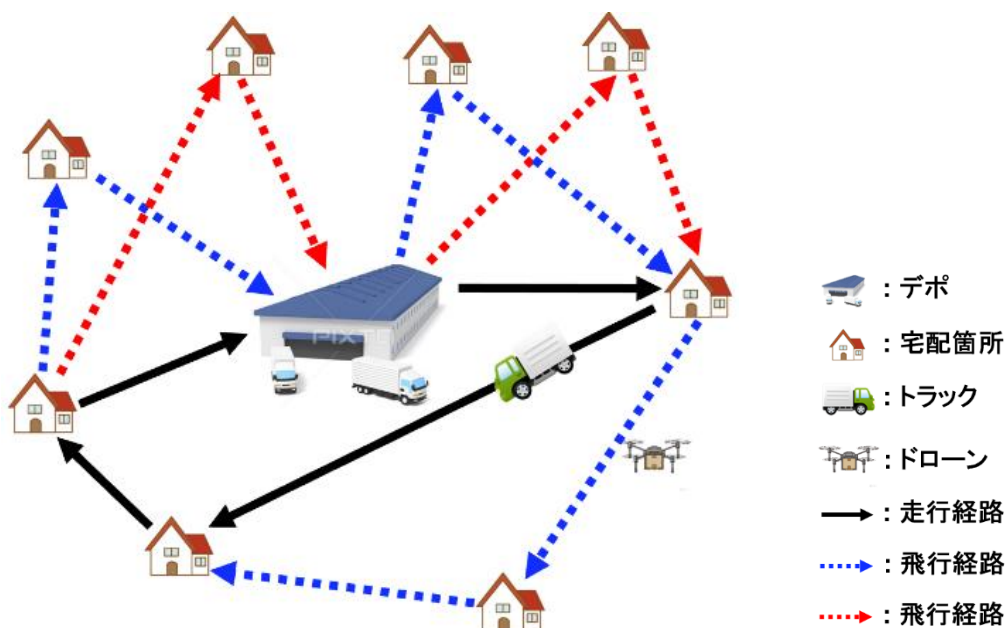


図 4.4 : M-Hybrid 宅配の概念図

さらに、図 4.5 に示すように、複数の宅配ドライバーに宅配業務を分散する宅配スタイルも考えられる。本論文では、これを「Multi-Agent 宅配」と呼ぶものとする。この宅配スタイルでは、Truck 宅配, Drone 宅配及び Hybrid 宅配が混在した経路を生成することが可能である。また、この種の宅配経路に関する問題は、その特性上経路の生成と宅配業務(タスク)の割り当てを同時に最適化する必要がある。

以下では、図 4.1～図 4.5 で示した各宅配スタイルにおける経路を表現する解表現方法とその遺伝的操作方法について示す。また、本論文における各宅配スタイルに関する宅配経路問題の目的関数及び制約条件を制約付き多目的最適化問題として定式化する。

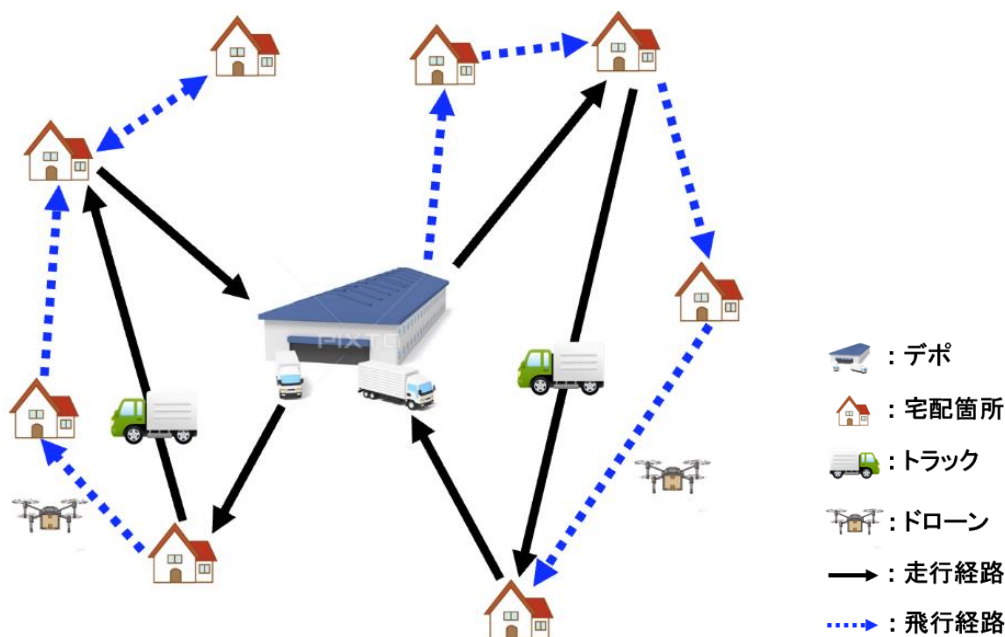


図 4.5 : Multi-Agent 宅配の概念図

## 4.2 解表現と遺伝的操作

### 4.2.1 Truck 宅配

式(4.1)は, Truck 宅配経路の解構造である  $T^i$  の定義式である.

$$T^i = [L_d, T_1^i, T_2^i, \dots, T_a^i, \dots, T_{NT_i}^i, L_d] \quad (4.1)$$

$(i = 1, 2, \dots, A \quad a = 1, 2, \dots, NT_i)$

ここで,  $L_d$  はデポ座標,  $T_a^i$  はトラックが宅配する WP 座標,  $A$  はトラックの保有台数,  $NT_i$  は  $i$  番目のトラックが宅配する WP の数を表す. 図 4.6 は,  $T^i$  の概念図を表す.

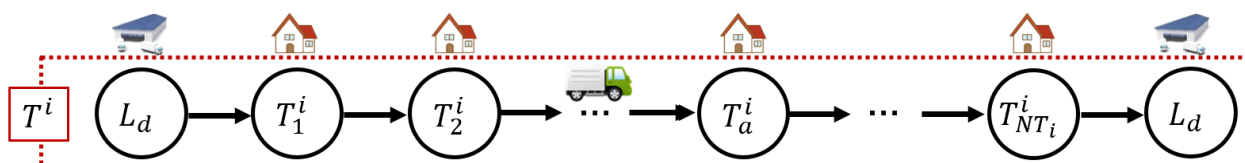


図 4.6 :  $T^i$  の概念図

上式及び上図に示すように,  $T^i$  はデポを出発点とした Truck 宅配経路を表しており, トラックによって配達される全ての WP が配達順に左から並べられている.  $T_a^i$  には  $N_{WP}$  個存在する WP のうちの 1 つが割り当てられる. この解構造に対する遺伝的操作方法を以下に 3 つ挙げる. なお, 文中の  $a1, a2, l1, l2$  は, 式(4.2)及び式(4.3)に示す定義域における任意の整数とする.

$$1 \leq a1 \leq a2 \leq A \quad (4.2)$$

$$1 \leq l1 \leq l2 \leq NT_i \quad (4.3)$$



$$T^i = [L_d, T_1^i, T_2^i, \dots, T_a^i, \dots, T_{NH_i}^i, L_d] \quad (4.7)$$

$$(a = 1, 2, \dots, NH_i)$$

$$D^{ij} = [D_{L_d}^{ij}, D_1^{ij}, D_2^{ij}, \dots, D_a^{ij}, \dots, D_{NH_i}^{ij}] \quad (4.8)$$

$$u^{ij} = [u_{L_d}^{ij}, u_1^{ij}, u_2^{ij}, \dots, u_a^{ij}, \dots, u_{NH_i}^{ij}] \quad (4.9)$$

$$u_a^{ij} \in \{0, 1\}$$

ここで、 $NH_i$ は $i$ 番目のトラックが宅配する $WP$ の総数を表す。図 4.8 は、 $T^i, D^{ij}$ 及び $u^{ij}$ の概念図を表す。

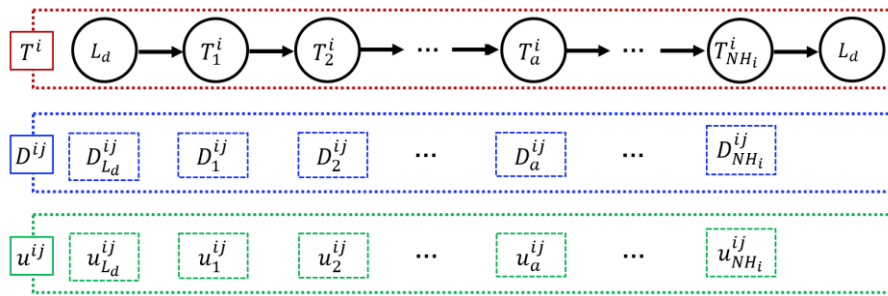


図 4.8 :  $T^i, D^{ij}$ 及び $u^{ij}$ の概念図

$D_a^{ij}$ は、式(4.10)に示すように $T_a^i$ を離陸したドローンが宅配する $WP$ が宅配順に並べられている。なお、 $h_{a,ij}$ は $T_a^i$ を離陸したドローンが宅配する $WP$ の総数を表す。

$$D_a^{ij} = [D_{a,1}^{ij}, D_{a,2}^{ij}, \dots, D_{a,b}^{ij}, \dots, D_{a,h_{a,ij}}^{ij}] \quad (4.10)$$

$$(b = 1, 2, \dots, h_{a,ij} )$$

$u_a^{ij}$ は $D_{a,h_{a,ij}}^{ij}$ での宅配を完了したドローンの行先を決定する。 $u_a^{ij} = 0$ であれば $T_a^i$ に帰還させ、 $u_a^{ij} = 1$ であれば $T_{a+1}^i$ に向けて飛行させる。この方式に従って $T_a^i - T_{a+1}^i$ 間におけるドローンの飛行経路の一例を表したものが図 4.9 である。

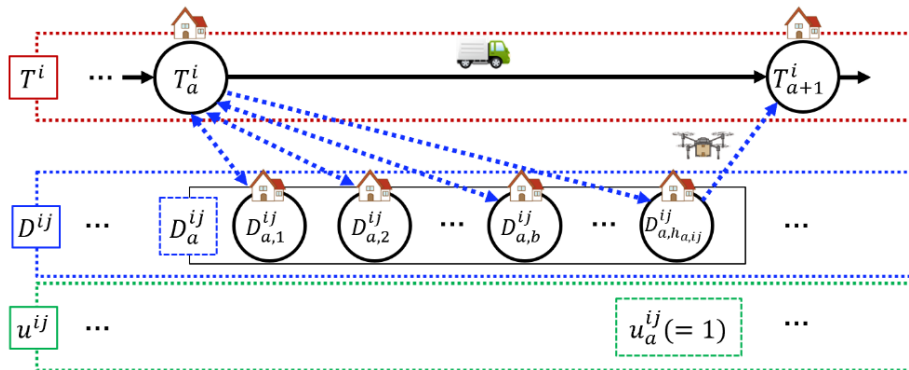


図 4.9 :  $D_a^{ij}$ の飛行経路の一例

$T^i, D^{ij}$  及び  $u^{ij}$  に対する遺伝的操作方法は、操作①～⑤の5種類に加え、以下に示す3種類を用いる。なお、文中の  $c1, n1, n2$  は、式(4.11)及び式(4.12)に示した定義域の任意の整数とする。

$$1 \leq c1 \leq h_{a,ij} \quad (4.11)$$

$$1 \leq n1 \leq n2 \leq NH_i \quad (4.12)$$

操作⑥：  $T^{a1}$  より  $T_{m1}^{a1}, D^{a2b1}$  より  $D_{m2}^{a2b1}$  を任意に選択し、 $T_{m1}^{a1}$  を  $D_{m2}^{a2b1}$  の任意の場所に挿入する。

操作⑦：  $D_{m1}^{a1b1}$  より  $D_{m1,n1}^{a1b1}$  を任意に選択し、 $T^{a2}$  の任意の WP 間に挿入する。

操作⑧：  $u^{a1b1}$  より  $u_{m1}^{a1b1}$  を任意に選択し、 $u_{m1}^{a1b1} = 1$  ならば  $u_{m1}^{a1b1} = 0, u_{m1}^{a1b1} = 0$  ならば  $u_{m1}^{a1b1} = 1$  とする。

### 4.3 問題設定

#### 4.3.1 地図データの作成

本研究では、より現実に近い環境下におけるドローン宅配のコスト削減効果を評価するため、図 4.10 に示すように Google Map より九州大学伊都キャンパスが所在する福岡市西区から糸島市内の一部地域の地図を引用し、このマップ上に示した場所を宅配経路生成の対象とするエリアとして設定する<sup>[98]</sup>。まず、この地図に画像処理を施して道路や山林、海などの形状を抽出することで、宅配エリアの道路形状や障害物、飛行可能エリア等を取得する。続いて、抽出した道路上の任意の場所にデポを1か所、WPを30か所配置する。同図に黄色の印とマゼンタ色の印でその位置を示す。

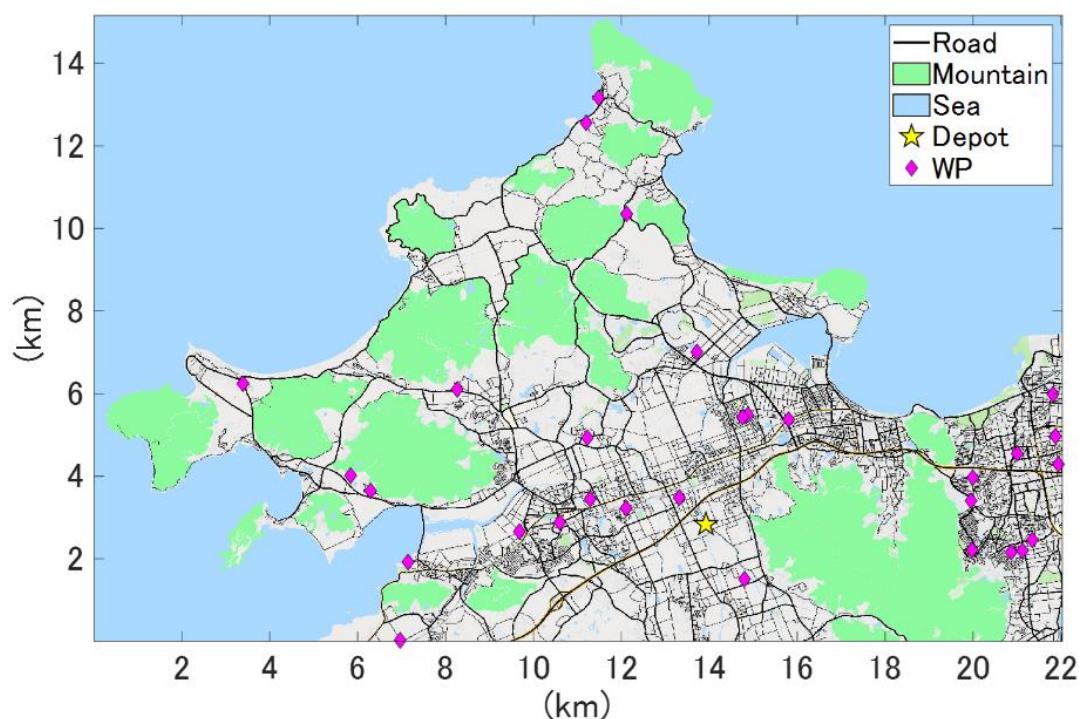


図 4.10：宅配エリア

この地図において、トラックは道路上のみ、ドローンは山林以外のエリアを自由に移動できると仮定する。

#### 4.3.2 事前計算

GA を用いた経路生成では、新しい経路が生成される度にその経路全体の距離コストを評価する必要があるため、その解探索には多くの計算資源を割かなければならない。一方で、評価計算を行う上で同様の経路の距離コストを算出することは頻繁に発生する。このように 1 度算出した実績のある経路を再び評価することは計算資源の無駄になるためできる限り避けることが望ましい。そこで、各WP間の最短経路とその距離コストを事前にデータベース化し、生成された経路全体の距離コストを算出する際にこのデータベースを参照することにより、経路の重複計算を回避し、計算速度を向上させることができる。この手法は満武氏らが提案した A\*-EC algorithm をベースにしている<sup>[99]</sup>。なお、DDP においてはトラックとドローンの両方を使用した宅配経路を生成することを想定するため、データベース化するWP間の最短経路は、トラックの陸路とドローンの空路の両方が必要である。以下にその詳細について示す。

##### (1) 陸路

道路形状を表現できるように図 4.10 の道路上に「通行可能ポイント」と呼称するポイントを複数設置する。これらのポイントは、互いに連結させることによって道路形状を表現できる程度の間隔で配置する。配置した通行可能ポイント間の障害物の有無はレイ・トレーシング法を用いて判断できる<sup>[100]</sup>。WP間の最短距離コストは、連結可能な通行可能ポイント間の距離コストに関する情報を元に A\*アルゴリズムを適用することによって算出できる。式 (4.13) は、算出したWP間の最短距離コストを表す。

$$TC_{d_1 d_2} = TC_{d_2 d_1} \quad (4.13)$$

$$(0 \leq d_1, d_2 \leq N_{WP})$$

ここで、 $d_1$  及び  $d_2$  は各 WP に割り振られた固有の番号を表し、 $0$  はデポ、 $1$  から  $N_{WP}$  は各 WP の固有番号である。算出した全ての WP 間の距離コストを格納したデータベースは、式 (4.14) に示すように表現できる。

$$TC = \begin{bmatrix} 0 & TC_{01} & \dots & TC_{0N_{WP}} \\ TC_{10} & 0 & \dots & TC_{1N_{WP}} \\ \vdots & \vdots & \ddots & \vdots \\ TC_{N_{WP}0} & TC_{N_{WP}1} & \dots & 0 \end{bmatrix} \quad (4.14)$$

##### (2) 空路

図 4.10 で示した山以外のエリアに縦横 4m 間隔でドローンが経路可能な点を配置する。これを「飛行可能ポイント」と呼称する。配置した飛行可能ポイント間における障害物の有

無とポイント間の距離コスト及び WP 間の飛行距離コストを陸路の場合と同様に算出する。式(4.15)は算出した飛行距離コストを表す。

$$DC_{d1d2} = DC_{d2d1} = \|D_{d1}^i - D_{d2}^i\| = \|D_{a,d1}^{ij} - D_{a,d2}^{ij}\| \quad (4.15)$$

$$0 \leq d1, d2 \leq N_{WP}$$

ここで、 $d1$ 及び $d2$ は各 WP を判別するための固有の番号である。前節と同様に番号を割り振ると、算出した全ての飛行距離コストは式(4.16)のように表現できる。

$$DC = \begin{bmatrix} 0 & DC_{01} & \cdots & DC_{0N_{WP}} \\ DC_{10} & 0 & \cdots & DC_{1N_{WP}} \\ \vdots & \vdots & \ddots & \vdots \\ DC_{N_{WP}0} & DC_{N_{WP}1} & \cdots & 0 \end{bmatrix} \quad (4.16)$$

#### 4.4 定式化

Truck 宅配, Drone 宅配及び Hybrid 宅配の経路生成問題を制約付き多目的最適化問題としてそれぞれ以下に示すように定式化する。

##### 4.4.1 Truck 宅配

Truck 宅配経路問題を式(4.17)に示すように定式化する。なお、使用できるトラックの台数を $A$ 、走行速度を $V_T$ 、ドライバーがWPに到着してから玄関先に荷物を届け終わるまでの時間を $t_p$ 、宅配ドライバーがWPに到着してからトラックを停車させるまでの時間を $t_{sp}$ 、エンジンを始動してから再発進するまでの時間を $t_{st}$ 、 $LD_{max}$ はトラックの走行可能距離、 $LT_{max}$ はドライバーの宅配可能時間を表す。

$$\left\{ \begin{array}{l} \min_{\mathbf{x}} \mathbf{F}(\mathbf{x}) = \min_{\mathbf{x}} [F_1(\mathbf{x}), F_2(\mathbf{x}), F_3(\mathbf{x})] \\ F_1(\mathbf{x}) = \sum_{i=1}^A \sum_{a=0}^{NT_i} TC_{T_a^i T_{a+1}^i} \quad (T_0^i = T_{NT_i+1}^i = L_d = 0) \\ F_2(\mathbf{x}) = \sum_{i=1}^A \left( \sum_{a=0}^{NT_i} \frac{TC_{T_a^i T_{a+1}^i}}{V_T} + NT_i(t_p + t_{st} + t_{sp}) \right) \\ F_3(\mathbf{x}) = \sum_{i=1}^A T_{i,Count} \quad T_{i,Count} = \begin{cases} 1, & NT_i > 0 \\ 0, & NT_i = 0 \end{cases} \\ \text{subject to} \\ g_1(\mathbf{x}) = F_1(\mathbf{x}) \leq LD_{max} \\ g_2(\mathbf{x}) = F_2(\mathbf{x}) \leq LT_{max} \\ g_3(\mathbf{x}) = F_3(\mathbf{x}) = A \end{array} \right. \quad (4.17)$$

ここで、目的関数は 3 つであり、 $F_1(\mathbf{x})$ はトラックの合計走行距離、 $F_2(\mathbf{x})$ は全ての宅配が完

了するまでに要した宅配時間,  $F_3(\mathbf{x})$ は使用するトラックの台数を表す. また,  $g_1(\mathbf{x}) \sim g_3(\mathbf{x})$ はそれぞれ $F_1(\mathbf{x}) \sim F_3(\mathbf{x})$ の制約条件を表す.

#### 4.4.2 Drone 宅配

Drone 宅配経路問題を式(4.18)に示すように定式化する. なお, 使用できるドローンの数を $B$ , 回収したドローンのバッテリーを新しいものと交換するために必要な時間を $t_{re}$ , ドローンがトラックを垂直離陸してから安全な高度に達するまでの時間と, 目的地上空から地上へ着陸するまでに要する時間を $t_{tl}$ , ドローンの飛行速度を $V_D$ , 飛行可能時間を $t_d$ , ドローンの積載可能重量を $W_d$ ,  $w_b^j$ は宅配先 $WP$ の荷物の重量を表す.

$$\left\{ \begin{array}{l} \min_{\mathbf{x}} \mathbf{F}(\mathbf{x}) = \min_{\mathbf{x}} [F_4(\mathbf{x}), F_5(\mathbf{x}), F_6(\mathbf{x})] \\ F_4(\mathbf{x}) = 2 \sum_{j=1}^B \sum_{b=1}^{ND_i} DC_{L_d} D_b^j \\ F_5(\mathbf{x}) = \sum_{j=1}^B \left( \sum_{b=1}^{ND_i} \frac{2DC_{L_d} D_b^j}{V_D} + ND_i \times (t_{re} + 4t_{tl}) \right) \\ F_6(\mathbf{x}) = \sum_{j=1}^B D_{j,Count} \quad D_{j,Count} = \begin{cases} 1, & ND_i > 0 \\ 0, & ND_i = 0 \end{cases} \\ \text{subject to} \\ g_1(\mathbf{x}) = F_6(\mathbf{x}) \leq B \\ g_2(\mathbf{x}) = \frac{2DC_{L_d} D_b^j}{V_D} + 4t_{tl} \leq t_d \\ g_3(\mathbf{x}) = w_b^j \leq W_d \end{array} \right. \quad (4.18)$$

ここで, 目的関数は 3 つであり,  $F_4(\mathbf{x})$ は合計飛行距離,  $F_5(\mathbf{x})$ はドローンの合計飛行時間,  $F_6(\mathbf{x})$ は使用するドローンの使用台数を表す. また,  $g_1(\mathbf{x})$ はドローンの使用台数,  $g_2(\mathbf{x})$ はドローンの 1 回の使用における飛行可能時間,  $g_3(\mathbf{x})$ はドローンの積載可能重量に関する制約条件である.

#### 4.4.3 Hybrid/M-Hybrid/Multi-Agent 宅配

Hybrid/M-Hybrid/Multi-Agent 宅配経路問題を式(4.19)に示すように定式化する. なお, 1 台のトラックに搭載できるドローンの数を $B$ とし, それ以外の記号で式(4.17)及び式(4.18)と重複する記号については同一の意味を持つものとする. また,  $w_{a,b}^{ij}$ は宅配先 $WP$ の荷物の重量を表す.

$$\begin{cases}
\min_{\mathbf{x}} F(\mathbf{x}) = \min_{\mathbf{x}} [F_1(\mathbf{x}), F_2(\mathbf{x}), F_3(\mathbf{x}), F_4(\mathbf{x}), F_5(\mathbf{x}), F_6(\mathbf{x})] \\
F_1(\mathbf{x}) = \sum_{i=1}^A \sum_{a=0}^{NH_i} TC_{T_a^i T_{a+1}^i} \quad (T_0^i = T_{NH_i+1}^i = L_d = 0) \\
F_2(\mathbf{x}) = \sum_{i=1}^A \sum_{a=0}^{NH_i} \left( f_{1,ai}(\mathbf{x}) + f_{3,ai}(\mathbf{x}) + t_{re} \sum_{j=1}^B u_a^{ij} \right) \\
F_3(\mathbf{x}) = \sum_{i=1}^A T_{i,Count} \quad T_{i,Count} = \begin{cases} 1, & NH_i > 0 \\ 0, & NH_i = 0 \end{cases} \\
F_4(\mathbf{x}) = \sum_{i=1}^A \sum_{j=1}^B \sum_{a=0}^{NH_i} (f_{1,aij}(\mathbf{x}) + f_{2,aij}(\mathbf{x}) + f_{3,aij}(\mathbf{x})) \\
F_5(\mathbf{x}) = \sum_{i=1}^A \sum_{j=1}^B \sum_{a=0}^{NH_i} (f_{4,aij}(\mathbf{x}) + f_{5,aij}(\mathbf{x}) + f_{6,aij}(\mathbf{x})) \\
F_6(\mathbf{x}) = \sum_{i=1}^A \sum_{j=1}^B D_{ij,Count} \quad D_{ij,Count} = \begin{cases} 1, & \sum_{a=0}^{NH_i} h_{a,ij} > 0 \\ 0, & \sum_{a=0}^{NH_i} h_{a,ij} = 0 \end{cases} \\
\text{subject to} \\
g_1(\mathbf{x}) = F_1(\mathbf{x}) \leq LD_{max} \\
g_2(\mathbf{x}) = F_2(\mathbf{x}) \leq LT_{max} \\
g_3(\mathbf{x}) = F_3(\mathbf{x}) \leq A \\
g_4(\mathbf{x}) = f_{4,aij}(\mathbf{x}) \leq t_d \\
g_5(\mathbf{x}) = f_{5,aij}(\mathbf{x}) \leq t_d \\
g_6(\mathbf{x}) = f_{6,aij}(\mathbf{x}) \leq t_d \\
g_7(\mathbf{x}) = F_6(\mathbf{x}) \leq B \\
g_8(\mathbf{x}) = w_{a,b}^{ij} \leq W_d
\end{cases} \tag{4.19}$$

ここで、式(4.19)内で用いた各関数の定義式を式(4.20)～式(4.29)に示す。

$$f_{1,aij}(\mathbf{x}) = 2 \sum_{b=1}^{h_{a,ij}-1} DC_{T_a^i D_{a,b}^{ij}} \tag{4.20}$$

$$f_{2,aij}(\mathbf{x}) = (1 - u_a^{ij}) \times 2DC_{T_a^i D_{a,h_{a,ij}}^{ij}} \tag{4.21}$$

$$f_{3,aij}(\mathbf{x}) = u_a^{ij} \times \left( DC_{T_a^i D_{a,h_{a,ij}}^{ij}} + DC_{D_{a,h_{a,ij}}^{ij} T_{a+1}^i} \right) \tag{4.22}$$

$$f_{4,aij}(\mathbf{x}) = \frac{f_{1,aij}(\mathbf{x})}{V_D} + 4t_{tl}(h_{a,ij} - 1) \quad (4.23)$$

$$f_{5,aij}(\mathbf{x}) = (1 - u_a^{ij}) \times \left( \frac{f_{2,aij}(\mathbf{x})}{V_D} + 4t_{tl} \right) \quad (4.24)$$

$$f_{6,aij}(\mathbf{x}) = u_a^{ij} \times \left( \frac{f_{3,aij}(\mathbf{x})}{V_D} + 4t_{tl} \right) \quad (4.25)$$

$$f_{7,aij}(\mathbf{x}) = f_{4,aij}(\mathbf{x}) + f_{5,aij}(\mathbf{x}) + t_{re}(h_{a,ij} - u_a^{ij}) \quad (4.26)$$

$$f_{1,ai}(\mathbf{x}) = \max[f_{7,ai1}(\mathbf{x}), f_{7,ai2}(\mathbf{x}), \dots, f_{7,aiB}(\mathbf{x})] \quad (4.27)$$

$$f_{2,ai}(\mathbf{x}) = \frac{TC_{T_a^i T_{a+1}^i}}{V_T} + t_p + t_{st} + t_{sp} \quad (4.28)$$

$$f_{3,ai}(\mathbf{x}) = \max[f_{6,ai1}(\mathbf{x}), f_{6,ai2}(\mathbf{x}), \dots, f_{6,aiB}(\mathbf{x}), f_{2,ai}(\mathbf{x})] \quad (4.29)$$

目的関数は 6 つであり、 $F_1(\mathbf{x})$ はトラックの合計走行距離、 $F_2(\mathbf{x})$ は全ての宅配が完了するまでに要した宅配時間、 $F_3(\mathbf{x})$ は使用するトラックの台数、 $F_4(\mathbf{x})$ は合計飛行距離、 $F_5(\mathbf{x})$ はドローンの合計飛行時間、 $F_6(\mathbf{x})$ は使用するドローンの使用台数を表す。 $g_1(\mathbf{x}) \sim g_3(\mathbf{x})$ については $F_1(\mathbf{x}) \sim F_3(\mathbf{x})$ 、 $g_4(\mathbf{x})$ 及び $g_5(\mathbf{x})$ についてはドローンの飛行可能時間、 $g_6(\mathbf{x})$ についてはドローンの使用可能台数、 $g_7(\mathbf{x})$ についてはドローンの積載可能重量に関する制約条件をそれぞれ意味する。また、式(4.19)内で用いた $f_{1,aij}(\mathbf{x}) \sim f_{7,aij}(\mathbf{x})$ 及び $f_{1,ai}(\mathbf{x})$ と $f_{2,ai}(\mathbf{x})$ の定義式について、 $f_{1,aij}(\mathbf{x})$ は $T_a^i$ を離陸したドローンが $D_{a,1}^{ij}$ から $D_{a,h_{a,ij}-1}^{ij}$ までの宅配を完了させるまでに要した合計飛行距離、 $f_{2,aij}(\mathbf{x})$ は $T_a^i$ を離陸したドローンが $D_{a,h_{a,ij}}^{ij}$ での宅配を完了させ、再び $T_a^i$ へ戻るまでに要する飛行距離、 $f_{3,aij}(\mathbf{x})$ は $T_a^i$ を離陸したドローンが $D_{a,h_{a,ij}}^{ij}$ での宅配を完了させ、その後 $T_{a+1}^i$ でトラックと合流するまでに要する飛行距離、 $f_{4,aij}(\mathbf{x}) \sim f_{6,aij}(\mathbf{x})$ はそれぞれ $f_{1,aij}(\mathbf{x}) \sim f_{3,aij}(\mathbf{x})$ の飛行時間にドローンの離着陸時間を加えた値を表す。 $f_{1,ai}(\mathbf{x})$ は、トラックが $T_a^i$ に到着してから $T_{a+1}^i$ へ向けて出発する直前までに要する時間であり、 $f_{7,aij}(\mathbf{x})$ は $f_{1,ai}(\mathbf{x})$ を表現するための補助式である。 $f_{2,ai}(\mathbf{x})$ は、トラックが $T_a^i$ を出発してから $T_{a+1}^i$ での宅配を完了させるまでに要する時間、 $f_{3,ai}(\mathbf{x})$ は $f_{2,ai}(\mathbf{x})$ と各 $f_{6,aij}(\mathbf{x})$ の最大値を表す。

#### 4.5 連続宅配

上記で示した DDP の式は ドローンで宅配できる場所が 1 回の使用で 1 か所までに限定されることを前提としている。仮にこの前提が将来の技術革新により克服され、複数の場所へ連続して宅配可能になったと仮定すると、宅配コストの低減をより一層増大させることが期待できる。ここで、ドローンの連続宅配が可能な「Drone 宅配」、 「Hybrid 宅配」及び

「Multi-Agent 宅配」を、「Consecutive(連続)」の頭文字をとってそれぞれ「C-Drone 宅配」, 「C-Hybrid 宅配」及び「C-Multi-Agent 宅配」と呼称する. 図 4.11(a)~(c)にこの拡張された宅配経路の概念図をそれぞれ示す.

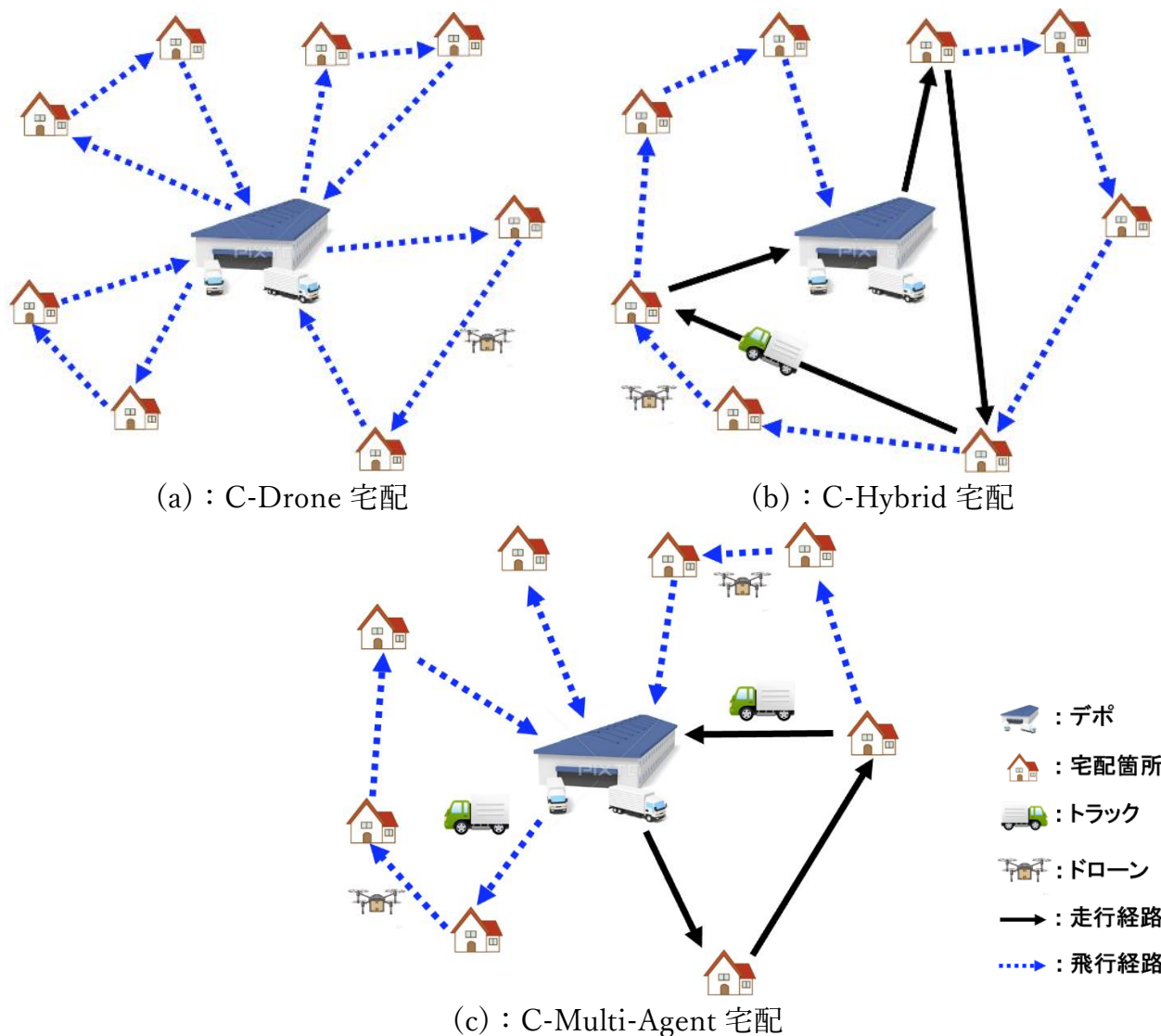


図 4.11 : 連続宅配の概念図

4.2 節で示した解構造では, 上記で示した連続宅配スタイルの経路を表現することができない. そこで, 下記に示すように解構造の一部を再定義する.

#### 4.5.1 C-Drone 宅配

式(4.4)で示した Drone 宅配経路の解構造に, 式(4.30)で示した  $u^j$  を加える.

$$u^j = [u_1^j, u_2^j, \dots, u_b^j, \dots, u_{ND_j}^j] \quad (4.30)$$

$$(j = 1, 2, \dots, B \quad b = 1, 2, \dots, ND_j)$$

上式において、 $u_b^j$ の値は $D_b^j$ での宅配を完了させたドローンの次の行先を決定する。もし、 $u_b^j = 0$ であれば $D_b^j$ での宅配を完了したドローンは $L_d$ へ帰還する。仮に、 $u_b^j = 1$ であれば、 $D_b^j$ での宅配を完了したドローンは次のWPである $D_{b+1}^j$ へ連続して宅配へ向かう。なお、 $D_{b+1}^j$ が存在しない場合、つまり $D_{ND_j}^j$ を離陸したドローンに関しては、 $L_d$ へ帰還させるしか選択肢がないため、 $u_{ND_j}^j = 0$ となる。図 4.12 は、C-Drone 宅配経路の一例を表している。

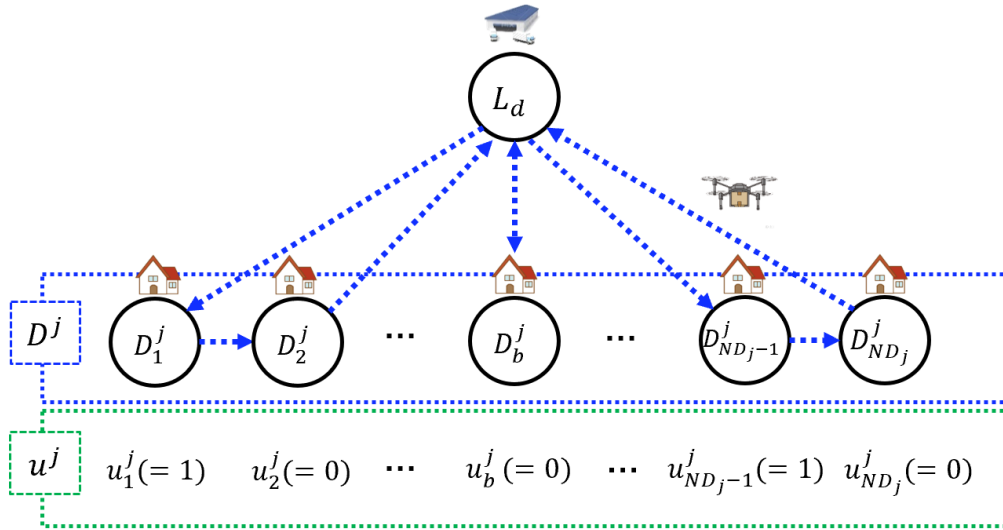


図 4.12 : C-Drone 宅配経路の解構造

上図で示したように、 $D^j$ と $u^j$ を組み合わせた解構造とすることによってドローンが連続して宅配可能な飛行経路を生成することが可能となる。この解構造の遺伝的操作方法に関しては、操作④及び⑤に加え、操作⑧を準用するものとする。

一方、解構造が変化することに伴い、式(4.18)に関しても再定義する必要がある。ここで、C-Drone 宅配の定式化を容易にするため、 $L_d$ を $e$ 回目に離陸したドローンが最初に宅配するWPを $D_b^j$ 、 $L_d$ へ帰還する直前に宅配するWPを $D_{b'}^j$ とおくと、 $b$ 及び $b'$ に対して式(4.31)及び式(4.32)を定義する。

$$q_{2e-1}^j = b \quad (4.31)$$

$$q_{2e}^j = b' \quad (4.32)$$

$$e = 1, 2, \dots, E_j (= \sum_{b=1}^{ND_j} |u_b^j - 1|)$$

上式に示すように、 $q_{2e-1}^j$ 及び $q_{2e}^j$ は、ドローンが宅配するWPの順番を意味する。これらを用いて式(4.18)を C-Drone 宅配経路問題として再定式化したものを式(4.33)に示す。なお、 $w_k^j$ は宅配先WPの荷物の重量を表す。

$$\begin{cases}
\min_{\mathbf{x}} \mathbf{F}(\mathbf{x}) = \min_{\mathbf{x}} [F_4(\mathbf{x}), F_5(\mathbf{x}), F_6(\mathbf{x})]^T \\
F_4(\mathbf{x}) = \sum_{j=1}^B \sum_{e=1}^{E_j} f_e(\mathbf{x}) \\
F_5(\mathbf{x}) = \sum_{j=1}^B \sum_{e=1}^{E_j} \left( \frac{f_e(\mathbf{x})}{V_D} + (q_{2e}^j - q_{2e-1}^j + 2) \times (t_{re} + 2t_{tl}) \right) \\
F_6(\mathbf{x}) = \sum_{j=1}^B D_{j, \text{count}} \quad D_{j, \text{count}} = \begin{cases} 1, & ND_i > 0 \\ 0, & ND_i = 0 \end{cases} \\
\text{subject to} \\
g_1(\mathbf{x}) = F_6(\mathbf{x}) \leq B \\
g_2(\mathbf{x}) = \frac{f_e(\mathbf{x})}{V_D} + 4t_{tl} \leq t_d \\
g_3(\mathbf{x}) = \sum_{k=q_{2e-1}^j}^{q_{2e}^j} w_k^j \leq W_d
\end{cases} \quad (4.33)$$

ここで、上式で用いた $f_e(\mathbf{x})$ の定義式を式(4.34)に示す。

$$f_e(\mathbf{x}) = DC_{L_d D_{q_{2e-1}^j}^j} + \sum_{k=q_{2e-1}^j}^{q_{2e}^j} DC_{D_k^j D_{k+1}^j} + DC_{D_{q_{2e-1}^j}^j L_d} \quad (4.34)$$

各関数の説明は、4.3節で示したDrone宅配の定義と同様である。

#### 4.5.2 C-Hybrid/C-Multi-Agent 宅配

Hybrid/M-Hybrid/Multi-Agent 宅配経路の解構造の一部である式(4.9)を式(4.35)のように再定義する。

$$u_a^{ij} = [u_{a,1}^{ij}, u_{a,2}^{ij}, \dots, u_{a,b}^{ij}, \dots, u_{a,h_{a,ij}}^{ij}] \quad (u_{a,b}^{ij} \in \{0,1\}, b = 1, 2, \dots, h_{a,ij}) \quad (4.35)$$

上式において、 $u_{a,b}^{ij}$ の値は $D_{a,b}^{ij}$ での宅配を完了させたドローンの次の行先を決定する。もし、 $u_{a,b}^{ij} = 0$ であれば $D_{a,b}^{ij}$ での宅配を完了したドローンは $T_a^i$ へ帰還する。仮に、 $u_{a,b}^{ij} = 1$ であれば、 $D_{a,b}^{ij}$ での宅配を完了したドローンは次のWPである $D_{a,b+1}^{ij}$ へ連続して宅配へ向かう。もし $D_{a,b+1}^{ij}$ が存在しない場合は $T_{a+1}^i$ へ向けて飛行させ、トラックと合流する。この方式に従って $T_a^i - T_{a+1}^i$ 間におけるドローンの飛行経路の一例を表したものが図 4.13 である。

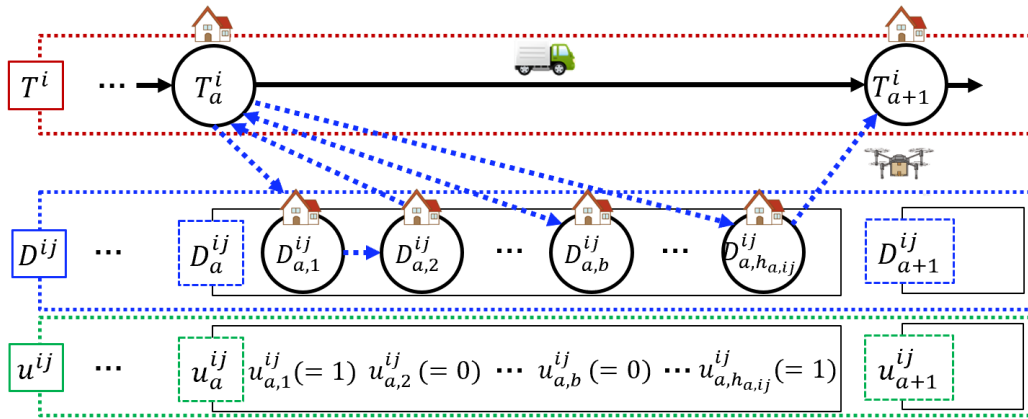


図 4.13 : C-Hybrid 宅配経路の解構造

上図で示したように、各  $D_{a,b}^{ij}$  に対して  $u_{a,b}^{ij}$  を与えることにより、ドローンが複数の  $WP$  を連続して宅配する飛行経路を表現できる。この解構造に対する遺伝的操作方法については、Hybrid/M-Hybrid/Multi-Agent 宅配で示した操作を準用するものとする。

一方、解構造が変化することに伴い、式(4.19)に関しても再定義する必要がある。ここで、C-Hybrid 宅配の定式化を容易にするため、C-Drone 宅配の場合と同様の要領で  $T_a^j$  を  $e$  回目に離陸したドローンが最初に宅配する  $WP$  を  $D_{a,b}^{ij}$ 、 $T_a^i$  へ帰還もしくは  $T_{a+1}^i$  へ向かう直前に宅配する  $WP$  を  $D_{a,b'}^{ij}$  とおき、この  $b$  及び  $b'$  に対して式(4.36)及び式(4.37)を定義する。

$$q_{a,2e-1}^{ij} = b \quad (4.36)$$

$$q_{a,2e}^j = b' \quad (4.37)$$

$$e = 1, 2, \dots, E_{a,ij} (= \sum_{b=1}^{h_{a,ij}} |u_{a,b}^{ij} - 1|)$$

上式を用いて式(4.19)を C-Hybrid 宅配経路問題として再定式化したものを式(4.38)に示す。なお、 $w_{a,k}^{ij}$  は宅配先  $WP$  である  $D_{a,k}^{ij}$  の荷物重量を表す。

$$\begin{cases}
\min_{\mathbf{x}} F(\mathbf{x}) = \min_{\mathbf{x}} [F_1(\mathbf{x}), F_2(\mathbf{x}), F_3(\mathbf{x}), F_4(\mathbf{x}), F_5(\mathbf{x}), F_6(\mathbf{x})] \\
F_1(\mathbf{x}) = \sum_{i=1}^A \sum_{a=0}^{NH_i} TC_{T_a^i T_{a+1}^i} \quad (T_0^i = T_{NH_i+1}^i = L_d = 0) \\
F_2(\mathbf{x}) = \sum_{i=1}^A \sum_{a=0}^{NH_i} \left( f_{1,ai}(\mathbf{x}) + f_{3,ai}(\mathbf{x}) + t_{re} \sum_{j=1}^B u_{a,h_{a,ij}}^{ij} \right) \\
F_3(\mathbf{x}) = \sum_{i=1}^A T_{i,Count} \quad T_{i,Count} = \begin{cases} 1, & NH_i > 0 \\ 0, & NH_i = 0 \end{cases} \\
F_4(\mathbf{x}) = \sum_{i=1}^A \sum_{j=1}^B \sum_{a=0}^{NH_i} (f_{1,aij}(\mathbf{x}) + f_{2,aij}(\mathbf{x}) + f_{3,aij}(\mathbf{x})) \\
F_5(\mathbf{x}) = \sum_{i=1}^A \sum_{j=1}^B \sum_{a=0}^{NH_i} (f_{4,aij}(\mathbf{x}) + f_{5,aij}(\mathbf{x}) + f_{6,aij}(\mathbf{x})) \\
F_6(\mathbf{x}) = \sum_{i=1}^A \sum_{j=1}^B D_{ij,Count} \quad D_{ij,Count} = \begin{cases} 1, & \sum_{a=0}^{NH_i} h_{a,ij} > 0 \\ 0, & \sum_{a=0}^{NH_i} h_{a,ij} = 0 \end{cases} \\
\text{subject to} \\
g_1(\mathbf{x}) = F_1(\mathbf{x}) \leq LD_{max} \\
g_2(\mathbf{x}) = F_2(\mathbf{x}) \leq LT_{max} \\
g_3(\mathbf{x}) = F_3(\mathbf{x}) \leq A \\
g_4(\mathbf{x}) = f_{4,aij}(\mathbf{x}) \leq t_d \\
g_5(\mathbf{x}) = f_{5,aij}(\mathbf{x}) \leq t_d \\
g_6(\mathbf{x}) = f_{6,aij}(\mathbf{x}) \leq t_d \\
g_7(\mathbf{x}) = F_6(\mathbf{x}) \leq B \\
g_8(\mathbf{x}) = \sum_{k=q_{a,2e-1}^{ij}}^{q_{a,2e}^{ij}} W_{a,k}^{ij} \leq W_d
\end{cases} \quad (4.38)$$

ここで、式(4.38)内で用いた各関数の定義式を式(4.39)～式(4.50)に示す。

$$f_{e1}(\mathbf{x}) = DC_{T_a^i D_{a,q_{a,2e-1}^{ij}}^{ij}} + \sum_{k=q_{a,2e-1}^{ij}}^{q_{a,2e}^{ij}} DC_{D_{a,k}^{ij} D_{a,k+1}^{ij}} + DC_{D_{a,q_{a,2e}^{ij}}^{ij} T_{a+1}^i} \quad (4.39)$$

$$f_{e2}(\mathbf{x}) = DC_{T_a^i D_{a,q_{a,2E_{a,ij}-1}^{ij}}^{ij}} + \sum_{k=q_{a,2E_{a,ij}-1}^{ij}}^{q_{a,2E_{a,ij}}^{ij}} DC_{D_{a,k}^{ij} D_{a,k+1}^{ij}} + DC_{D_{a,q_{a,2E_{a,ij}}^{ij}}^{ij} T_{a+1}^i} \quad (4.40)$$

$$f_{1,aij}(\mathbf{x}) = \sum_{e=1}^{E_{a,ij}-1} f_{e1}(\mathbf{x}) \quad (4.41)$$

$$f_{2,aij}(\mathbf{x}) = (1 - u_{a,b}^{ij}) \times f_{e1}(\mathbf{x}) \quad (4.42)$$

$$f_{3,aij}(\mathbf{x}) = u_{a,b}^{ij} \times f_{e2}(\mathbf{x}) \quad (4.43)$$

$$f_{4,aij}(\mathbf{x}) = \frac{f_{1,aij}(\mathbf{x})}{V_D} + 2t_{tl} \sum_{e=1}^{E_{a,ij}-1} (q_{a,2e}^{ij} - q_{a,2e-1}^{ij} + 2) \quad (4.44)$$

$$f_{5,aij}(\mathbf{x}) = (1 - u_{a,b}^{ij}) \times \left( \frac{f_{2,aij}(\mathbf{x})}{V_D} + 2t_{tl} (q_{a,2E_{a,ij}}^{ij} - q_{a,2E_{a,ij}-1}^{ij} + 2) \right) \quad (4.45)$$

$$f_{6,aij}(\mathbf{x}) = u_{a,b}^{ij} \times \left( \frac{f_{3,aij}(\mathbf{x})}{V_D} + 2t_{tl} (q_{a,2E_{a,ij}}^{ij} - q_{a,2E_{a,ij}-1}^{ij} + 2) \right) \quad (4.46)$$

$$f_{7,aij}(\mathbf{x}) = f_{4,aij}(\mathbf{x}) + f_{5,aij}(\mathbf{x}) + t_{re} \times E_{a,ij} \quad (4.47)$$

$$f_{1,ai}(\mathbf{x}) = \max[f_{7,ai1}(\mathbf{x}), f_{7,ai2}(\mathbf{x}), \dots, f_{7,aiB}(\mathbf{x})] \quad (4.48)$$

$$f_{2,ai}(\mathbf{x}) = \frac{TC_{T_a^i T_{a+1}^i}}{V_T} + t_p + t_{st} + t_{sp} \quad (4.49)$$

$$f_{3,ai}(\mathbf{x}) = \max[f_{6,ai1}(\mathbf{x}), f_{6,ai2}(\mathbf{x}), \dots, f_{6,aiB}(\mathbf{x}), f_{2,ai}(\mathbf{x})] \quad (4.50)$$

各関数の説明は、4.3節で示したHybrid宅配の定義と同様である。

## 第 5 章 PIP 法の改良

### 5.1 はじめに

第 3 章では, 制約付き多目的最適化問題の解法として GA をベースとした PIP 法(以下, 従来型 PIP 法と呼ぶ)を提案し, 第 4 章では DDP を制約付き多目的最適化問題として定式化した. この提案手法を用いて DDP を解く場合, 2 つの問題点が生じる. 1 つ目は, 従来型 PIP 法の評価関数である  $D_{is}(\mathbf{x})$  の値が適用する最適化問題次第では無限に発散し, 解の選択・淘汰が不可能になってしまうことである. 2 つ目は, 同じ問題設定であったとしても実行可能解が得られる場合と得られない場合があることや, 生成される選好解に大きなバラツキが生じてしまうことである. 本章では, これらの問題点の原因について考察するとともに, その解決方法として改良型暫定理想点法(以下, 「改良型 PIP 法」という.)を提案する.

### 5.2 従来型 PIP 法の問題点

#### 5.2.1 DBZ 問題

従来型 PIP 法の評価関数  $D_{is}(\mathbf{x})$  の値が問題設定次第で無限に発散する原因について以下に示す. 式(3.6)で示した暫定理想点と解点のユークリッド距離を表す評価関数は, 式(5.1)のように表現できる.

$$D_{is}(\mathbf{x}) = \left( \sum_{l=1}^M \left( 1 - \frac{F_l(\mathbf{x})}{F_{lpro}} \right)^2 \right)^{\frac{1}{2}} \quad (5.1)$$

ここで, 式(4.19)で示した Hybrid 宅配経路問題に従来型 PIP 法を適用すると, 解探索の過程でトラックのみによる宅配経路が実行可能解として生成された場合, ドローンの合計飛行距離  $F_4(\mathbf{x})$ , その合計飛行時間  $F_5(\mathbf{x})$ , そしてドローンの使用台数  $F_6(\mathbf{x})$  の値は

$$F_4(\mathbf{x}) = F_5(\mathbf{x}) = F_6(\mathbf{x}) = 0$$

となるので, 従来型 PIP 法の定義に従うと,  $F_4(\mathbf{x})$ ,  $F_5(\mathbf{x})$  及び  $F_6(\mathbf{x})$  の暫定最適値は

$$F_{4pro} = F_{5pro} = F_{6pro} = 0$$

となる. すると, 式(6-1)において,  $F_4(\mathbf{x})/F_{4pro}$ ,  $F_5(\mathbf{x})/F_{5pro}$  及び  $F_6(\mathbf{x})/F_{6pro}$  の項は 0 割り計算となるため, 結果的に  $D_{is}(\mathbf{x})$  の値は無限に発散する.

上記に示した例のように,  $D_{is}(\mathbf{x})$  の値が 0 割り計算によって無限に発散することを本論文では「Division By Zero (DBZ)問題」と呼ぶ. DBZ 問題が生じた場合, 解の選択・淘汰ができなくなるため, それ以降の解探索も不可能となる.

### 5.2.2 IC 問題

2.2 節で言及したように, GA における解探索では, 交叉・交配及び突然変異と呼ばれる 2 つの操作によって新たな解を生成する. 交叉・交配は 2 つの親同士(異なる親同士もあれば, 全く同一の親同士のペアとなる場合もある)を組み合わせによって子を生み出す操作である一方, 突然変異はある 1 つの個体の解構造を一部変化させることによって子を生み出す操作である. GA ではこの 2 つの操作を組み合わせることによって大域的解探索を実現するが, DDP では特に交叉・交配の操作が極めて困難である事情が存在する. その理由を以下に示す.

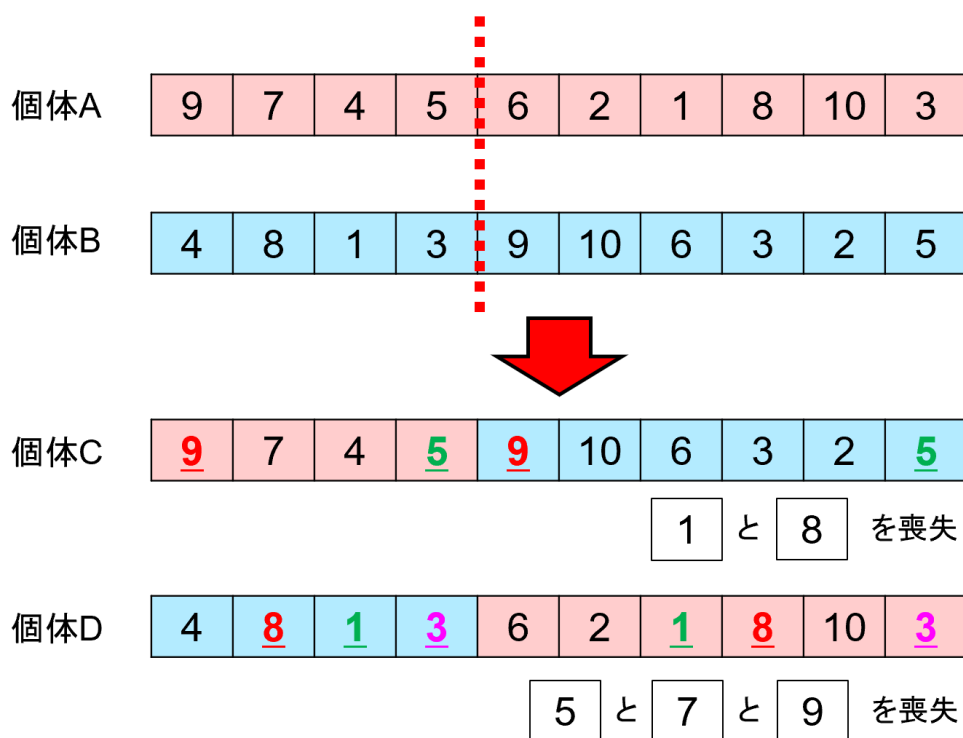


図 5.1 : 経路生成問題における「交叉・交配」の例

図 5.1 は, 1~10 までの番号を割り振られた WP を巡回する経路を表した解表現の一例である. 上図において, 例えば個体 A は, 9 の番号が割り振られた WP から出発して 9→7→4→5→6→2→1→8→10→3 の順番で各 WP を巡回し, 最終的に 9 の WP へ戻る経路を表している. 個体 A と個体 B に対して 1 点交叉を適用して個体 C 及び D という新たな個体が生成され

たと仮定した場合、赤、緑、マゼンタ色の番号で示したように同じWPを重複して巡回する経路が生成されてしまうことがわかる。それと同時に、本来経由すべきWPが重複したWPの数だけ解構造から除外されてしまっている様子も確認できる。解構造が第4章で示したようなバイナリ表現であれば問題ないが、経路生成問題の場合は解の「交叉・交配」後も重複WPの特定と除外されたWPの復元という作業に多くの計算資源を割く必要が生じる。

上記の問題点を克服する手法として、Goldbergらが提案した Partially Mapped Crossover(PMX)法, Davisらが提案した Order Crossover(OX)法, Oliverらが提案した Cycle Crossover(CX)法, Whitleyらが提案した edge recombination Crossover(EX)法, Yamamuraらが提案したサブツアー交換交叉(SXX)法, Maekawaらが提案した枝交換交叉(EXX)法など、経路生成問題における交叉演算子はこれまで数多く提案されてきた<sup>[64]</sup>。しかしながら、いずれも図5.1で示したような単純な順列となる解構造を前提としており、第4章で示したようなDDP特有の複雑な解表現にこれらの手法を応用することは困難である。これが従来型PIP法を用いてDDPを適用する際に交叉・交配による解操作を適用できない理由である。

交叉・交配による解探索が不可能である以上、DDPにおける解探索は4.2節で示すように解構造そのものを一部変更することで新たな解を生成する突然変異のみに依存せざるを得ない。この場合、本来であれば2つの遺伝的操作によって解探索しなければならないものをその片方である突然変異のみで実行することになるため、解探索の効率が著しく損なわれる恐れがある。具体的には、解探索が停滞しやすくなるので収束までの計算時間が大幅に増大すると考えられる。解探索に要することのできる時間は有限であるため、解探索が収束する以前に計算を打ち切ることもしこり得る。同じ問題設定でも実行可能解が得られる場合と得られない場合があることや、生成される選好解に大きなバラツキが生じてしまうのは、このように解探索が停滞しているにもかかわらず計算を打ち切ることによる原因があると考えられる。このように、解探索が収束する以前に計算を打ち切らざるを得ない状態に陥ることを、本論文では「未熟な収束 (Immature Convergence(IC))問題」と呼ぶ。

解探索の収束に関して所謂「初期収束」あるいは「早熟収束」といった用語が存在するが、これらは一般的なGAの解探索において、早い段階で解集団内の多様性が損なわれることによって生じる問題のことを指すものであり、IC問題とは性質を異にする<sup>48)</sup>。

### 5.3 改良型 PIP 法

上記で示したDBZ(Division By Zero)問題及びIC(Immature Convergence)問題に対処するため、DDPを宅配コストに関する単目的最適化問題に変換するとともに、GAをベースとした従来型PIP法の解探索プロセスにタブーサーチを組み合わせた新しいPIP法を提案する。本論文では、この2つの手法を取り入れた従来型PIP法を「改良型PIP法」と呼ぶことにする。

### 5.3.1 変換係数による DBZ 問題の回避

DBZ 問題は、式(5.1)の分母に 1 を加算することで容易に回避することが可能であるが、その操作が意思決定者にとって最も適当な選好解を得る上で妥当であるかどうかは議論の余地が残る。4.3.1 節でも言及したように、本研究ではより現実に近い環境下におけるドローン宅配のコスト削減効果を評価することを重視しているため、可能な限り実情報を反映した評価関数を定義することが望ましい。そうすることによって、得られた選好解にも一定の妥当性を与えることができると考える。

そこで、変換係数を用いて DDP の全ての目的関数を宅配コストに変換する手法を提案する。具体的には、DDP の各目的関数を宅配コストに変換する変換係数  $CF_I$  を定義し、式(3.4)及び(3.5)で示した解点  $\mathbf{C}_{sol}(\mathbf{x})$  及び暫定理想点  $\mathbf{C}_{ideal}$  の定義式を、式(5.2)及び式(5.3)に示すように再定義する。

$$\mathbf{C}_{sol}(\mathbf{x}) = [CF_1 \times F_1(\mathbf{x}), \dots, CF_I \times F_I(\mathbf{x}), \dots, CF_M \times F_M(\mathbf{x})] \quad (5.2)$$

$$\mathbf{C}_{ideal} = [CF_1 \times F_{1pro}, \dots, CF_I \times F_{Ipro}, \dots, CF_M \times F_{Mpro}] \quad (5.3)$$

また、上式を式(3.6)に代入することによって、 $D_{is}(\mathbf{x})$  は式(5.4)に示すように再定義される。

$$\begin{cases} \min_{\mathbf{x}} F(\mathbf{x}) = \min_{\mathbf{x}_F} D_{is}(\mathbf{x}) = \min_{\mathbf{x}_F} \|\mathbf{C}_{ideal} - \mathbf{C}_{sol}(\mathbf{x})\| = \min_{\mathbf{x}_F} \left( \sum_{I=1}^M CF_I (F_{Ipro} - F_I(\mathbf{x}))^2 \right)^{\frac{1}{2}} \\ \text{subject to } P(\mathbf{x}) = 0 \end{cases} \quad (5.4)$$

上式から分かるように、 $F_{Ipro}$  の値にかかわらず  $D_{is}(\mathbf{x})$  が無限に発散することはないため、結果的に DBZ 問題を回避することができる。また、DDP の各目的関数  $F_I(\mathbf{x})$  が宅配コストに変換されて単目的最適化問題として再定式化されるため、式(5.5)で示すように得られた宅配経路の合計宅配コストも算出することが可能になる。

$$C_{total}(\mathbf{x}) = \sum_{I=1}^M (CF_I \times F_I(\mathbf{x})) \quad (5.5)$$

ここで、変換係数  $CF_I$  を算出するための具体的な方法について以下に示す。日本トラック協会の報告書によれば、荷物の配達に直接必要な宅配コストは全体の 86% を占め、その内訳は人件費、燃料費、車両維持費、及び減価償却費となっている<sup>[18]</sup>。図 5.2 で示す円グラフは、これら 4 つのカテゴリの比率を示したものである。

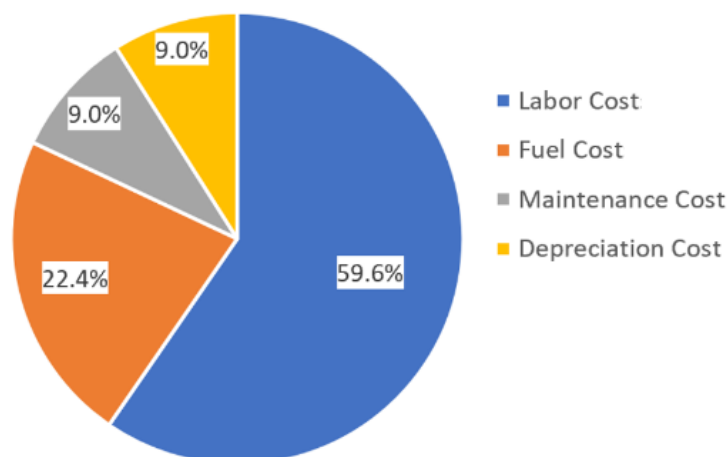


図 5.2：宅配コストの内訳

上図で示した情報により，式(5.6)で示す関係式が成り立つ．

$$\left\{ \begin{array}{l} \frac{CF_2 \times F_2(x)}{CF_1 \times F_1(x) + CF_2 \times F_2(x) + CF_3 \times F_3(x)} \times 100 = 59.6\% \\ \frac{CF_1 \times F_1(x)}{CF_1 \times F_1(x) + CF_2 \times F_2(x) + CF_3 \times F_3(x)} \times 100 = 22.4\% + 9.0\% \\ \frac{CF_3 \times F_3(x)}{CF_1 \times F_1(x) + CF_2 \times F_2(x) + CF_3 \times F_3(x)} \times 100 = 9.0\% \end{array} \right. \quad (5.6)$$

ここで，通常の宅配サービスにおいて，ドライバーは最短経路となるように宅配作業を実施していると仮定すれば，式(5.6)における $F_1(x)$ 、 $F_2(x)$ 及び $F_3(x)$ の値は，式(4.17)で示したトラック宅配経路問題を $F_1(x)$ のみに関する最適化問題として解くことによって得ることが可能である．図 5.3 で示した宅配経路図は，実際に GA を用いて生成したトラック宅配の最短経路である．各 WP に添えられた数字は宅配順序である．計算の結果，

$$F_1(x) = 81.2 \text{ [km]}$$

$$F_2(x) = 3.46 \text{ [h]}$$

$$F_3(x) = 1 \text{ [台]}$$

となることがわかった．これらの値を式(5.6)に代入すると， $CF_1 \sim CF_3$ が次のように得られる．

$$CF_1 = 0.0213 \text{ [円/m]}$$

$$CF_2 = 0.264 \text{ [円/s]}$$

$$CF_3 = 497 \text{ [円/台]}$$

ドローンに関連する変換係数については，本論文ではトラックに対するドローンの相対コスト比 $R$ で決定すると仮定する．このとき， $CF_4 \sim CF_6$ は次のように得られる．

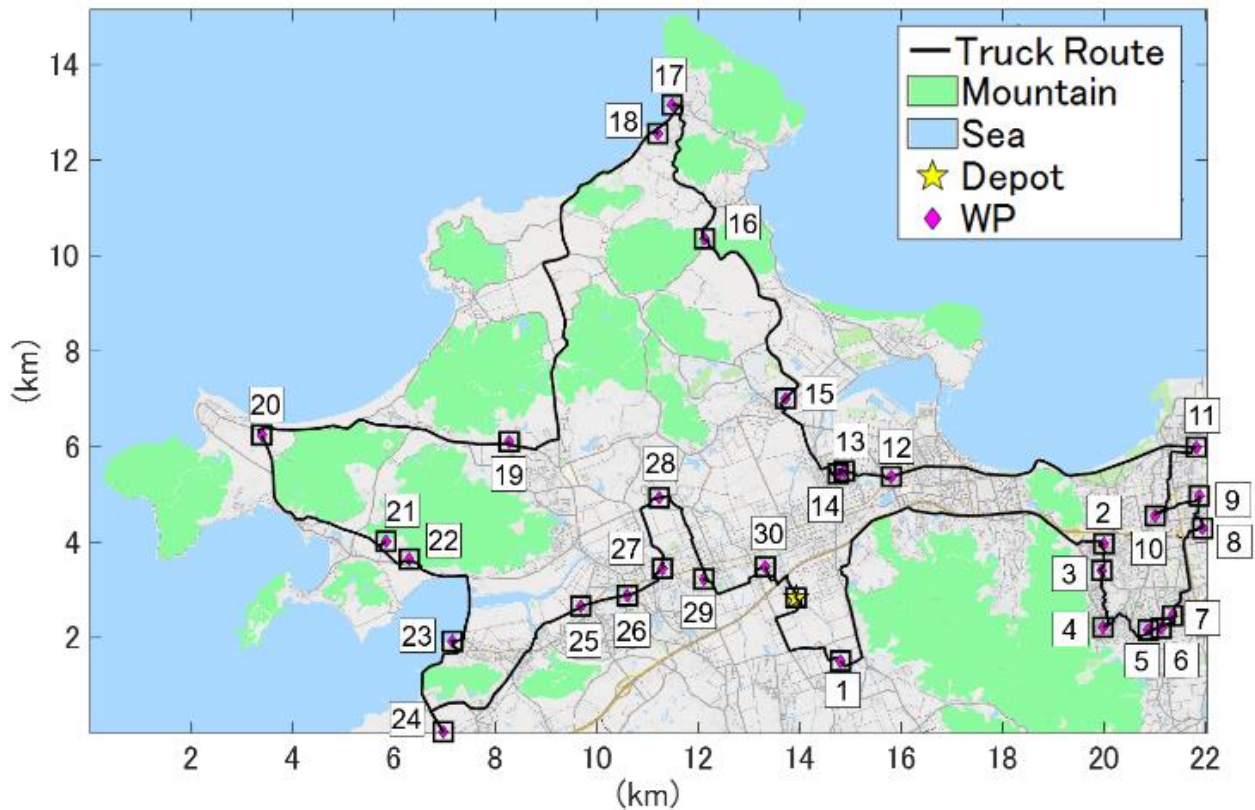


図 5.3 : GA を用いて生成した最短トラック宅配経路

$$CF_4 = R \times CF_1[\text{円/m}] \quad (5.7)$$

$$CF_5 = R \times CF_2[\text{円/s}] \quad (5.8)$$

$$CF_6 = R \times CF_3[\text{円/台}] \quad (5.9)$$

以上の示したように、変換係数は宅配サービスに関する実情報に基づいて推定した。この計算手法は、評価関数 $D_{is}(\mathbf{x})$ に実情報を反映させることができるので、重み係数などのパラメータ調整を試行錯誤する必要がない利点がある。

### 5.3.2 タブーサーチによる IC 問題への対処

IC 問題は、突然変異のみに依存した解探索を実行することで解探索が収束するまでの計算時間が大幅に増大し、結果的に解探索が停滞している状態で計算を打ち切ることにより起きている。この問題に対処するため、解探索が停滞した場合は、その解をタブーリスト(以降では $TL$ と表記する)に記録した上で解集団内から除外する操作を行う。その後、解集合内にある他の解をランダムに選択し、その選択された解をその時点における最良解とする。こうすることで、一時的に解探索を改悪することにはなるが、まだ探索していない他の解空間へ強制的に移動することで解探索の停滞を回避することが可能となる。その後、一度停滞した解へ回帰することを防ぐために、 $TL$ に存在する解が解集団内に生成された場合は常に除外し続けるように操作する。このような手法をタブーサーチ(以下、 $TS$ と呼ぶ)という(2.2節

参照). ただし, TS は本来, 局所的最適解からの脱出を意図して提案された手法であるため, 本論文で指摘したような IC 問題においても有効であるかどうかは検証する必要がある. 次節以降では, その妥当性評価について示す.

以下に TS を組み合わせた改良型 PIP 法の解探索プロセスを示す.

### <Procedure 1> (実行可能解の生成)

- <Step 1> TLを初期化し, 世代 $g \leftarrow 0, g' \leftarrow 0$ とする.
- <Step 2> ランダムに生成した $s$ 個の初期個体 $\mathbf{x}$ の集合を $p$ とし, 世代 $g \leftarrow 1$ とする.
- <Step 3>  $P(\mathbf{x}') \leq P(\mathbf{x})$  ( $\mathbf{x}' \in p, \forall \mathbf{x} \in p$ )となる $\mathbf{x}'$ を $\mathbf{x}^* \leftarrow \mathbf{x}'$ とする.
- <Step 4>  $P(\mathbf{x}^*) = 0$ のとき,  $\mathbf{x}_F^* \leftarrow \mathbf{x}^*$ として<Procedure 2>へ進む.
- <Step 5>  $\mathbf{x}^*$ とその遺伝的操作により生成した $(s-1)$ 個の $\mathbf{x}$ の集合を $p'$ とし,  $p \leftarrow p'$ ,  $g \leftarrow g+1$ とする.
- <Step 6>  $g > G_{max}$ ならば<Step 11>に進む.
- <Step 7> TLに含まれる $\mathbf{x}$ を $p$ から消去する.
- <Step 8>  $g' \leq G_{tabu}$ ならば<Step 3>に戻る.
- <Step 9>  $g' > G_{tabu}$ であれば,  $\mathbf{x}^*$ をTLに追加する.
- <Step 10>  $\mathbf{x}^*$ 以外の $\mathbf{x}$ を $p$ から任意に選択して $\mathbf{x}^* \leftarrow \mathbf{x}$ とする. その後,  $g' \leftarrow 0$ として<Step 5>に戻る.
- <Step 11> 実行可能解の生成は困難とみなし, <Procedure 1>を終了して問題設定を再考する.

### <Procedure 2> (多目的最適化)

- <Step 12> TLを初期化し,  $g' \leftarrow 0$ とする.
- <Step 13>  $\mathbf{x}_F^*$ とその遺伝的操作により生成した $(s-1)$ 個の $\mathbf{x}$ の集合を $p'$ とし,  $p \leftarrow p'$ ,  $g \leftarrow g+1$ とする.
- <Step 14>  $F_l(\mathbf{x}_F') \leq F_l(\mathbf{x}_F)$  ( $\mathbf{x}_F' \in p, \forall \mathbf{x}_F \in p$ )となる $\mathbf{x}_F'$ を選択し,  $F_{l_{pro}} \leftarrow F_l(\mathbf{x}_F')$ とする.
- <Step 15>  $D_{is}(\mathbf{x}_F'') \leq D_{is}(\mathbf{x}_F)$  ( $\mathbf{x}_F'' \in p, \forall \mathbf{x}_F \in p$ )となる $\mathbf{x}_F''$ を選択し,  $\mathbf{x}_F^* \leftarrow \mathbf{x}_F''$ とする.
- <Step 16>  $g > G_{max}$ ならば<Step 21>に進む.
- <Step 17> TLに含まれる $\mathbf{x}_F$ を $p$ より消去する.
- <Step 18>  $g' \leq G_{tabu}$ ならば<Step 13>に戻る.
- <Step 19>  $g' > G_{tabu}$ ならば, TLに $\mathbf{x}_F^*$ を追加する.
- <Step 20>  $p$ より $\mathbf{x}_F^*$ 以外の $\mathbf{x}_F$ を任意に選択して $\mathbf{x}_F^* \leftarrow \mathbf{x}_F$ とする. もし $p$ 内に $\mathbf{x}_F^*$ 以外の $\mathbf{x}_F$ が存在しないときは, TLより $\mathbf{x}_F^*$ を除外して<Step 13>に戻る. それ以外の場合は,  $g' \leftarrow 0$ として<Step 13>に戻る.
- <Step 21>  $C_{total}(\mathbf{x}_{best}) \leq C_{total}(\mathbf{x}_F)$  ( $\mathbf{x}_{best} \in TL, \forall \mathbf{x}_F \in TL$ )となる $\mathbf{x}_{best}$ を選好解とする.

## 5.4 妥当性評価

5.3 節で示した改良型 PIP 法の妥当性を評価するため、いくつかの検証を行う。1 つ目は、TS を組み合わせた GA によってベンチマーク問題である TSPLIB を解き、その生成解と最適解の比較を行う<sup>[101]</sup>。2 つ目は、従来型 PIP 法と改良型 PIP 法をそれぞれ用いて Hybrid 宅配経路問題を解き、その計算結果を比較する。

### 5.4.1 TSPLIB への適用結果

ベンチマーク問題である TSPLIB を用いて GA と TS の混成手法(「GA+TS」と表記する。)の有用性を検証する。使用した TSPLIB は berlin52, eli76, lin105, ch150 及び KroA200 とする<sup>[102]</sup>。これらの名称の最後に添えられている数字は、経由すべき点の個数を表している。この解探索において適用する解表現とその遺伝的操作については、第 4 章のトラック宅配経路の生成に関して述べた手法を準用するものとする。つまり、この節における GA の解探索についても、「交叉・交配」を使用しない「突然変異」のみによる操作を適用するものとする。こうすることで、突然変異のみの解探索によって生じると考えられる IC 問題の対処方法として TS が有効であるかどうかを確認することができる。それに伴い、GA 単体を適用する場合についても突然変異による操作のみを適用する。

各ベンチマーク問題に対して GA 単体と GA+TS をそれぞれ適用して得られた解と既知の最適解を比較した結果を図 5.4 に示す。なお、解探索における母集団内の個体数を  $P_n = 200$ 、最大世代数を  $G_{max} = 500,000$  とし、最大世代数に到達した時点で探索を終了するものとする。また、選択操作に関しては上位個体のみを次世代へ継承する単純なエリート戦略を適用する。その際、GA 単体におけるエリート数は  $E_n = 20$  に設定した。

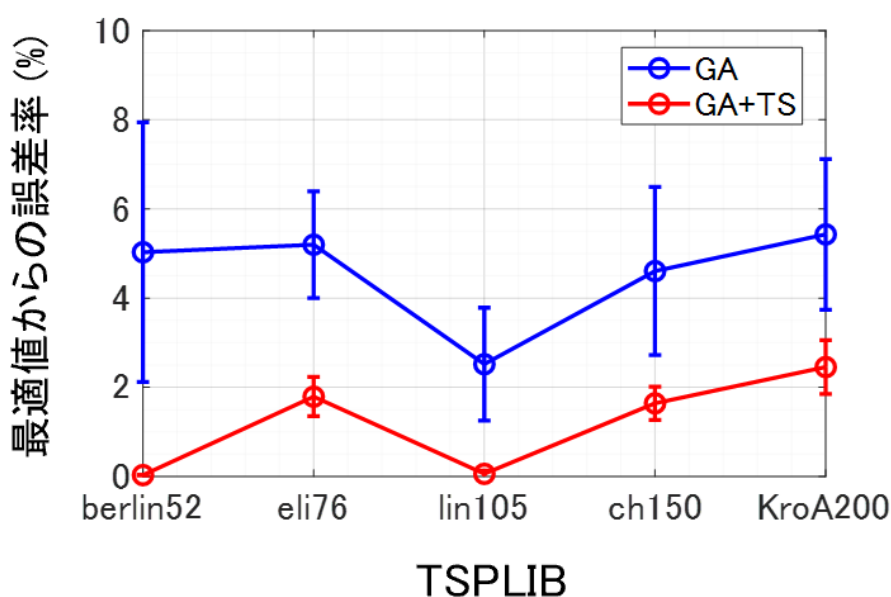
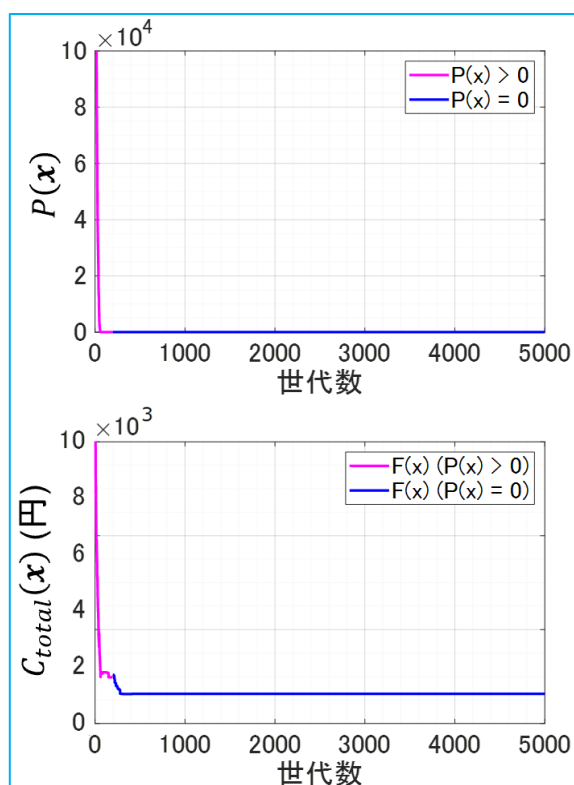


図 5.4 : GA と GA+TS の比較

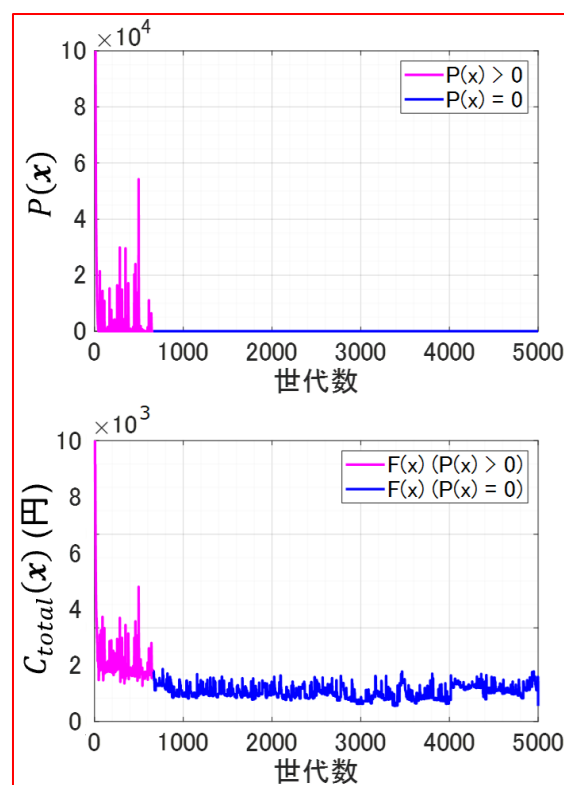
上図は、GA 単体(青色のライン)及び GA+TS(赤色のライン)をそれぞれ用いて得られた解と最適解の距離コストを比較して、その誤差率の大きさをベンチマーク問題ごとにプロットしてグラフ化したものである。プロットされた点は試行回数 10 回の平均値であり、I 字のラインはその標準偏差を表す。この結果から分かるように、GA 単体で生成した解の最適値からの平均誤差率は 2.5%~5.4%であるのに対し、GA+TS を用いて生成した解の平均誤差率は 0%~2.45%である。このことから、明らかに後者の解探索性能が優れていることが確認できる。また、標準偏差に関しても、GA 単体が  $\pm 1.2\%$ ~ $\pm 2.9\%$ であるのに対して、GA+TS は 0%~ $\pm 0.60\%$ である。したがって、後者の方が前者より半分以上バラつきが小さい。以上のことから、GA+TS の方が GA より優れた解を生成できる可能性が高いことがいえる。

#### 5.4.2 DDP への適用結果

Hybrid 宅配経路問題を従来型 PIP 法及び改良型 PIP 法をそれぞれ用いて解いた結果を比較する。生成された Hybrid 宅配経路の詳細については第 6 章で説明する。この節では、改良型 PIP 法の有用性を説明するため、両手法を適用した際の  $P(x)$  と  $C_{total}(x)$  の変移を表すグラフを図 5.5(a)及び(b)に示す。ここで示す 4 つのグラフは、 $P(x) > 0$  の部分をマゼンタ色、 $P(x) = 0$  の部分を青色で表されている。表 5.1 には計算条件を示す。



(a) 従来型 PIP 法の適用結果



(b) 改良型 PIP 法の適用結果

図 5.5：従来型及び改良型 PIP 法の適用結果

表 5.1 : 計算条件

項目	値	単位	項目	値	単位
$P_n$	100	[個]	$t_d$	30	[min]
$G_{tabu}$	10	[世代]	$t_{tl}$	30	[s]
$G_{max}$	5000	[世代]	$t_{re}$	30	[s]
$A$	1	[台]	$t_p$	2	[min]
$B$	1	[台]	$W_d$	2	[kg]
$N_{WP}$	30	[地点]	$w_{a,b}^{ij}$	1~2	[kg]
$V_t$	10	[m/s]	$LD_{max}$	100	[km]
$V_d$	15	[m/s]	$LT_{max}$	4	[h]
$R$	0.1		$E_n$	10	[個]

図 5.5(a)の $P(\mathbf{x})$ に関するグラフについて、従来型 PIP 法では約 200 世代を超えると $P(\mathbf{x}) = 0$ となる実行可能解の生成に到達していることがわかる。この間における $C_{total}(\mathbf{x})$ のグラフのマゼンタ色の部分について注目すると、解探索当初は一様に減少していたグラフが僅かに上下している様子を確認できる。これは、従来型 PIP 法の第 1 段階における解探索が、 $P(\mathbf{x})$ の最小化のみを優先する一方、 $C_{total}(\mathbf{x})$ の増減に関しては全く考慮していないためである。実行可能解が生成されてからは約 10 数世代の間は $C_{total}(\mathbf{x})$ の改善が見られるが、その後 300 世代から 5000 世代に到達するまでの間は一貫して値が変化していない。この間、 $P(\mathbf{x}) = 0$ の状態は維持されているので、想定通りの解探索を実行できているといえる一方で、最終的に得られた解が局所的最適解なのか、一時的な解探索の停滞の末に得られた解なのかは判別できない。しかしながら、解探索の大部分が停滞していたことはこのグラフから明らかである。

図 5.5(b)の $P(\mathbf{x})$ に関するグラフについては、 $P(\mathbf{x}) = 0$ となる実行可能解が得られるまでの間、グラフが上下に振動している様子を確認できる。これは、 $P(\mathbf{x})$ の値が世代数 $G_{tabu}$ にわたって停滞した場合にタブーリスト $TL$ へその解が追加され、一時的に解探索を改悪する方向へ進めたために生じた現象といえる。その後、700 世代を超えたあたりで実行可能解の生成に成功している。この間、 $C_{total}(\mathbf{x})$ の値も上下に増減しているが、これについても同様の理由によるものである。実行可能解が生成された後については、タブーリスト $TL$ に解が追加される度に解集団内からそれ以外の実行可能解を 1 つランダムに選択してそれ以外の個体は除外されるので、 $P(\mathbf{x}) = 0$ の状態を保持したまま $C_{total}(\mathbf{x})$ の値が小刻みに変動している。5000 世代に到達した時点で $TL$ 内に存在する解の中で最も $C_{total}(\mathbf{x})$ の値が小さくなる解を最適解とするので、 $C_{total}(\mathbf{x})$ のグラフの最後はこれまでで最も低い位置に降下している様子が確認できる。

従来型 PIP 法及び改良型 PIP 法を用いて得られた最適解をそれぞれ $\mathbf{x}_1, \mathbf{x}_2$ とおくと、

$$C_{total}(x_1) = 4631[\text{円}]$$

$$C_{total}(x_2) = 4333[\text{円}]$$

であった。この結果から、明らかに後者の方が優れた解が生成されていることが確認できる。同様の計算を計 100 回繰り返して比較した統計結果を図 5.6 に示す。

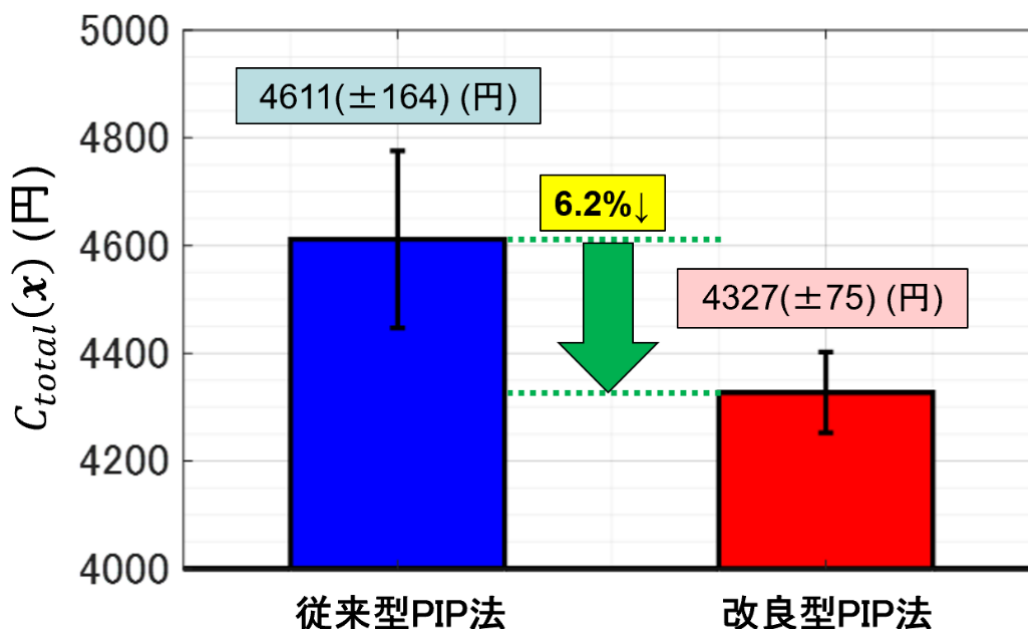


図 5.6 :  $C_{total}(x)$ に関する従来型 PIP 法と改良型 PIP 法の比較

上図の各棒グラフに記載された I 字のラインは標準偏差を表す。この計算結果から、改良型 PIP 法が従来型 PIP よりも平均して約 6.2% 宅配コストの小さい Hybrid 宅配経路を生成していることがわかる。また、解のバラツキについても半分以下となっている。この計算結果より、改良型 PIP 法の有用性が示された。

#### 5.4.3 実行可能解の生成確率

この節では、Hybrid 宅配問題において改良型 PIP 法によって実行可能解の生成確率がどの程度改善されるのかを検証する。ここで、従来型 PIP 法では 100 回の解探索で計 36 回しか実行可能解を得ることのできなかつた制約条件を以下に示す。

$$LD_{max} \leq 50[\text{km}]$$

$$LT_{max} \leq 2.78[\text{h}]$$

上記以外の計算条件については表 5.1 と同様とする。計算回数は同じく 100 回とする。図 5.7 はその比較結果を表す。

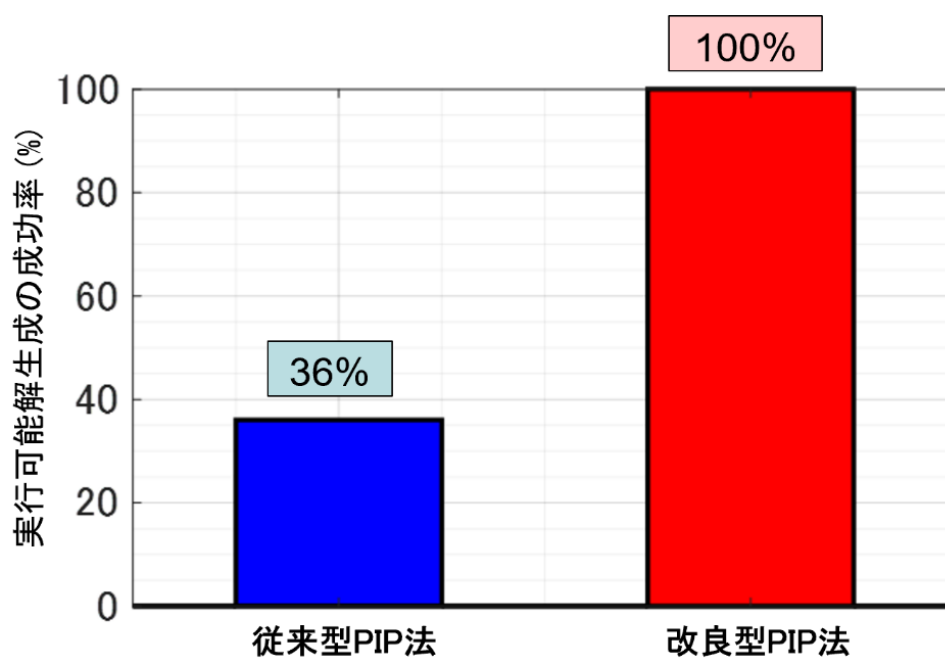


図 5.7：実行可能解の生成確率の比較

上図より改良型 PIP 手法では実行可能解の生成確率が 100% に改善されることがわかる。この計算結果から、従来型 PIP 法に TS を組み合わせることによって解探索第 1 段階の実行可能解の生成及び第 2 段階の多目的最適化の解探索性能が共に改善されることが示された。

## 第6章 計算結果と考察

### 6.1 はじめに

本章では、第5章で提案した改良型 PIP 法を用いて第4章で示した各宅配スタイルの経路生成問題を解くとともに、生成した Truck 宅配経路を基準としてその他の宅配経路を比較することで、ドローン宅配サービスによるコスト削減効果の定量的評価を目的とする。

はじめに、改良型 PIP 法を用いて Truck 宅配経路(トラックのみを用いたスタイル), Drone 宅配経路(ドローンのみを用いたスタイル), C-Drone 宅配経路(複数の WP を連続して宅配可能なドローンを使用した Drone 宅配), Hybrid 宅配経路(トラックとドローンを組み合わせたスタイル), C-Hybrid 宅配経路(複数の WP を連続して宅配可能なドローンを使用した Hybrid 宅配)及び M-Hybrid 宅配経路(複数のドローンを使用する Hybrid 宅配)をそれぞれ生成し、目的関数ごとにその比較を行う。

次に、ドローン宅配サービスのコスト削減効果が問題設定に大きく依存していると想定されるので、 $R$ (トラックに対するドローンの相対コスト比), ドローンの飛行速度 $V_D$ , 離着陸時間 $t_{cl}$ 及び飛行可能時間 $t_a$ の計4つのパラメータをそれぞれ単独で増減させ、それによってコスト削減効果がどの程度変化するかをグラフ化し、それぞれの影響について考察する。

最後に、複数のトラックとドローンを使用する Multi-Agent 宅配経路問題について考察する。具体的には、地図上の WP 数を増やして必然的に複数のエージェントにタスクを分散させるよう問題設定を変更する。その後、 $R$ の値に応じて Truck 宅配に対する Multi-Agent 宅配経路のコスト削減効果がどのように変化するか解析する。

これらの解析を通じて、ドローン宅配サービスのコスト削減効果に関する知見を得る。

### 6.2 宅配経路生成

以下に改良型 PIP 法を用いて生成した Truck 宅配, Drone 宅配, C-Drone 宅配, Hybrid 宅配, C-Hybrid 宅配及び M-Hybrid 宅配経路図を示す。計算条件については、表 6.1 及び表 6.2, 各宅配経路の評価値については表 6.3 にそれぞれ示す。

表 6.1 : 共通の計算条件

項目	値	単位	項目	値	単位
$P_n$	100	[個]	$t_d$	30	[min]
$G_{tabu}$	10	[世代]	$t_{tl}$	30	[s]
$G_{max}$	5000	[世代]	$t_{re}$	30	[s]
$N_{WP}$	30	[地点]	$t_p$	2	[min]
$V_t$	10	[m/s]	$w_a^i, w_b^j, w_{a,b}^{ij}$	2	[kg]
$V_d$	15	[m/s]	$LD_{max}$	100	[km]
$R$	0.1		$LT_{max}$	8.33	[h]

表 6.2 : 宅配スタイルごとの計算条件

項目	宅配スタイルと値						単位
	Truck	Drone	C-Drone	Hybrid	C-Hybrid	M-Hybrid	
$A$	1	0	0	1	1	1	[台]
$B$	0	1	1	1	1	2	[台]
$W_d$	0	2	2	2	4	4	[kg]

表 6.3 : 各宅配経路の目的関数値と宅配コストの対比表

項目	宅配スタイルと評価値						単位
	Truck	Drone	C-Drone	Hybrid	C-Hybrid	M-Hybrid	
$F_1(\mathbf{x})$ :トラックの合計走行距離	81.2	0	0	52.9	45.3	47.7	[km]
$F_2(\mathbf{x})$ :トラックの合計走行時間	3.46	0	0	2.33	2.14	1.94	[h]
$F_3(\mathbf{x})$ :トラックの使用台数	1	0	0	1	1	1	[台]
$F_4(\mathbf{x})$ :ドローンの合計飛行距離	0	381.2	214.6	76.6	80.1	127.4	[km]
$F_5(\mathbf{x})$ :ドローンの合計飛行時間	0	8.01	4.72	1.92	2.02	3.03	[h]
$F_6(\mathbf{x})$ :ドローンの使用台数	0	1	1	1	1	2	[台]
$C_{total}(\mathbf{x})$ :合計宅配コスト	5,517	1,629	955	4,241	3,912	4,012	[円]

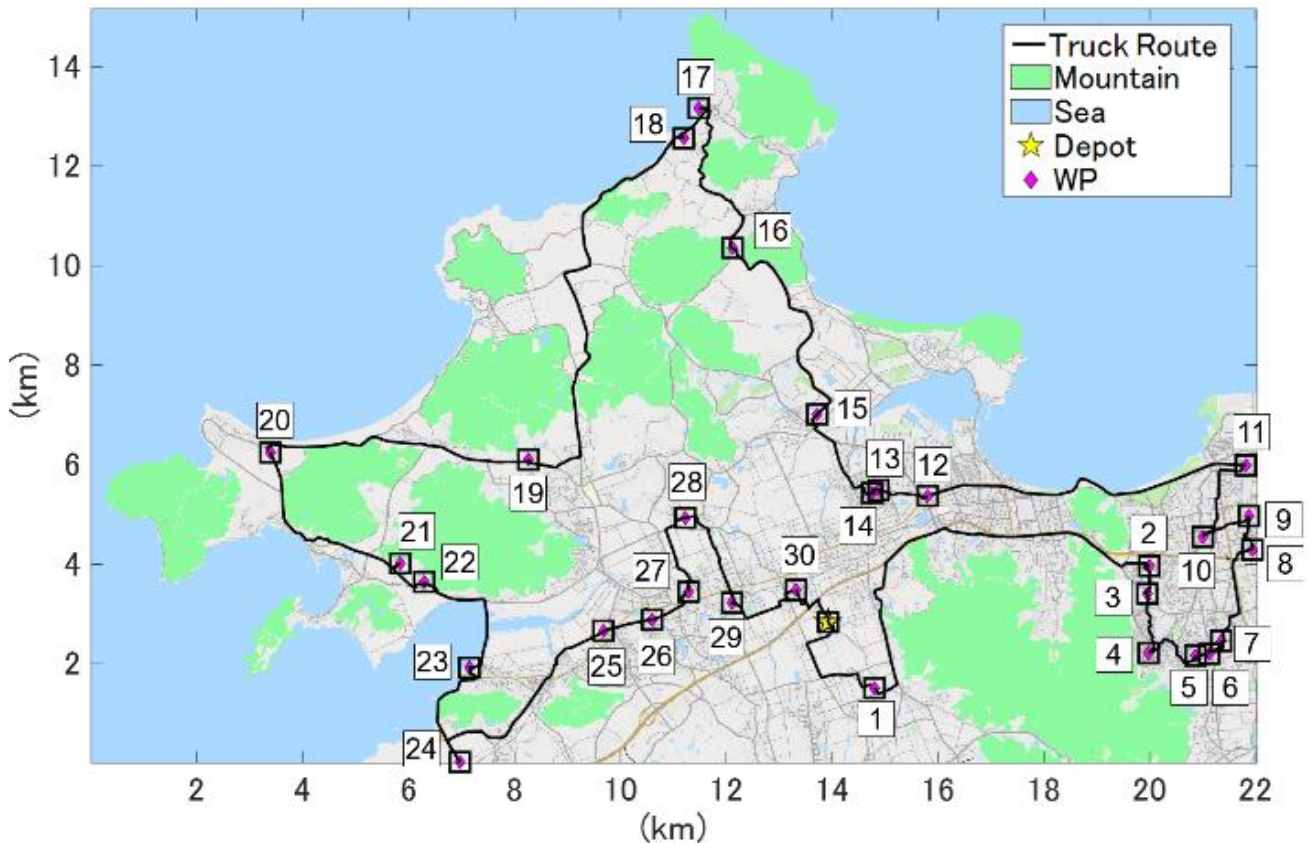


図 6.1 : Truck 宅配経路

図 6.1 は、改良型 PIP 法を用いて生成された Truck 宅配経路である。この経路図は、図 5.3 で示した宅配経路と同一であった。つまり、 $F_1(\mathbf{x}) \sim F_3(\mathbf{x})$  ( $F_2(\mathbf{x})$  はトラックの合計走行時間、 $F_3(\mathbf{x})$  はトラックの使用台数) の同時最適化によって得られる宅配経路と  $F_1(\mathbf{x})$  (トラックの合計走行距離) のみを最適化して得られる経路が一致したということである。このような結果が得られる理由を考察する。

$F_3(\mathbf{x})$  の値は、トラックの使用可能台数が  $A = 1$  に設定されていたことから、この問題設定下では実質的に  $F_3(\mathbf{x}) = 1$  で固定化される。そのため、式(4.17)で示した最適化問題は、結果的に  $F_1(\mathbf{x})$  及び  $F_2(\mathbf{x})$  の 2 目的最適化問題であることがわかる。また、Truck 宅配経路問題における  $F_1(\mathbf{x})$  と  $F_2(\mathbf{x})$  の最適化はそれぞれ同義である。なぜなら、単なる最短経路問題では、移動距離を最短にすれば、移動時間も結果的に最短になるからである。つまり、 $F_1(\mathbf{x})$  と  $F_2(\mathbf{x})$  はトレードオフの関係とはならない。このような要因により生成された 2 つの経路図が一致したと考えられる。

Truck 宅配経路の各評価値については、表 6.3 に示す。以降に示すドローン宅配サービスの宅配経路に関しては、生成された Truck 宅配経路の計算結果を比較対象とすることでそのコスト削減効果を評価する。

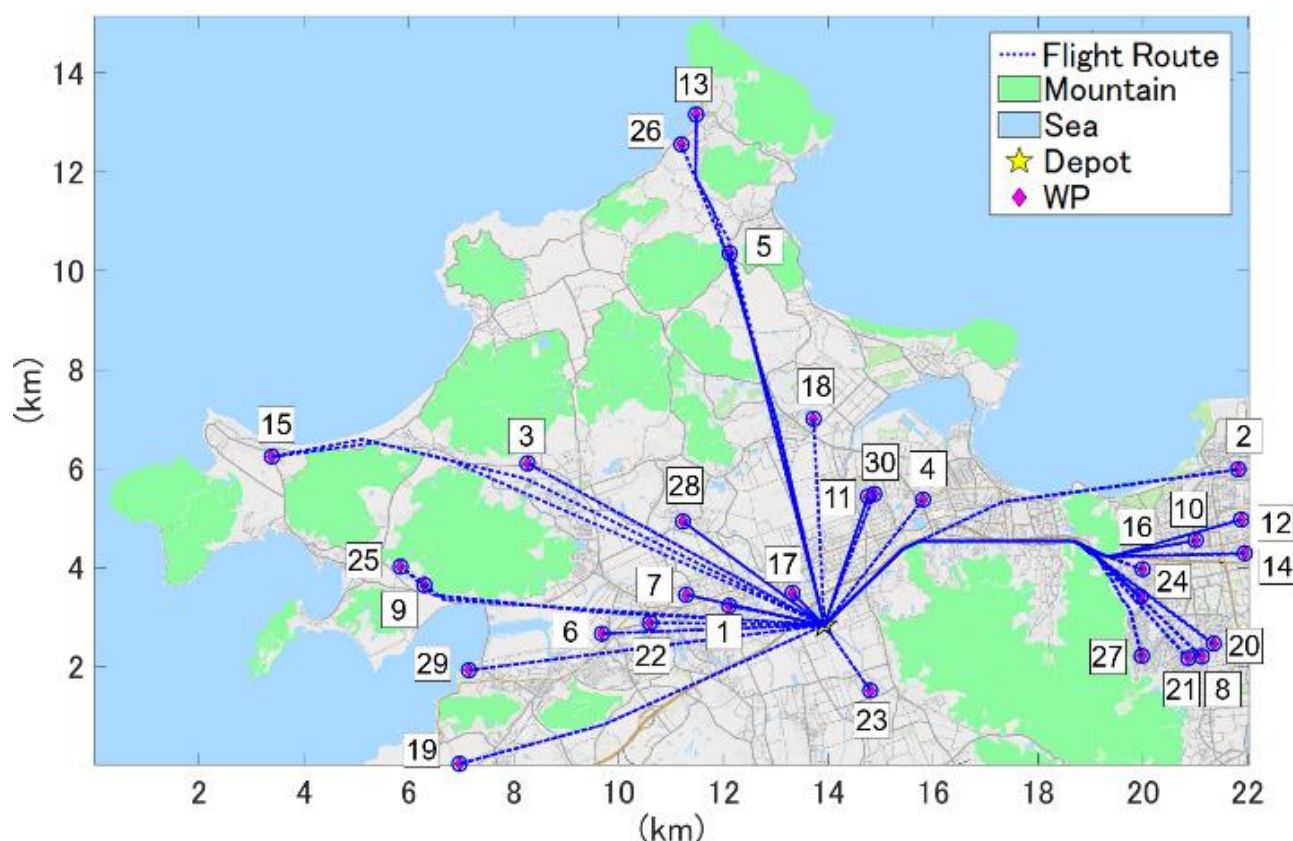


図 6.2 : Drone 宅配経路

図 6.2 は、改良型 PIP 法を用いて生成した Drone 宅配経路である。図 4.2 で示した Drone 宅配の概念図と同様、デポ(上図に示した放射状の青色の破線の中心点)から各宅配場所へドローンを使って 1 か所ずつ荷物を運ぶ飛行経路が生成されている様子が確認できる。このとき、ドローンの飛行経路は、淡緑色で表された山のエリアを避けるように生成されている。これは、山のエリアを障害物として避けるよう事前に指定していたためである。

ここで、表 6.3 より、Drone 宅配経路と Truck 宅配経路を  $C_{total}(\mathbf{x})$  について比較すると

$$C_{total}(\mathbf{x}) : \left( 1 - \frac{1,629[\text{円}]}{5,517[\text{円}]} \right) \times 100 = 70.5\%$$

となり、Drone 宅配によって 70%以上宅配コストを削減できていることがわかる。これほど大幅に宅配コストを削減できた理由として、トラックに対するドローンの相対コスト比が  $R = 0.1$  に設定されていることと、人件費に相当する  $F_2(\mathbf{x})$  が宅配コストに含まれないよう設定していることが挙げられる。一方で、ドローン 1 台を約 8 時間かけて 400[km] 近く飛行させなければならない宅配経路となっていることを考慮すれば、安全性や作業効率、実現可能性など様々な観点から鑑みて決して理想的とは言えない。また、問題設定を変更すれば得られる効果も大幅に変化する可能性があるため、この結果だけを以て Drone 宅配が優れているとはいえない。

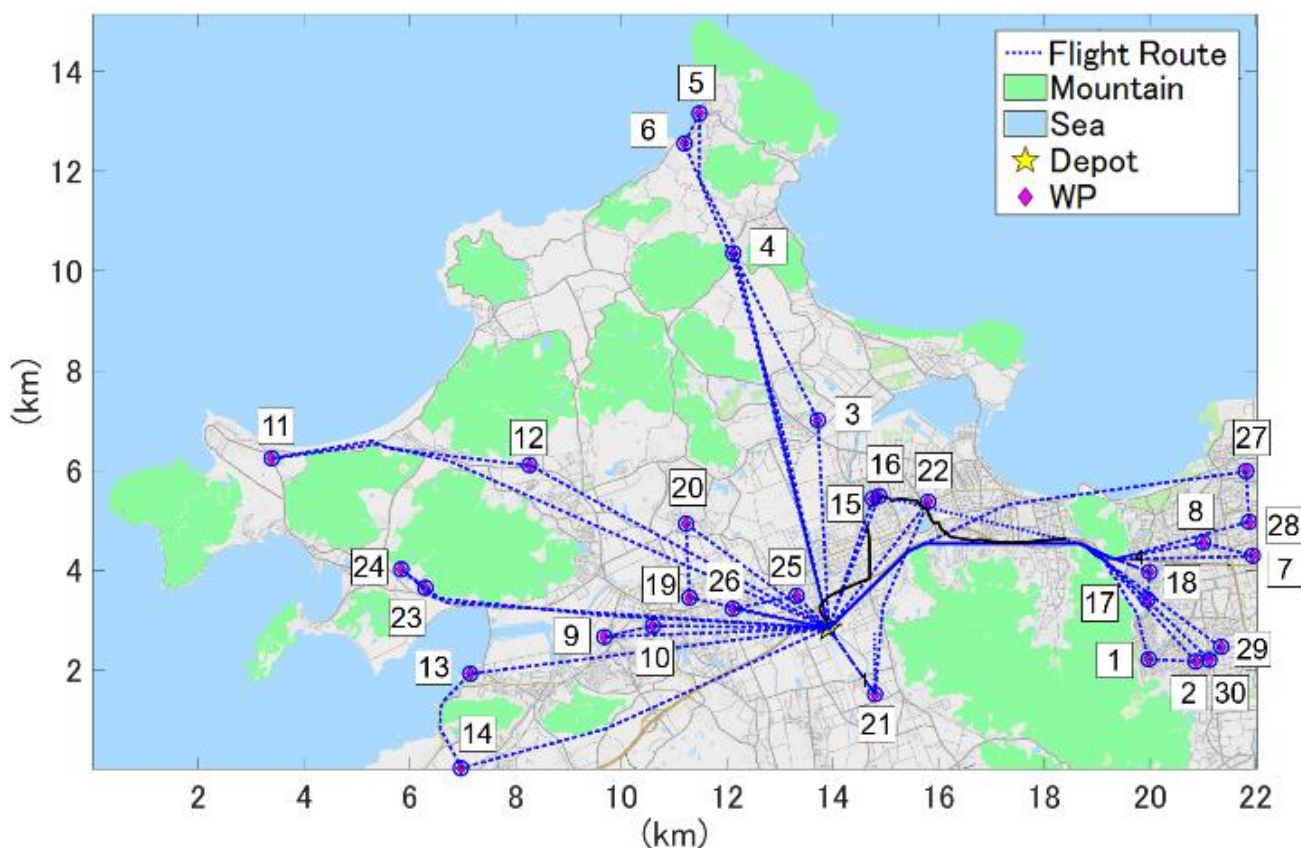


図 6.3 : C-Drone 宅配経路

図 6.3 は、改良型 PIP 法を用いて生成した C-Drone 宅配経路である。図 6.2 で示した Drone 宅配経路では、ドローンは 1 か所宅配するたびにデポへ帰還していたが、上図で示した C-Drone 宅配経路は、例えばデポ→13 番→14 番→デポのように 2 か所の WP へ連続して宅配した後にデポへ帰還する経路が生成されていることがわかる。

ここで、表 6.3 より、C-Drone 宅配経路と Truck 宅配経路を  $C_{total}(\mathbf{x})$  について比較すると、

$$C_{total}(\mathbf{x}) : \left( 1 - \frac{955[\text{円}]}{5,517[\text{円}]} \right) \times 100 = 82.7\%$$

となり、Drone 宅配よりも多く宅配コストを削減していることがわかる。また、C-Drone 宅配経路と Drone 宅配を  $F_4(\mathbf{x})$  及び  $F_5(\mathbf{x})$  について比較すると、

$$F_4(\mathbf{x}) : \left( 1 - \frac{214.6[\text{km}]}{381.2[\text{km}]} \right) \times 100 = 43.7\%, \quad F_5(\mathbf{x}) : \left( 1 - \frac{4.72[\text{h}]}{8.01[\text{h}]} \right) \times 100 = 41.1\%$$

となり、C-Drone 宅配の方が Drone 宅配より飛行距離を 43.7%、飛行時間を 41.1%削減できることがわかった。このことから、連続宅配が可能なドローンを使用する方がより優れたコスト削減効果が得られることが示されたといえる。

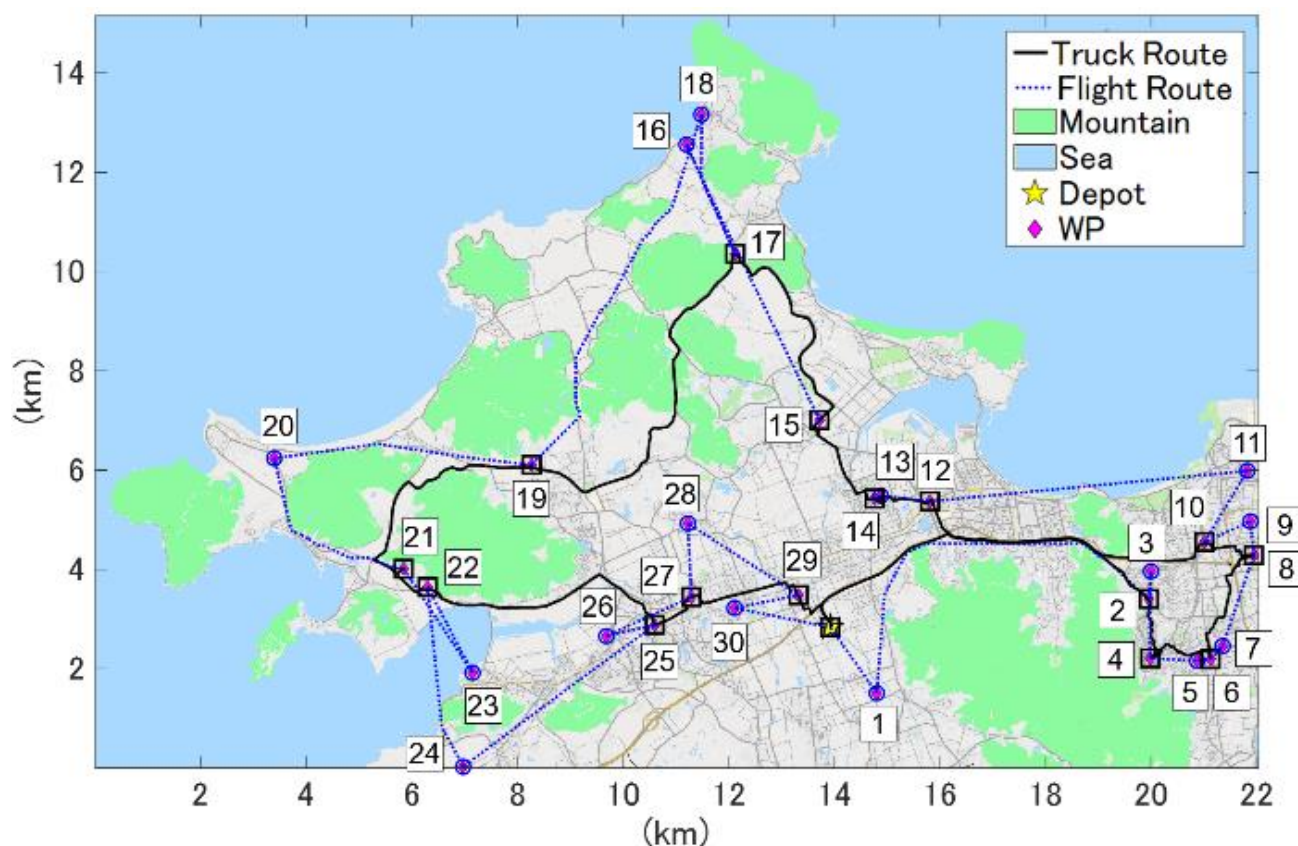


図 6.4 : Hybrid 宅配経路

図 6.4 は、改良型 PIP 法を用いて生成した Hybrid 宅配経路である。この図より、例えばトラックが 19 番→21 番に向かう間、ドローンは 20 番での宅配へ向かう経路が描かれている。つまり、両者が別々の宅配場所へ同時に宅配する経路が生成されているといえる。

ここで、表 6.3 より、Hybrid 宅配経路と Truck 宅配経路を  $F_1(\mathbf{x})$ 、 $F_2(\mathbf{x})$  及び  $C_{total}(\mathbf{x})$  について比較すると、

$$F_1(\mathbf{x}) : \left(1 - \frac{52.9 [\text{km}]}{81.2 [\text{km}]}\right) \times 100 = 34.9\%, \quad F_2(\mathbf{x}) : \left(1 - \frac{2.33[\text{h}]}{3.46[\text{h}]}\right) \times 100 = 32.7\%$$

$$C_{total}(\mathbf{x}) : \left(1 - \frac{4,241[\text{円}]}{5,517 [\text{円}]}\right) \times 100 = 23.1\%$$

となり、いずれも Truck 宅配より多く削減できることがわかる。また、Hybrid 宅配経路と Drone 宅配を  $F_4(\mathbf{x})$  及び  $F_5(\mathbf{x})$  について比較すると、

$$F_4(\mathbf{x}) : \left(1 - \frac{76.6 [\text{km}]}{381.2[\text{km}]}\right) \times 100 = 79.9\%, \quad F_5(\mathbf{x}) : \left(1 - \frac{1.92[\text{h}]}{8.01[\text{h}]}\right) \times 100 = 76.0\%$$

となり、Drone 宅配よりも飛行距離、飛行時間ともに大幅に削減できることがわかる。このことから、Hybrid 宅配は、宅配コストの削減効果だけに関しては Drone 宅配より小さいが、各目的関数の値そのものは Truck 宅配や Drone 宅配より小さいといえる。

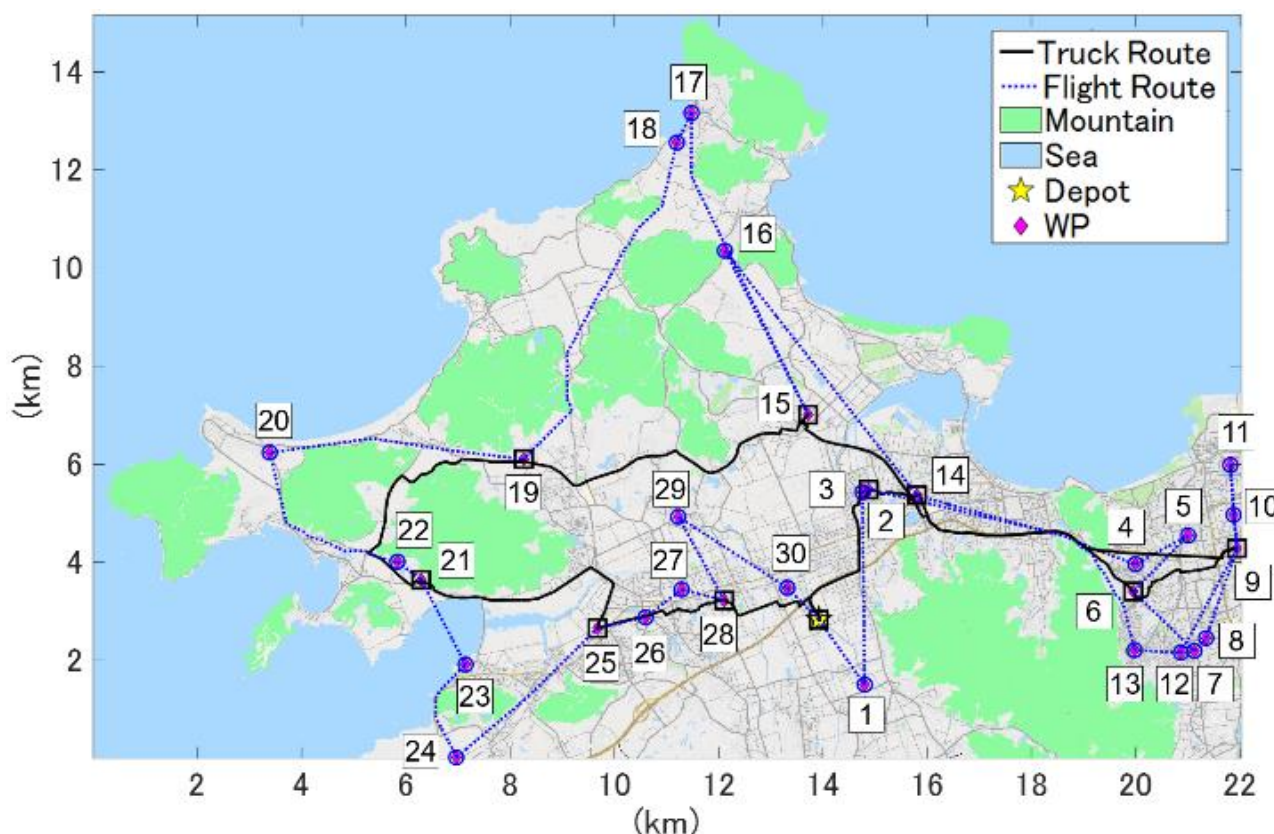


図 6.5 : C-Hybrid 宅配経路

図 6.5 は、改良型 PIP 法を用いて生成した C-Hybrid 宅配経路である。この図からわかるように、Hybrid 宅配経路と同様にトラックとドローンが別々の WP へ同時に宅配していることに加え、ドローンが 2 か所の WP を連続して宅配している様子が確認できる。

ここで、表 6.3 より、C-Hybrid 宅配経路と Truck 宅配経路を  $F_1(\mathbf{x})$ 、 $F_2(\mathbf{x})$  及び  $C_{total}(\mathbf{x})$  について比較すると、

$$F_1(\mathbf{x}) : \left(1 - \frac{45.3 \text{ [km]}}{81.2 \text{ [km]}}\right) \times 100 = 44.2\%, \quad F_2(\mathbf{x}) : \left(1 - \frac{2.14 \text{ [h]}}{3.46 \text{ [h]}}\right) \times 100 = 38.2\%$$

$$C_{total}(\mathbf{x}) : \left(1 - \frac{3,912 \text{ [円]}}{5,517 \text{ [円]}}\right) \times 100 = 29.1\%$$

となり、いずれも Truck 宅配より多く削減できることがわかる。また、これらの数値は前述した Hybrid 宅配経路のコスト削減効果( $F_1(\mathbf{x})$ )に関しては 34.9%、 $F_2(\mathbf{x})$ に関しては 32.7%、 $C_{total}(\mathbf{x})$ に関しては 23.1%よりも大きいことがわかる。この結果から、Drone 宅配に限らず、Hybrid 宅配に関しても連続宅配が可能なドローンを使用する方がより優れたコスト削減効果が得られることが示されたといえる。

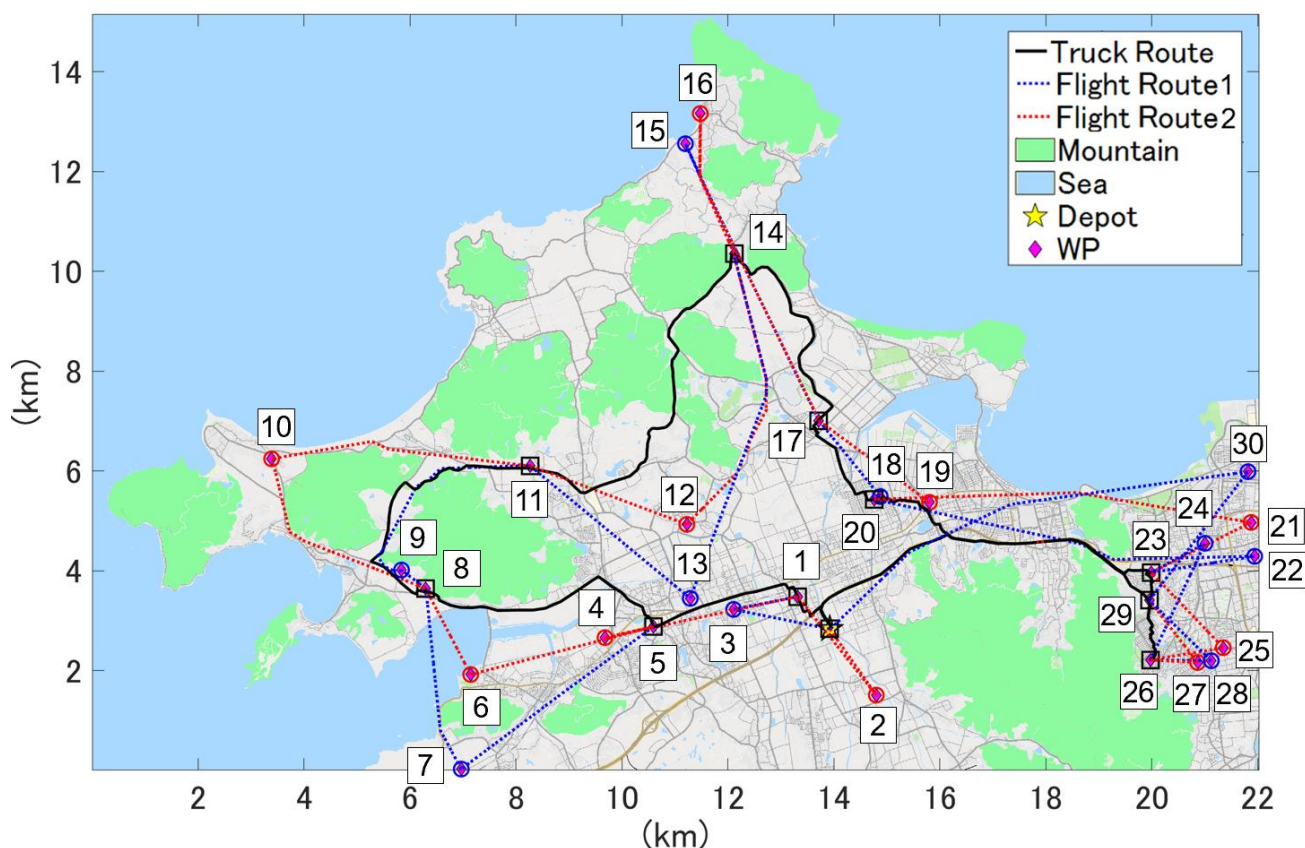


図 6.6 : M-Hybrid 宅配経路

図 6.6 は、改良型 PIP 法を用いて生成した M-Hybrid 宅配経路である。この図からわかるように、Hybrid 宅配経路と同様にトラックとドローンが別々の WP へ同時に宅配していることに加え、2 台のドローンを同時に使用して別々の WP に宅配している様子が確認できる(青色の破線と赤色の破線でそれぞれの飛行経路を表示)。

ここで、表 6.3 より、M-Hybrid 宅配経路と Truck 宅配経路を  $F_1(\mathbf{x})$ ,  $F_2(\mathbf{x})$  及び  $C_{total}(\mathbf{x})$  について比較すると、

$$F_1(\mathbf{x}) : \left(1 - \frac{47.7 [\text{km}]}{81.2 [\text{km}]}\right) \times 100 = 41.3\%, \quad F_2(\mathbf{x}) : \left(1 - \frac{1.94 [\text{h}]}{3.46 [\text{h}]}\right) \times 100 = 43.9\%$$

$$C_{total}(\mathbf{x}) : \left(1 - \frac{4,012 [\text{円}]}{5,517 [\text{円}]}\right) \times 100 = 27.3\%$$

となり、いずれも Truck 宅配より多く削減できることがわかる。また、これらの数値は前述した Hybrid 宅配経路のコスト削減効果( $F_1(\mathbf{x})$ )に関しては 34.9%,  $F_2(\mathbf{x})$  に関しては 32.7%,  $C_{total}(\mathbf{x})$  に関しては 23.1% よりも大きくなるが、C-Hybrid 宅配のコスト削減効果  $F_1(\mathbf{x})$  に関しては 44.2%,  $F_2(\mathbf{x})$  に関しては 38.2%,  $C_{total}(\mathbf{x})$  に関しては 29.1% と比較すると、 $F_2(\mathbf{x})$  以外に関しては僅かに劣ることがわかった。この結果のみではどちらがより有利となるかが判別できないため、次節以降でより詳細に比較する。

### 6.3 宅配コスト削減効果と計算条件の関係

この節では、図 4.10 で示したように、福岡市西区から糸島市までを対象とした宅配エリアにおけるドローン宅配のコスト削減効果について考察する。

#### 6.3.1 相対コスト比

6.2 節では、トラックに対するドローンの相対コスト比を  $R = 0.1$  と任意に設定し、各種ドローン宅配経路のコスト削減効果を評価した。この  $R$  の値が変化すれば計算結果も当然変化すると考えられる。そこで、 $0 \leq R \leq 1$  の範囲におけるコスト削減効果の変移をグラフ化し、コスト削減効果と  $R$  の関係を考察する。図 6.7 はその計算結果を表す。各プロット点の試行回数は計 10 回とし、各プロット点には I 字でその標準偏差を示した。

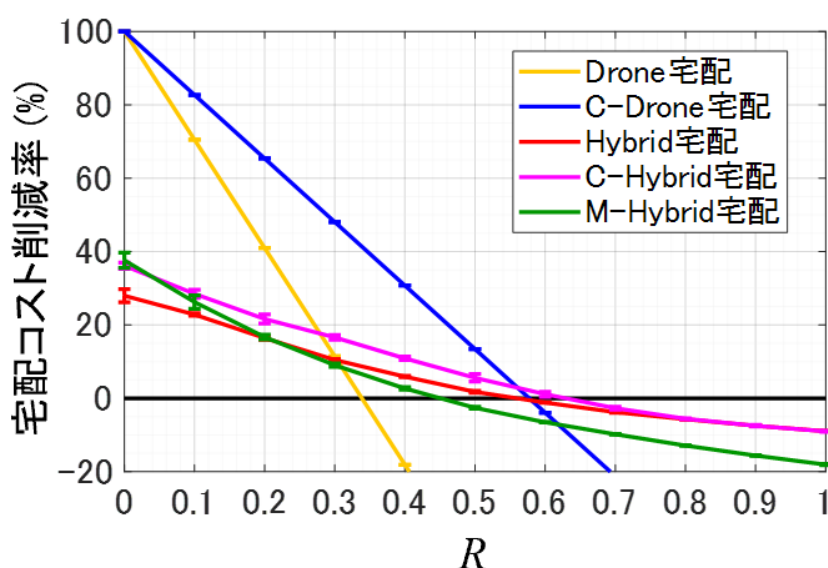


図 6.7：宅配コスト削減率と  $R$  の関係

上図で示したグラフについて考察した結果を以下に列挙する。

- ✓ 黄色の線で示した Drone 宅配と青色の線で示した C-Drone 宅配のコスト削減率は、 $R = 0$  において共に 100% となっている。これは、ドローンに関連する宅配コストが全て 0 に換算されるためである。
- ✓ Drone 宅配では  $0.3 < R < 0.4$ 、C-Drone 宅配では  $0.5 < R < 0.6$  のいずれかの時点でコスト削減効果が 0 となるところがあり、 $R$  がそれ以上に増加すると逆に Truck 宅配より割高になる。これは、 $R$  を一定水準以上低減しなければドローン宅配サービスに期待されるコスト削減効果は期待できないことを意味する。つまり、少なくとも  $R \leq 0.5$  であることがその効果が見込める判断基準になる。
- ✓  $R$  の増加に伴って Drone 宅配及び C-Drone 宅配のグラフは線形に減少している。これは、 $R$  の値が変化しても目的関数値が変化しないことに起因する。なぜなら、Drone 宅配は訪問する順序が変化しても各宅配先とデポの往復飛行の順序が変化するだけだから

である。C-Drone 宅配に関しては、連続して訪問する宅配場所の組み合わせが変化すれば目的関数値は変化するが、今回の問題設定下では全て同じ飛行経路が生成されたために直線状のグラフになったと考えられる。

- ✓ Drone 宅配より C-Drone 宅配の方が  $R$  の増加に伴って減少していく宅配コスト削減率の勾配が小さい。これは、連続宅配が  $R$  の値にかかわらずコスト削減率を向上させる効果があることを示唆している。
- ✓ 赤線で示した Hybrid 宅配、マゼンタ色の線で示した C-Hybrid 宅配、そして緑色で示した M-Hybrid 宅配のコスト削減率は、Drone 宅配や C-Drone 宅配の場合と異なり、 $R = 0$  であったとしてもそのコスト削減率は約 30%弱～40%弱しか得られていない。これは、Hybrid 宅配経路関連にはトラックの合計走行時間  $F_2(\mathbf{x})$ 、すなわち人件費に該当する宅配コストが含まれているが、Drone 宅配関連にはそれが無いことに起因する。
- ✓ Hybrid 宅配と C-Hybrid 宅配を比較すると、コスト削減率が 0 になるまで一貫して C-Hybrid 宅配の方が優れていることがわかる。つまり、Drone 宅配の場合と同じく、Hybrid 宅配においてもドローンによる連続宅配が有用であることが示された。
- ✓ C-Hybrid 宅配と M-Hybrid 宅配を比較すると、 $R = 0$  では同水準だった両者のコスト削減率が  $R$  の増加に伴って M-Hybrid 宅配の方がより早く減少することが確認できる。また、 $R = 0.2$  では、Hybrid 宅配と M-Hybrid 宅配のコスト削減率がほぼ同水準となり、 $0.4 < R < 0.5$  の範囲で 3 つの Hybrid 宅配関連では最も早い段階で M-Hybrid 宅配のコスト削減効果が 0 になる。このように推移する理由として、 $R$  の値が大きくなる程 Hybrid 宅配や C-Hybrid 宅配より 1 台多くドローンを使用する M-Hybrid 宅配の方が結果的により多くの宅配コストが必要になるためだと考えられる。したがって、Hybrid 宅配に関しては複数のドローンを 1 台のトラックで同時に運用することは必ずしも有効とはいえない。
- ✓ Drone 宅配と Hybrid 宅配を比較すると、 $R$  が 0 に近いほど Drone 宅配の方がコスト削減効果において有利であるが、 $R = 0.3$  の時点で両者はほぼ同水準となり、それ以上は Hybrid 宅配の方が有利になるという逆転現象が発生している。このことから、Drone 宅配と Hybrid 宅配のどちらのスタイルが有利であるかは  $R$  の値次第といえる。

以上より、ドローン宅配サービスのコスト削減効果について以下の知見が得られた。

- Hybrid 宅配に関しては、複数のドローンを同時に運用するより、複数の宅配場所を連続して宅配可能なドローンを使用する方が有利である。
- $R$  の値が 0 に近いほど Hybrid 宅配より Drone 宅配の方がより大きなコスト削減効果を期待できるが、逆に  $R$  の値が大きくなるほど、今度は Hybrid 宅配の方が有利となり、 $R \geq 0.5$  でいずれのコスト削減効果も失われると考えられる。
- ドローン宅配サービスのコスト削減効果は、 $R$  の値を低減するほど増大する。

### 6.3.2 移動速度

6.2 節では、 $R = 0.1$ におけるトラックの走行速度が $V_T = 10[\text{m/s}]$ に対して、ドローンの飛行速度を $V_D = 15[\text{m/s}]$ に設定した。この $V_D$ の値が変化すれば、そのコスト削減効果も変化することが予想される。そこで、 $V_D$ の値を $5[\text{m/s}] \leq V_D \leq 30[\text{m/s}]$ の範囲で変化させ、その際のコスト削減率をグラフ化することで、トラックに対するドローンの相対速度とコスト削減効果の関係を考察する。このグラフを図 6.8 に示す。各プロット点の試行回数は計 10 回である。なお、Drone 宅配と H-Drone 宅配のグラフの一部が欠損している理由は、その飛行速度ではデポの遠方にある宅配場所を往復する飛行時間が足りなかったためである。

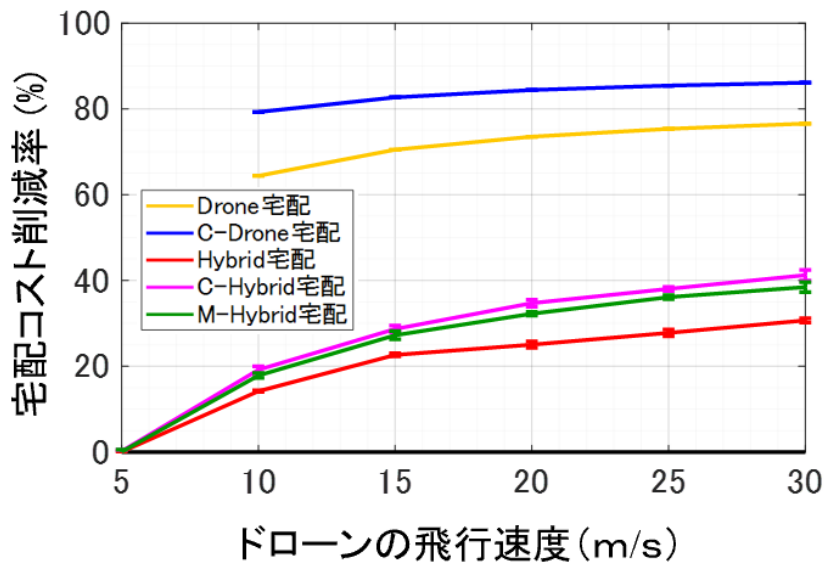


図 6.8：コスト削減率とドローンの飛行速度の関係

上図で示したグラフについて考察した結果を以下に示す。

- ✓ グラフ全体としては、 $V_T$ に対して $V_D$ が大きくなるほど宅配コスト削減率は向上することが確認できる。ただし、向上率の勾配は $V_D$ の相対速度が大きくなるほど減少していく。
- ✓ 黄色の線で示した Drone 宅配と赤色の線で示した Hybrid 宅配を比較すると、宅配コスト削減率そのものは Drone 宅配の方が有利であるが、Hybrid 宅配は $V_D = 5[\text{m/s}]$ であっても実行可能解が得られることや、 $V_D$ の相対速度向上に伴う宅配コスト削減効果の向上が Drone 宅配より大きいという特徴を持つ。この傾向は、 $V_T$ に対する $V_D$ の値が小さいときほど顕著である。

以上の解析結果から、ドローン宅配サービスでは $V_T$ に対する $V_D$ の相対速度が大きくなるほど宅配コストの削減効果も向上するが、向上率そのものは徐々に鈍化していくことがわかった。鈍化する理由は、トラックとドローンの離合を繰り返す Hybrid 宅配では、一方の移動速度が向上しても、もう一方の移動速度が付随して速くならない限り、結果的に両者が合流する際の待機時間が増加するだけだからと考えられる。

### 6.3.3 離着陸時間

6.2 節では、 $R = 0.1$ におけるドローンの離着陸時間、すなわち地上から安全な高度まで上昇する時間と、目標地点上空から接地するまでに要する時間をそれぞれ $t_{tl} = 30[s]$ に設定して計算した。この $t_{tl}$ の値が変化すれば、それによって得られる宅配コストの削減効果も変化することが予想される。そこで、 $t_{tl}$ の値を $0[s] \leq t_{tl} \leq 180[s]$ の範囲で変化させ、この際のコスト削減率の変移をグラフ化することで、ドローンの離着陸時間とコスト削減効果の関係を明らかにする。このグラフを図 6.9 に示す。各プロット点の試行回数は計 10 回である。

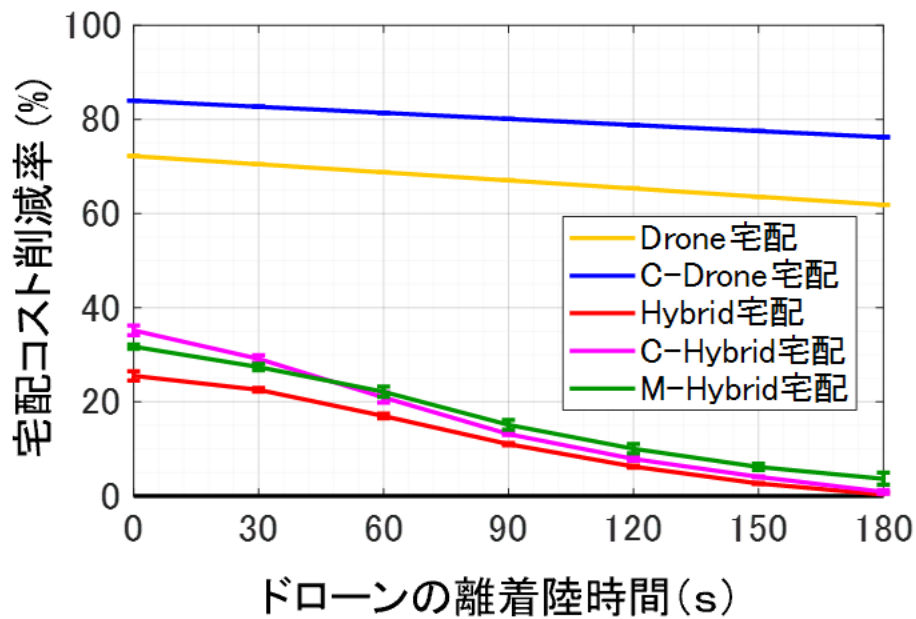


図 6.9：コスト削減率とドローンの離着陸時間の関係

上図に示されたグラフ全体から、 $t_{tl}$ の値が大きくなるほど全ての宅配スタイルにおいてそのコスト削減率が減少していく様子を確認できる。特に、赤色の線で示した Hybrid 宅配、マゼンタ色の線で示した C-Hybrid 宅配及び緑色の線で示した M-Hybrid 宅配に関しては、黄色の線で示した Drone 宅配と青色の線で示した C-Drone 宅配のグラフよりも減少する勾配が大きいことがわかる。この理由として、2つの Drone 宅配スタイルは、 $t_{tl}$ の値がどんなに大きく変化しても生成される経路そのものはほとんど変化せず、 $t_{tl}$ の値が増加した分だけ飛行時間が長くなるだけであるが、3つの Hybrid 宅配スタイルに関しては、ドライバーがドローンを回収する際の待ち時間が $t_{tl}$ の増加にともなって長くなるので、その分だけコスト削減効果が損なわれるからである。よって、ドローンを目標地点上空から所定の位置まで誘導する時間や、地上から安全な空域まで上昇する時間が長くなるほど、Hybrid 宅配よりも Drone 宅配の方がコスト削減率の観点から有利になる可能性がある。

### 6.3.4 飛行可能時間

この節では、 $R = 0.1$ におけるドローンの飛行可能時間 $t_d$ と宅配コスト削減率の関係を明らかにする。具体的には、 $t_d$ の値を $10[\text{min}] \leq t_d \leq 40[\text{min}]$ の範囲で $5[\text{min}]$ ずつ変化させた際のコスト削減率の変移をグラフ化し、その計算結果について考察する。図 6.10 にその計算結果を示す。各プロット点の試行回数は計 10 回である。なお、Drone 宅配と C-Drone 宅配のグラフが $10[\text{min}] \leq t_d \leq 25[\text{min}]$ 間で欠損している理由は、この計算条件下ではデポから遠方に位置する宅配場所まで直接ドローンで宅配するために最低でも $t_d$ が $30[\text{min}]$ 以上必要だったためである。なお、 $t_d$ 以外の計算条件は 6.2 節と同様とする。

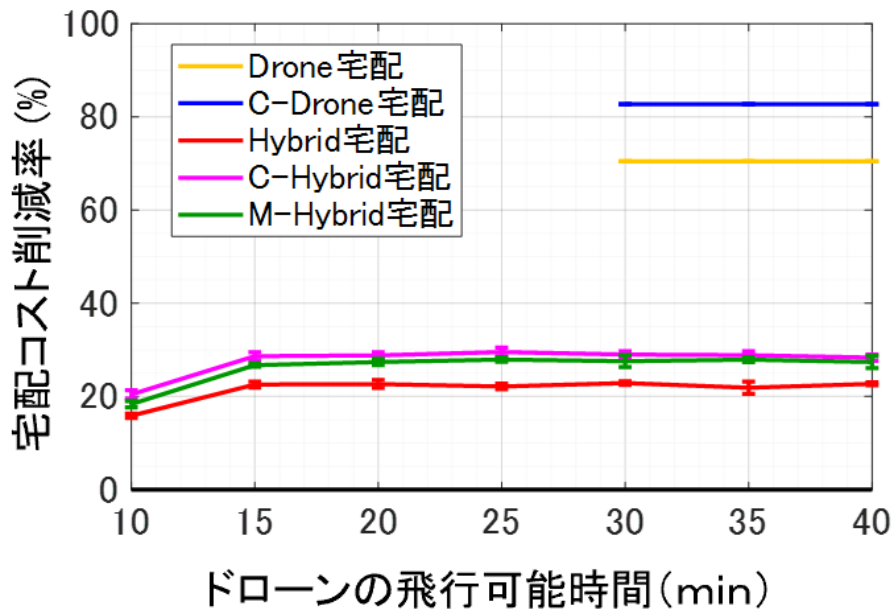


図 6.10：コスト削減率とドローンの飛行可能時間の関係

上図で示したグラフの考察した結果を以下に示す。

- ✓ 黄色の線で示した Drone 宅配及び青色の線で示した C-Drone 宅配については、 $t_d$ の値が増加しても生成される宅配経路に変化が生じないため、その宅配コスト削減率は一定値となっている。ただし、 $t_d$ が小さくなると、今度はデポと宅配場所間を往復するために必要な飛行時間を満足することができなくなるため、上図のように実行可能解が得られない場合がある。よって、デポと宅配場所間を往復する最低限の飛行時間は確保すべきであるが、それ以上の飛行時間は不要と判断できる。
- ✓ 赤線で示した Hybrid 宅配、マゼンタ色で示した C-Hybrid 宅配及び緑色の線で示した M-Hybrid 宅配の宅配コスト削減率は、 $t_d$ の値にかかわらず、常に C-Hybrid 宅配、M-Hybrid 宅配及び Hybrid 宅配の順番で優れており、 $t_d = 15[\text{min}]$ 以降については、3 者とも一定値で推移していることがわかる。つまり、 $t_d$ の値が一定値を超えると、それ以上  $t_d$ を大きくしても効果はないことを示唆している。

## 6.4 宅配経路とタスク割当の同時最適化

第4章で示した Multi-Agent 宅配と C- Multi-Agent 宅配は、宅配経路とタスク割当の同時最適化問題である。この節では、 $R$ の値を  $0 \leq R \leq 1$  の範囲で変化させた場合における Multi-Agent 宅配と C-Multi-Agent 宅配のコスト削減率をグラフ化し、 $R$ と連続宅配のコスト削減効果の関係について考察する。表 6.4 及び表 6.5 に計算条件を示す。6.2 節における計算条件の違いとして、宅配場所の数を 30 か所から 120 か所、使用できるトラックとドローンの数をそれぞれ 4 台に増加させていることと、宅配可能な時間を 3 時間に設定していることである。最も大きな違いは、生成される宅配経路のスタイルを問わない点である。つまり、タスク分散により複数の経路が生成された場合、それらの経路はどの宅配スタイルであってもよいということである。ただし、連続宅配に関しては C-Multi-Agent 宅配のみ可能とする。また、前節までにドローンの積載可能重量を  $W_d = 2[kg]$  と設定していたが、C-Multi-Agent 宅配に関しては  $W_d = 6[kg]$  に設定する。これにより、連続してドローンが宅配できる  $WP$  の数を最大 3 か所にまで拡張させる。トラックに搭載できるドローンの数は全て 1 台のみに限定する。なお、問題の規模が拡大していることや複雑さが増していることを踏まえ、局所的最適解への収束と判定する世代数  $G_{tabu}$  及び解探索の最大世代数  $G_{max}$  の設定値を 6.2 節の 10 倍とする。各プロット点の試行回数は計 5 回とする。図 6.11 はその計算結果を示す。

表 6.4：共通の計算条件

項目	値	単位	項目	値	単位
$P_n$	100	[個]	$t_d$	30	[min]
$E_n$	1	[個]	$t_{tl}$	30	[s]
$G_{tabu}$	50	[世代]	$t_{re}$	30	[s]
$G_{max}$	50000	[世代]	$t_p$	2	[min]
$N_{WP}$	120	[地点]	$w_b^j, w_{a,b}^{ij}$	2	[kg]
$V_t$	10	[m/s]	$LD_{max}$	100	[km]
$V_d$	15	[m/s]	$LT_{max}$	3	[h]
$A$	4	[台]			

表 6.5：宅配スタイルごとの計算条件

項目	宅配スタイルと値			単位
	Truck	Multi-Agent	C- Multi-Agent	
$B$	0	4	4	[台]
$W_d$	0	2	6	[kg]

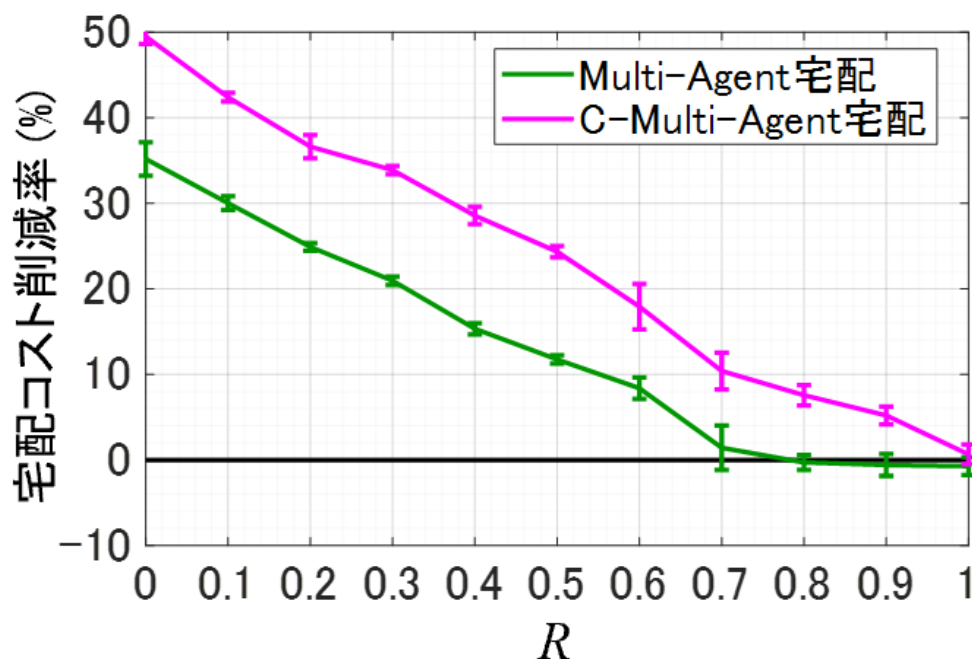


図 6.11：宅配コスト削減率とRの関係

上図のグラフより，連続宅配が可能な C-Multi-Agent 宅配が一貫して Multi-Agent 宅配より優れた宅配コスト削減率を有していることが示された．宅配コスト削減率とは，これまでと同じく Truck 宅配に対してどれだけ  $C_{total}(x)$  を削減できたかを意味する．ここで，120 か所ある WP のうち，ドローンが宅配する割合が R の値に応じてどのように変化するのかを表したグラフを図 6.12 に示す．

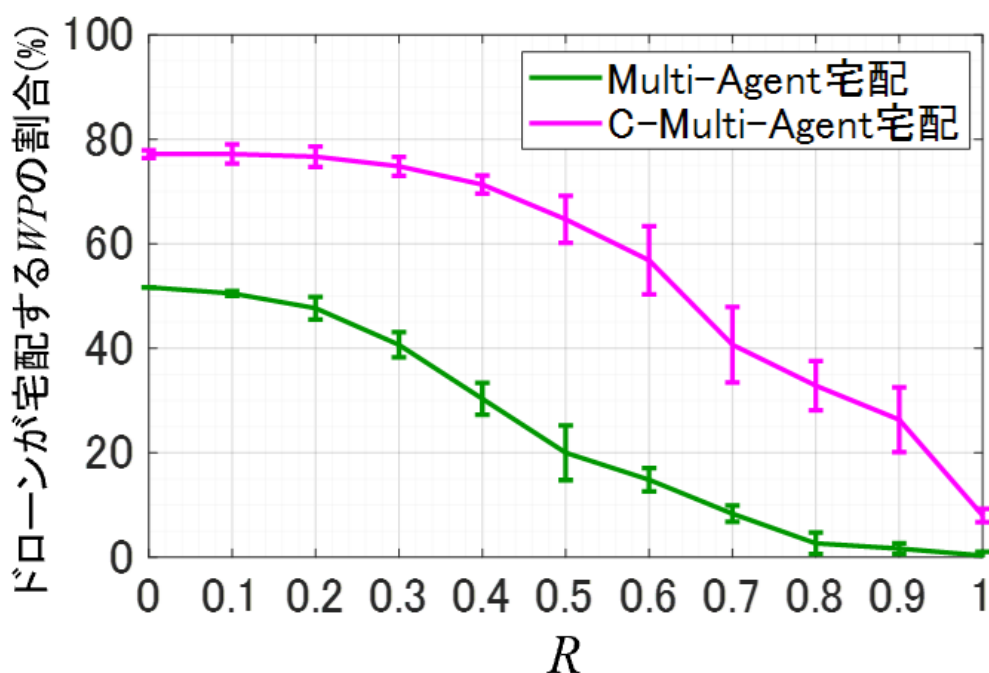


図 6.12：ドローンが宅配する WP の割合と R の関係

図 6.12 で示した 2 つのグラフはいずれも、 $R$ の値が増加する程ドローンが宅配する  $WP$  の割合が減少している。つまり、 $R$ の値が小さい場合には積極的にドローンを活用する経路が生成され、そうでない場合にはトラックが主に宅配する経路が生成されることを意味する。また、緑色の線で示された Multi-Agent 宅配とマゼンタ色で示された C-Multi-Agent 宅配を比較すると、宅配コスト削減効果が大きいのは一貫して後者であることがわかる。したがって、Multi-Agent 宅配に関してもドローンによる連続宅配が有用であることが示されたといえる。

## 6.5 まとめ

- トラックに対するドローンの相対コストを可能な限り最小化することと、ドローンが複数の宅配場所を連続して宅配できるように性能を向上させることは、宅配スタイルにかかわらずドローン宅配サービスのコスト削減効果を向上させる上で有効である。
- トラックに対するドローンの相対コストが半分以上に増加すると、ドローン宅配サービスによるコスト削減効果はどの宅配スタイルを選択してもほぼ望めなくなる。
- Truck 宅配と Drone 宅配では、宅配コスト面だけでは明らかに後者の方が有利であるが、その実現可能性や作業効率、安全性という面を含めれば、一概にその優劣を判断することはできない。
- Drone 宅配と Hybrid 宅配のどちらがコスト削減効果の面で有利になるかは、 $R$ の値に依存するため、一概にその優劣を判断することはできない。
- Hybrid 宅配は、ドローンの飛行可能時間が短時間であったとしても、トラックと組み合わせることによって実行可能解を生成できる可能性があるが、Drone 宅配についてはそのような柔軟性がない。
- Hybrid 宅配は、宅配作業中に何度もトラックとドローンが合流を繰り返すため、その際の一方向の待ち時間が増加すると作業効率が損なわれる恐れがある。一方の Drone 宅配には待ち時間は生じないので、その懸念がない利点がある。
- ドローンの飛行速度と飛行可能時間は一定値を超えるとほとんどコスト削減効果の向上に寄与しない。
- ドローンの離着陸時間が長くなると、Drone 宅配よりもコスト削減効果を大きく損なう恐れがある。
- 連続宅配によるコスト削減効果の向上は Multi-Agent 宅配経路にも同様に期待できる。

## 第7章 結論

### 7.1 まとめ

本研究では、近年世界中で活発な研究開発が行われているドローン宅配サービスの経路生成問題を制約付き多目的最適化問題として定式化するとともに、その解法として遺伝的アルゴリズムをベースとした「暫定理想点法」という新しい多目的最適化手法を提案した。また、本手法を用いてドローン宅配サービスが従来のサービスと比較することで、どの程度のコスト削減効果が期待できるのかを定量的に評価する試みを行った。

ドローン宅配サービスは、宅配需要の増加と慢性的な宅配ドライバー不足に悩まされる我が国の物流業界を支える有望な手段とし注目されているが、技術的なハードルに加え、法的問題、騒音やプライバシーの保護等に対する国民感情など、未だ数多くの解決すべき諸課題が残されているが、この研究ではその実現を見据えた運用の在り方を中心に検討した。

暫定理想点法の解探索プロセスは2段階に分かれている。第1段階では、与えられた全ての制約条件を満足する実行可能解を生成するため、制約条件から逸脱した偏差量の総和が0となる解を探索する。実行可能解の生成に成功すれば、第2段階である多目的最適化に移行する。このプロセスでは、全ての目的関数を最適化可能な理想的な解を意味する暫定理想点と、実行可能解の座標点を意味する解点とのユークリッド距離を定義し、これを最小化することで選好解を生成する。本手法の有用性、妥当性を検証するため、制約付き単目的最適化問題であるGO問題と制約なし多目的最適化問題であるDTLZ問題と呼ばれる2種類のベンチマーク問題を用いて解探索を行った。その結果、適用した全てのベンチマーク問題において本手法の有効性が概ね確認できた。

次に、トラックのみを用いた従来の宅配サービスに対して、デポからドローンを使って直接宅配場所まで荷物を届ける宅配スタイルや、トラックとドローンを組み合わせた宅配スタイル、ドローンが複数の宅配場所へ連続して荷物を届ける宅配スタイル等に関する解表現方法と遺伝的操作方法を提案するとともに、これらの経路生成問題を制約付き多目的最適化問題として定式化した。

上記の宅配経路問題に対して当初提案した暫定理想点法を適用すると、宅配コストに関する実情報が評価関数に反映されていないために解探索途中でその評価値が無限に発散する可能性があることがわかった。そこで、変換係数を用いてこの宅配経路問題を宅配コストに

関する単目的最適化問題として解く手法を提案した。一方、独自に提案した解表現の構造は複雑であるが故に遺伝的アルゴリズムによる交叉交配演算子を適用できず、突然変異のみに依存した探索効率の悪い計算を強いられるという別の問題点も存在することがわかった。そこで、暫定理想点法の解探索にタブーサーチを組み合わせることを提案した。本手法の有用性を示すため、ベンチマーク問題である TSPLIB に突然変異のみを用いた遺伝的アルゴリズムと、これにタブーサーチを組み合わせた手法をそれぞれ適用し、既知の最適値と生成解との誤差率及び標準偏差を比較した。その結果、後者の方が誤差率及び標準偏差のいずれの点においても優れた解探索が実現していることを確認した。また、暫定理想点法についてもタブーサーチを組み合わせた解探索の方が改良前の手法より宅配経路問題において実行可能解の生成確率が大幅に改善されることや、生成された選好解の宅配コスト及び標準偏差が小さくなることを確認した。

最後に、福岡市西区から糸島市内を宅配エリアとする宅配経路問題に改良した暫定理想点法を適用し、従来のトラックのみを用いた宅配経路に対するドローン宅配のコスト削減効果を定量的に評価した。その結果、トラックに対するドローンの相対コスト比をできるだけ小さくすることや、複数か所を連続して宅配可能なドローンを使用することがドローン宅配のコスト削減効果を向上させる上で有効になり得る可能性があることを示した。また、この相対コスト比の値次第で宅配コストの面で有利になる宅配スタイルが変化することや、ドローンの飛行速度及び飛行可能時間、離着陸時間といったドローンに関する性能諸元をそれぞれ単独で向上しても宅配コスト削減に必ずしも寄与しない場合があることなど、ドローン宅配に関する様々な知見を得ることができた。

## 7.2 今後の課題

本研究で得られたドローン宅配経路は、より現実に近い環境と宅配コストに関する実情報を用いているため、ある程度その計算結果には妥当性があると考えられる。しかしながら、得られた知見はあくまで特定の問題設定と不確実性を考慮しない理想的な環境下を前提としたものであることから、他の問題設定においても一般的に通用するものではないことは明白である。

上記の課題を踏まえた今後の課題としては、仮想的な宅配エリアを自由に設計できる計算環境を用意し、様々な問題設定下で多角的な視点からドローン宅配によるコスト削減効果を評価し、より普遍的な知見を得られることが求められる。また、交通渋滞や再配達といった実際に想定される不確実性を考慮した経路生成手法を確立し、動的な環境下におけるドローン宅配の便益評価についても検討する必要がある。

以上のように、様々な問題設定に対してドローン宅配サービスの潜在的な便益を幅広く検証することが、その運用の在り方を検討する上で有益な知見を提供すると考える。

## Appendix A

この Appendix では、制約付き単目的最適化問題のベンチマーク問題である GO 問題のうち、PIP 法の妥当性評価のために使用した go01 から go10 までの計 10 種類の定義式とその最適値及び設計変数についてそれぞれ示す。

(1) go1

go01 の定義式を式(A.1)に示す。

$$\left( \begin{array}{l}
 F(\mathbf{x}) = 5 \sum_{i=1}^4 x_i - 5 \sum_{i=1}^4 x_i^2 - \sum_{i=5}^{13} x_i \\
 g_1(\mathbf{x}) = 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \leq 0 \\
 g_2(\mathbf{x}) = 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \leq 0 \\
 g_3(\mathbf{x}) = 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \leq 0 \\
 g_4(\mathbf{x}) = -8x_1 + x_{10} \leq 0 \\
 g_5(\mathbf{x}) = -8x_2 + x_{11} \leq 0 \\
 g_6(\mathbf{x}) = -8x_3 + x_{12} \leq 0 \\
 g_7(\mathbf{x}) = -2x_4 - x_5 + x_{10} \leq 0 \\
 g_8(\mathbf{x}) = -2x_6 - x_7 + x_{11} \leq 0 \\
 g_9(\mathbf{x}) = -2x_8 - x_9 + x_{12} \leq 0 \\
 0 \leq x_i \leq 1 (i = 1, \dots, 9) \\
 0 \leq x_i \leq 100 (i = 10, 11, 12) \\
 0 \leq x_{13} \leq 1
 \end{array} \right. \quad (\text{A.1})$$

上式より、go01 問題では目的関数  $F(\mathbf{x})$  に対して制約条件  $g_j(\mathbf{x})$  が計 9 個、設計変数  $x_i$  が計 13 個ずつ与えられていることがわかる。この問題の最適値は次のように与えられている。

$$F(\mathbf{x}_{opt}) = -15.0$$

(2) go2

go02 の定義式を式(A.2)に示す。

$$\left( \begin{array}{l} F(\mathbf{x}) = \left| \frac{\sum_{i=1}^n \cos^4(x_i) - 2 \prod_{i=1}^n \cos^2(x_i)}{\sqrt{\sum_{i=1}^n i x_i^2}} \right| \\ g_1(\mathbf{x}) = 0.75 - \prod_{i=1}^n x_i \leq 0 \\ g_2(\mathbf{x}) = \sum_{i=1}^n x_i - 7.5n \leq 0 \\ 0 \leq x_i \leq 10 (i = 1, \dots, 20) \end{array} \right. \quad (\text{A.2})$$

上式より, go02 問題では目的関数 $F(\mathbf{x})$ に対して制約条件 $g_j(\mathbf{x})$ が計 2 個, 設計変数 $x_i$ が計 20 個ずつ与えられていることがわかる. この問題の最適値は次のように与えられる.

$$F(\mathbf{x}_{opt}) = 0.8036$$

ただし, この時の設計変数に関しては引用文献内に示されていないため, 省略する.

### (3)go3

go03 の定義式を式(A.3)に示す.

$$\left( \begin{array}{l} F(\mathbf{x}) = (\sqrt{n})^n \prod_{i=1}^n x_i \\ g_1(\mathbf{x}) = \sum_{i=1}^n x_i^2 - 1 = 0 \\ 0 \leq x_i \leq 1 (i = 1, \dots, 10) \\ 1 \leq i \leq n \end{array} \right. \quad (\text{A.3})$$

上式より, go03 問題では目的関数 $F(\mathbf{x})$ に対して制約条件 $g_j(\mathbf{x})$ が 1 個, 設計変数 $x_i$ が計 10 個ずつ与えられていることがわかる. この問題の最適値は次のように与えられている.

$$F(\mathbf{x}_{opt}) = 1$$

### (4)go4

go04 の定義式を式(A.4)に示す.

$$\begin{cases}
 F(\mathbf{x}) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40792.141 \\
 g_1(\mathbf{x}) = 85.334407 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 - 92 \leq 0 \\
 g_2(\mathbf{x}) = -85.334407 - 0.0056858x_2x_5 - 0.0006262x_1x_4 + 0.0022053x_3x_5 \leq 0 \\
 g_3(\mathbf{x}) = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_3^2 - 110 \leq 0 \\
 g_4(\mathbf{x}) = -80.51249 - 0.0071317x_2x_5 - 0.0029955x_1x_2 - 0.0021813x_3^2 + 90 \leq 0 \\
 g_5(\mathbf{x}) = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4 - 25 \leq 0 \\
 g_6(\mathbf{x}) = -9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 + 20 \leq 0 \\
 78 \leq x_1 \leq 102 \\
 33 \leq x_2 \leq 45 \\
 27 \leq x_i \leq 45 \quad (i = 3, 4, 5)
 \end{cases} \quad (\text{A.4})$$

上式より, go04 問題では目的関数 $F(\mathbf{x})$ に対して制約条件 $g_j(\mathbf{x})$ が計 6 個, 設計変数 $x_i$ が計 5 個ずつ与えられていることがわかる. この問題の最適値は次のように与えられている.

$$F(\mathbf{x}_{opt}) = -30665.539$$

(5) go5

go05 の定義式を式(A.5)に示す.

$$\begin{cases}
 F(\mathbf{x}) = 3x_1 + 0.000001x_1^2 + 2x_2 + (0.000002/3)x_2^3 \\
 g_1(\mathbf{x}) = -4x_4 + x_3 - 0.55 \leq 0 \\
 g_2(\mathbf{x}) = -x_3 + x_4 - 0.55 \leq 0 \\
 g_3(\mathbf{x}) = 1000 \sin(-x_3 - 0.25) + 1000 \sin(-x_4 - 0.25) + 894.8 - x_1 = 0 \\
 g_4(\mathbf{x}) = 1000 \sin(x_3 - 0.25) + 1000 \sin(x_3 - x_4 - 0.25) + 894.8 - x_2 = 0 \\
 g_5(\mathbf{x}) = 1000 \sin(x_4 - 0.25) + 1000 \sin(x_4 - x_3 - 0.25) + 1294.8 = 0 \\
 0 \leq x_1 \leq 1200 \\
 0 \leq x_2 \leq 1200 \\
 -0.55 \leq x_3 \leq 0.55 \\
 -0.55 \leq x_4 \leq 0.55
 \end{cases} \quad (\text{A.5})$$

上式より, go05 問題では目的関数 $F(\mathbf{x})$ に対して制約条件 $g_j(\mathbf{x})$ が計 5 個, 設計変数 $x_i$ が計 4 個ずつ与えられていることがわかる. この問題の最適値は次のように与えられている.

$$F(\mathbf{x}_{opt}) = 5126.4981$$

(6) go6

go06 の定義式を式(A.6)に示す.

$$\begin{cases}
 F(\mathbf{x}) = (x_1 - 10)^3 + (x_2 - 20)^3 \\
 g_1(\mathbf{x}) = -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0 \\
 g_2(\mathbf{x}) = (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \leq 0 \\
 13 \leq x_1 \leq 100 \\
 0 \leq x_2 \leq 100
 \end{cases} \quad (\text{A.6})$$

上式より, go06 問題では目的関数 $F(\mathbf{x})$ に対して制約条件 $g_j(\mathbf{x})$ が計 2 個, 設計変数 $x_i$ が計 2 個ずつ与えられていることがわかる. この問題の最適値は次のように与えられている.

$$F(\mathbf{x}_{opt}) = -6961.8139$$

(7)go7

go07 の定義式を式(A.7)に示す.

$$\left( \begin{array}{l} F(\mathbf{x}) = x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + \\ (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45 \\ g_1(\mathbf{x}) = -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \leq 0 \\ g_2(\mathbf{x}) = 10x_1 - 8x_2 - 17x_7 + 2x_8 \leq 0 \\ g_3(\mathbf{x}) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \leq 0 \\ g_4(\mathbf{x}) = 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 \leq 0 \\ g_5(\mathbf{x}) = 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 \leq 0 \\ g_6(\mathbf{x}) = x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \leq 0 \\ g_7(\mathbf{x}) = 0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30 \leq 0 \\ g_8(\mathbf{x}) = -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \leq 0 \\ -10 \leq x_i \leq 10 \quad (i = 1, \dots, 10) \end{array} \right. \quad (\text{A.7})$$

上式より, go07 問題では目的関数 $F(\mathbf{x})$ に対して制約条件 $g_j(\mathbf{x})$ が計 8 個, 設計変数 $x_i$ が計 10 個ずつ与えられていることがわかる. この問題の最適値は次のように与えられている.

$$F(\mathbf{x}_{opt}) = 24.3062$$

(8)go8

go08 の定義式を式(A.8)に示す.

$$\left( \begin{array}{l} F(\mathbf{x}) = \frac{\sin^3(2\pi x_1)\sin(2\pi x_2)}{x_1^3(x_1 + x_2)} \\ g_1(\mathbf{x}) = x_1^2 - x_2 + 1 \leq 0 \\ g_2(\mathbf{x}) = 1 - x_1 + (x_2 - 4)^2 \leq 0 \\ 0 \leq x_1 \leq 10 \\ 0 \leq x_2 \leq 10 \end{array} \right. \quad (\text{A.8})$$

上式より, go08 問題では目的関数 $F(\mathbf{x})$ に対して制約条件 $g_j(\mathbf{x})$ が計 2 個, 設計変数 $x_i$ が計 2 個ずつ与えられていることがわかる. この問題の最適値は次のように与えられている.

$$F(\mathbf{x}_{opt}) = 0.095825$$

(9)go9

go09 の定義式を式(A.9)に示す.

$$\begin{cases}
 F(\mathbf{x}) = (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7 \\
 g_1(\mathbf{x}) = -127 + 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \leq 0 \\
 g_2(\mathbf{x}) = -282 + 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \leq 0 \\
 g_3(\mathbf{x}) = -196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leq 0 \\
 g_4(\mathbf{x}) = 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \leq 0 \\
 -10 \leq x_i \leq 10 \quad (i = 1, \dots, 7)
 \end{cases} \quad (\text{A.9})$$

上式より, go09 問題では目的関数 $F(\mathbf{x})$ に対して制約条件 $g_j(\mathbf{x})$ が計 4 個, 設計変数 $x_i$ が計 7 個ずつ与えられていることがわかる. この問題の最適値は次のように与えられている.

$$F(\mathbf{x}_{opt}) = 680.630$$

(1 0)go10

go10 の定義式を式(A.10)に示す.

$$\begin{cases}
 F(\mathbf{x}) = x_1 + x_2 + x_3 \\
 g_1(\mathbf{x}) = -1 + 0.0025(x_4 + x_6) \leq 0 \\
 g_2(\mathbf{x}) = -1 + 0.0025(x_5 + x_7 - x_4) \leq 0 \\
 g_3(\mathbf{x}) = -1 + 0.01(x_8 - x_5) \leq 0 \\
 g_4(\mathbf{x}) = -x_1x_6 + 833.33252x_4 + 100x_1 - 83333.333 \leq 0 \\
 g_5(\mathbf{x}) = -x_2x_7 + 1250x_5 + x_2x_4 - 1250x_4 \leq 0 \\
 g_6(\mathbf{x}) = -x_3x_8 + 1250000 + x_3x_5 - 2500x_5 \leq 0 \\
 100 \leq x_1 \leq 10000 \\
 1000 \leq x_i \leq 10000 \quad (i = 2, 3) \\
 10 \leq x_i \leq 1000 \quad (i = 4, \dots, 8) \\
 -10 \leq x_i \leq 10 \quad (i = 1, \dots, 10)
 \end{cases} \quad (\text{A.10})$$

上式より, go10 問題では目的関数 $F(\mathbf{x})$ に対して制約条件 $g_j(\mathbf{x})$ が計 6 個, 設計変数 $x_i$ が計 10 個ずつ与えられていることがわかる. この問題の最適値は次のように与えられている.

$$F(\mathbf{x}_{opt}) = 7049.3307$$





## (3) DTLZ7

DTLZ7 の定義式を式(B.5)に示す.

$$\begin{cases} \min F_1(x_1) = x_1, \\ \min F_2(x_2) = x_2, \\ \vdots \\ \min F_{M-1}(x_{M-1}) = x_{M-1}, \\ \min F_M(\mathbf{x}) = (1 + g(x_M))h(f_1, f_2, \dots, f_{M-1}, g), \\ \text{subject to } 0 \leq x_i \leq 1 \ (i = 1, 2, \dots, n). \end{cases} \quad (\text{B.5})$$

ここで,  $g(\mathbf{x}_M)$ 及び $|\mathbf{x}_M|$ は式(B.6)及び式(B.7)で示すように与えられる.

$$g(\mathbf{x}_M) = 1 + \frac{9}{|\mathbf{x}_M|} \sum_{x_i \in \mathbf{x}_M} x_i, \quad (\text{B.6})$$

$$h(f_1, f_2, \dots, f_{M-1}, g) = M - \sum_{i=1}^{M-1} \left[ \frac{f_i}{1+g} (1 + \sin(3\pi f_i)) \right] \quad (\text{B.7})$$

DTLZ7 の $g(\mathbf{x}_M)$ は, 上記の DTLZ1 や DTLZ3 の場合と異なり,  $g(\mathbf{x}_M) = 1$ となる解がパレート最適解となる. 図 B.3 は,  $M = 2$ の場合における DTLZ7 のパレートフロンティアの形状を表す.

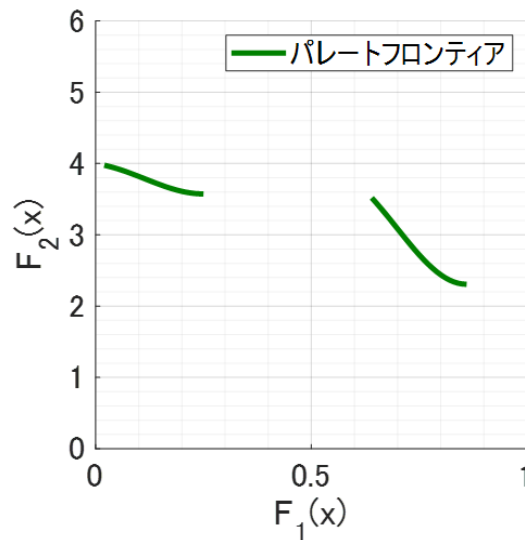


図 B.3 : DTLZ7 のパレートフロンティア形状

## 参考文献

- [1] J. A. Besada, L. Bergesio, I. Campaña, D. Vaquero-Melchor, J. López-Araquistain, A. M. Bernardos, J. R. Casar.: Drone Mission Definition and Implementation for Automated Infrastructure Inspection Using Airborne Sensors, *Sensors*, vol. 18, no. 4, pp. 1170, 2018.
- [2] Z. Kaleem, M. H. Rehmani.: Amateur drone monitoring: State-of-the-art architectures key enabling technologies and future research directions, *IEEE Wireless Commun.*, vol. 25, no. 2, pp. 150-159, 2018.
- [3] R. U. M. Mogili, B. B. V. L. Deepak.: Review on application of drone systems in precision agriculture, *Procedia Comput. Sci.*, vol. 133, pp. 502-509, 2018.
- [4] Harvey, M. C., Rowland, J. V., and Luketina, K. M.: Drone with thermal infrared camera provides high resolution georeferenced imagery of the Waikite geothermal area, New Zealand. *J. Volcan. Geotherm.* 325, pp. 61-69, 2016.
- [5] Claesson, A., Bäckman, A., Ringh, M., Svensson, L., Nordberg, P., Djärv, T., and Hollenberg, J.: Time to delivery of an automated external defibrillator using a drone for simulated out-of-hospital cardiac arrests vs emergency medical services, *Jama*, 317, pp. 2332-2334, 2017.
- [6] Wilson, R. L.: Ethical issues with use of drone aircraft, In *Proceedings of the IEEE 2014 International Symposium on Ethics in Engineering, Science, and Technology.*, IEEE Press 2014-5, 2014.
- [7] 野波健蔵.: ドローン技術の現状と課題およびビジネス最前線, *情報管理*, 59(11): pp. 755-763, 2017.
- [8] 野波健蔵.: 世界のドローン開発動向と農業応用からみた課題と展望, *計測と制御*, 55(9), pp. 780-787, 2016.
- [9] 久保大輔.: 無人航空機システム（ドローン）の歴史と技術発展, *計測と制御*, 56(1), pp. 12-17, 2017.
- [10] 関口大介, 岩崎覚史.: *ドローンビジネス参入ガイド*, 翔泳社, 2017.

- 
- [11] 野波健蔵.: 「空の産業革命」をもたらすドローンの課題と展望, 計測と制御, 56(1), pp. 2-2, 2017.
- [12] 総務省統計局, 平成 27 年国勢調査, 世帯構造等基本集計結果, 2017.
- [13] 原俊彦, 森田朗.: 国立社会保障・人口問題研究所編「日本の人口動向とこれからの社会—人口潮流が変える日本と世界」, 東京大学出版会, 人口学研究, pp. 69-72, 2017.
- [14] 国土交通省, 日本物流団体連合会, 「数字でみる物流 2018」, 2018.
- [15] 国土交通省 HP, 「平成 29 年度宅配便取扱実績について」, 2017.  
<https://www.mlit.go.jp/common/001252227.pdf>
- [16] 厚生労働省「賃金構造基本統計調査」, 2018.
- [17] 国土交通省 HP, 「宅配便と再配達の現状」, 2018.  
[http://www.mlit.go.jp/seisakutokatsu/freight/re\\_delivery\\_reduce.html](http://www.mlit.go.jp/seisakutokatsu/freight/re_delivery_reduce.html)
- [18] 公益社団法人全日本トラック協会, 日本のトラック輸送産業現状と課題 2019, 2019
- [19] 総務省, 平成 30 年度労働力調査, 2018.
- [20] Amazon HP, Ycco Mini Pocket Drone Newest X20 Headless Mode 2.4Ghz Nano LED RC Quadcopter Altitude Hold 500w Pixel Multi-Protection Drone for Kids and Beginners to Play Indoor-Red  
<https://www.amazon.com/Ycco-Quadcopter-Multi-Protection-Beginners-Indoor-Red/dp/B07S8PYXBJ>
- [21] DJI HP, Mavic 2, <https://www.dji.com/jp/mavic-2>
- [22] NewsWitch HP, 「3 時間も連続飛行できる新型ドローンの正体」  
<https://newswitch.jp/p/16670>
- [23] Gizmodo HP, 「中国のドローンは人を乗せて空を飛ぶ. およそ 1,000 回テスト済み」  
<https://www.gizmodo.jp/2018/02/chinese-drone-1000-test.html>
- [24] Murray, C. C., and Chu, A. G.: The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery, *Transport. Res. Part C: Emerg. Technol.*, 54, pp. 86–109, 2015.
- [25] Boysen, N., Schwerdfeger, S., and Weidinger, F.: Scheduling last-mile deliveries with truck-based autonomous robots, *Europ. J Operat. Res.*, 271, pp. 1085–1099, 2018.
- [26] Rabta, B., Wankmüller, C., and Reiner, G.: A drone fleet model for last-mile distribution in disaster relief operations, *Int. J. Disaster Risk Reduct.*, 28, pp. 107–112, 2018.
- [27] Ferrandez, S. M., Harbison, T., Weber, T., Sturges, R., and Rich, R.: Optimization of a truck-drone in tandem delivery network using k-means and genetic algorithm, *J. Indust. Eng. Manag. JIEM*, 9, pp. 374–388. 2016.

- 
- [28] Chiang, W. C., Li, Y., Shang, J., and Urban, T. L.: Impact of drone delivery on sustainability and cost: Realizing the UAV potential through vehicle routing optimization, *J. Appl. Energ.*, 242, pp. 1164–1175, 2019.
- [29] Dorling, K., Heinrichs, J., Messier, G. G., and Magierowski, S.: Vehicle routing problems for drone delivery, *Trans. Syst, Man, Cyber Syst, IEEE J*, 47, pp. 70–85, 2016.
- [30] UPS pressroom, UPS Flight Forward, CVS Pharmacy to Develop Drone Delivery Applications, 2019.  
<https://pressroom.ups.com/pressroom/ContentDetailsViewer.page?ConceptType=PressReleases&id=1571676331520-698>
- [31] Supply Chain 247, Transportation news, UPS Drone Meets UPS Delivery Truck, 2017  
[https://www.supplychain247.com/article/ups\\_drone\\_meets\\_ups\\_delivery\\_truck/ups\\_capital](https://www.supplychain247.com/article/ups_drone_meets_ups_delivery_truck/ups_capital)
- [32] 土屋武司.: ドローンの法規制. *IATSS Review (国際交通安全学会誌)*, 44(2), pp. 108-116, 2019.
- [33] 徳永博樹.: 改正航空法の概要と環境整備に向けた取り組み (空の産業革命—「飛行ロボット」としての次世代ドローン—). *日本機械学会誌*, 121(1200), pp. 8-9, 2018.
- [34] 国立研究開発法人, 新エネルギー・産業技術総合開発機構 HP, 「ロボット・ドローンが活躍する省エネルギー社会の実現プロジェクト」, 2017  
[https://www.nedo.go.jp/activities/ZZJP2\\_100080.html](https://www.nedo.go.jp/activities/ZZJP2_100080.html)
- [35] 浜中雅俊.: ドローン飛行網の最適設計手法, 第78回全国大会講演論文集, pp. 279-280, 2016.
- [36] 経済産業省, 空の産業革命に向けたロードマップ 2019 (小型無人機の安全な利活用のための技術開発と環境整備, 2019.6.21), 2019.  
<https://www.kantei.go.jp/jp/singi/kogatamujinki/pdf/siryou12.pdf>
- [37] 古谷知之, 南政樹.: ドローン前提社会, *IATSS Review*, *国際交通安全学会誌*, 44(2), pp. 92-99, 2019.
- [38] 国土交通省, 物流政策課 物流を取り巻く現状について, 2018.  
<http://www.mlit.go.jp/common/001258392.pdf>
- [39] 経済産業省, 買物弱者対策に関する実態調査結果報告書, 2017.  
[http://www.soumu.go.jp/main\\_content/000496982.pdf](http://www.soumu.go.jp/main_content/000496982.pdf)
- [40] 国土交通省, 過疎地域等におけるドローン物流ビジネスモデル検討会中間とりまとめ, 2019.
- [41] LO, Andrew W. SIAH, Kien Wei WONG, Chi Heem.: Machine learning with statistical imputation for predicting drug approvals. *Harvard Data Science Review*, 2019.

- 
- [42] Philip E.: Iceland's consumers try drone delivery: The startup Aha takes on Amazon with basic drones bearing burgers, *IEEE Spectrum*, 55(10), pp. 12-13, 2018.
- [43] 穴井宏和, 齊藤努, 今日から使える! 組合せ最適化 離散問題ガイドブック, 講談社出版, 2015.
- [44] 穴井宏和, 数理最適化の実践ガイド, 講談社出版, 2013.
- [45] 大林茂, 航空機の多目的最適設計, *人工知能学会誌*, 18(5), pp. 495-501, 2003.
- [46] Grefenstette, J., Gopal, R., Rosmaita, B., and Van Gucht, D.: Genetic algorithms for the traveling salesman problem. In *Proceedings of the first International Conference on Genetic Algorithms and their Applications*, Lawrence Erlbaum, pp. 160-168, 1985.
- [47] John C. Martin.: *Introduction to Languages and the Theory of Computation*, 2nd edition ed, McGraw-Hill, ISBN 0-07-040845-9, Section 9.6, Nondeterministic Turing machines, pp. 277-281, 1997.
- [48] 相吉英太郎, 安田恵一郎, メタヒューリスティクスと応用, 電気学会出版, 2007.
- [49] Yaguchi, K., Tamura, K., Yasuda, K., and Ishigame, A.: Basic study of proximate optimality principle based combinatorial optimization method. In *2011 IEEE International Conference on Systems, Man, and Cybernetics*. IEEE, pp. 1753-1758, 2011.
- [50] Goldberg, D. E., and Holland, J. H.: Genetic algorithms and machine learning, *Machine learning*, 3(2), pp. 95-99, 1988.
- [51] J. H. Holland.: *Adaptation in natural and artificial systems*, University of Michigan Press, Ann Arbor, 1975.
- [52] Yao, X., Liu, Y., and Lin, G.: Evolutionary programming made faster. *IEEE Transactions on Evolutionary computation*, 3(2), pp. 82-102, 1999.
- [53] Sinha, N., Chakrabarti, R., and Chattopadhyay, P. K.: Evolutionary programming techniques for economic load dispatch. *IEEE Transactions on evolutionary computation*, 7(1), pp. 83-94, 2003.
- [54] W. Banzhaf, P. Nordin, R.E. Keller, and F.D. Francone. *Genetic Programming, An Introduction*. Morgan Kaufmann, 1998.
- [55] Yuryevich, J., and Wong, K. P.: Evolutionary programming based optimal power flow algorithm, *IEEE transactions on Power Systems*, 14(4), pp. 1245-1250, 1999.
- [56] Hansen, N.: The CMA evolution strategy: a comparing review. In *Towards a new evolutionary computation*, Springer, Berlin, Heidelberg, pp. 75-102, 2006.
- [57] Knowles, J. D., and Corne, D. W.: Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary computation*, 8(2), pp. 149-172, 2000.

- 
- [58] D. B. Fogel. *Evolutionary Computation. Toward a New Philosophy of Machine Intelligence*. IEEE Press, Piscataway, NJ, 1995.
- [59] Bäck, T., Fogel, D. B., and Michalewicz, Z.: *Evolutionary computation 1: Basic algorithms and operators*. CRC press, 2018.
- [60] A.E. Eiben and J.E. Smith, *Introduction to Evolutionary Computing*, Springer, 2003.
- [61] Fogel, D. B.: What is evolutionary computation, *IEEE spectrum*, 37(2), pp. 26-32, 2000.
- [62] 北野宏明, 遺伝的アルゴリズム 2, 産業図書出版, 1995.
- [63] 北野宏明, 遺伝的アルゴリズム 4, 産業図書出版, 2000.
- [64] 三宮信夫, 喜多一, 玉置久, 岩本貴史.: *遺伝アルゴリズムと最適化*, 朝倉書店, 1998.
- [65] Shi, Yuhui.: Particle swarm optimization: developments, applications and resources. In: *Proceedings of the 2001 congress on evolutionary computation*, IEEE Cat. No. 01TH8546. IEEE, pp. 81-86, 2001.
- [66] Eberhart, R., and Kennedy, J.: Particle swarm optimization. In *Proceedings of the IEEE international conference on neural networks*, Vol. 4, pp. 1942-1948, 1995.
- [67] Glover, F., & Laguna, M.: Tabu search. In *Handbook of combinatorial optimization*, Springer, Boston, MA, pp. 2093-2229, 1998.
- [68] Gallego, R. A., Romero, R., and Monticelli, A. J.: Tabu search algorithm for network synthesis, *IEEE Transactions on Power Systems*, 15(2), pp. 490-495, 2000.
- [69] Abido, M. A.: Optimal power flow using tabu search algorithm. *Electric power components and systems*, 30(5), pp. 469-483, 2002.
- [70] Burke, E., De Causmaecker, P., and Berghe, G. V.: A hybrid tabu search algorithm for the nurse rostering problem. In *Asia-Pacific Conference on Simulated Evolution and Learning*, Springer, Berlin, Heidelberg, pp. 187-194, 1998.
- [71] Aarts, E., and Korst, J.: *Simulated annealing and Boltzmann machines*, 1988.
- [72] Van Laarhoven, P. J., and Aarts, E. H.: *Simulated annealing. In Simulated annealing: Theory and applications*, Springer, Dordrecht, pp. 7-15, 1987.
- [73] Dorigo, M., and Di Caro, G.: Ant colony optimization: a new meta-heuristic. In *Proceedings of the 1999 congress on evolutionary computation-CEC99*, IEEE, Vol. 2, pp. 1470-1477, 1999.
- [74] Dorigo, M., and Birattari, M.: *Ant colony optimization*. Springer US, pp. 36-39, 2010.
- [75] Wolpert, D. H., and Macready, W. G.: No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1), pp. 67-82, 1997.
- [76] Marler, R. T., and Arora, J. S.: The weighted sum method for multi-objective optimization: *Structural and multidisciplinary optimization*, 41(6), pp. 853-862, 2010.

- 
- [77] Deb, K.: Multi-objective optimization. In Search methodologies. Springer, Boston, MA, pp. 403-449, 2014.
- [78] Fonseca, C. M., & Fleming, P. J.: Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization. In *Icga Vol. 93*, pp. 416-423, 1993.
- [79] Marler, R. T., and Arora, J. S.: Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26(6), pp. 369-395, 2004.
- [80] Konak, A., Coit, D. W., and Smith, A. E.: Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering and System Safety*, 91(9), pp. 992-1007, 2006.
- [81] 高濱徹行, 阪井節子.:  $\epsilon$  制約遺伝的アルゴリズムによる制約付き最適化, *情報処理学会論文誌*, 47(6), 1861-1871, 2006.
- [82] Fonseca, C. M., and Fleming, P. J.: Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization. In *Icga, Vol. 93*, pp. 416-423, 1993.
- [83] Parsopoulos, K. E., Tasoulis, D. K., Pavlidis, N. G., Plagianakos, V. P., and Vrahatis, M. N.: Vector evaluated differential evolution for multiobjective optimization. In *Proceedings of the 2004 Congress on Evolutionary Computation, IEEE Cat. Vol. 1*, pp. 204-211, 2004.
- [84] Murata, T., and Ishibuchi, H.: MOGA: Multi-objective genetic algorithms. In *IEEE international conference on evolutionary computation, Vol. 1*, pp. 289-294, 1995.
- [85] Murata, T., Ishibuchi, H., and Tanaka, H.: Multi-objective genetic algorithm and its applications to flowshop scheduling. *Computers and Industrial Engineering*, 30(4), pp. 957-968, 1996.
- [86] Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *International conference on parallel problem solving from nature*. Springer, Berlin, Heidelberg, pp. 849-858, 2000.
- [87] Deb, K., and Goel, T.: Controlled elitist non-dominated sorting genetic algorithms for better convergence. In *International conference on evolutionary multi-criterion optimization*. Springer, Berlin, Heidelberg, pp. 67-81. 2001.
- [88] Kuriakose, S., and Shunmugam, M. S.: Multi-objective optimization of wire-electro discharge machining process by non-dominated sorting genetic algorithm. *Journal of materials processing technology*, 170(1-2), pp. 133-141, 2005.
- [89] Zitzler, E., Laumanns, M., and Thiele, L. SPEA2: Improving the strength Pareto evolutionary algorithm. *TIK-report*, pp. 103, 2001.

- 
- [90] Abido, M. A.: Multiobjective optimal VAR dispatch using strength pareto evolutionary algorithm. In 2006 IEEE International Conference on Evolutionary Computation, pp. 730-736, 2006.
- [91] Zitzler, E., and Thiele, L.: An evolutionary algorithm for multiobjective optimization: The strength pareto approach. TIK-report, 43, 1998.
- [92] Omagari, H., and Higashino, S. I.: Provisional-Ideal-Point-Based Multi-objective Optimization Method for Drone Delivery Problem. International Journal of Aeronautical and Space Sciences, 19(1), pp. 262-277, 2018.
- [93] Omagari, H.: Potential benefits assessment on home delivery service using multiple drones. In Proceedings of the 31st International Council of Aeronautical Sciences, Belo Horizonte, Brazil, ICAS 2018.
- [94] Omagari, H., and Higashino, S. I.: Aspiration-point-based multi-objective path planning method for an unmanned aerial vehicle. TRANSACTIONS OF THE JAPAN SOCIETY FOR AERONAUTICAL AND SPACE SCIENCES, AEROSPACE TECHNOLOGY JAPAN, 15(APISAT-2016), a99-a108, 2017
- [95] Runarsson, T. P., and Yao, X.: Stochastic ranking for constrained evolutionary optimization. IEEE Transactions on evolutionary computation, 4(3), pp. 284-294, 2000.
- [96] Koziel, S., Michalewicz, Z.: Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization. Evolutionary computation, 7(1), pp. 19-44, 1999.
- [97] Deb, K., Thiele, L., Laumanns, M., and Zitzler, E.: Scalable test problems for evolutionary multiobjective optimization. In Evolutionary multiobjective optimization, Springer, London, pp. 105-145, 2005.
- [98] Google map, <http://yagijijii.com/google/nonlabelmap.htm>
- [99] Mitsutake, K., and Higashino, S.: An A\*-EC hybrid path planning method for waypoints traveling problem considering terrain, AIAA Guid., Navig. Contr. Conf. Exhibit, 2008.
- [100] 満武勝嗣, 東野伸一郎.: 障害物回避を考慮したリアルタイム 3次元飛行経路生成法の高速化. 日本航空宇宙学会論文集, 58(677), 153-163, 2010.
- [101] Reinelt, G.: TSPLIB—A traveling salesman problem library. ORSA journal on computing, 3(4), pp. 376-384, 1991.
- [102] MP-TESTDATA - The TSPLIB Symmetric Traveling Salesman Problem Instances <http://elib.zib.de/pub/mp-testdata/tsp/tsplib/tsp/>

## 謝辞

学位論文の執筆にあたり、九州大学大学院工学府飛行力学研究室の東野伸一郎准教授におかれましては、一貫して多大なるご支援とご指導を賜りました。特に国際学会での口頭発表と査読論文投稿時には自己の経験を交えながら最後まで親身になってご指導していただきました。この研究に1つの区切りをつけることができたことは、紛れもなく先生からのご指導の賜物であると心より御礼申し上げます。また、本大学院で研究に専念する機会を与えて下さった航空自衛隊の関係者各位の皆様におかれましては、5年の長きにわたり一貫して大学での研究生活をご支援していただきました。このような素晴らしい機会を与えてくれた部隊の皆様には感謝の念が絶えません。ここに改めて御礼申し上げます。研究室OBの皆様や後輩には研究室ゼミを通じてお互いに研究内容について深く議論し合い、研鑽することができました。このような貴重な機会があったからこそ自身の研究もより深く探求することができたと感じております。本当にありがとうございました。最後に、拙著を上梓するこの日まで研究生活を日々献身的に支え続けてくれた妻、そして自身の励みとなってくれた娘の存在に心から感謝申し上げます。