

倉庫を介して貨客混載を行うデマンドバスシステムの モデル化とシミュレーション解析

武田 淳弘¹ 北 栄輔^{1,a)}

概要：人口減少等で通常のバスシステムでは採算が取れない地域の一部では、需要に基づいてスケジュールや経路などを決定するデマンドバスが導入されている。他方で、貨物と人員を共に一つの車両で運搬する貨客混載の取り組みは収益を向上し運用費用をさらに減少できる事から注目を集めている。本研究では、倉庫を用いた物流下で貨客混載を行うデマンドバスの配送計画問題を混合整数計画問題として定式化し、静的な状況で貨客混載の効果について検証を行った。一方で、提案モデルは NP 困難に分類される問題であることから、数理ソルバーや効率的な厳密解法を用いたとしても現実的な時間で有用な解が得られない可能性がある。このため、本研究では近傍探索の手法を取り入れた遺伝的アルゴリズムを用いて実行時間内に有用な解を得られる事を示した。そして、提案モデルの計算時間は従来の一般的な貨客混載を行うデマンドバスに適用された数理モデルよりも短くなる事を示した。

キーワード：デマンドバス、貨客混載、Dial-a-Ride 問題、配送計画問題、Mobility as a Service (MaaS)

Modeling and simulation analysis of a demand bus system with mixing freight and passenger through a warehouse

Keywords: Demand-responsive transport, Ridesharing, Dial-a-ride Problem, Vehicle Routing Problem, Mobility as a Service (MaaS)

1. はじめに

あらかじめ決められた運航経路と運航計画のもとで乗客を乗降させる通常の固定路線バスに対して、乗客の需要に応じて経路やスケジュールを決定するバスはデマンドバスと呼ばれている。日本では人口減少や高齢化等で通常の固定路線バスシステムでは採算が取れない地域の一部で、運用費用を減少させる目的でデマンドバスシステムが導入されている。しかしながら、それでもなおデマンド交通を赤字で運営する事業者は少なくない [1]。

これに対して、貨物と人員を同時に運搬する貨客混載の取り組みは、さらなる収入改善や運行費用を削減できる可能性があることから注目を集めている。このような貨客混載の取り組みは、人流と物流を統合して最適化を行うため

Mobility as a Service (MaaS) の一種の取り組みとしても注目を集めており、実証実験が行われている。加えて、日本では法改正などにより自家用有償旅客運送で貨客混載を行う事が可能になった。これにより、民間事業者ではない自治体が運営するデマンド交通などでも貨客混載の試みがなされている [2]。

一方で、実世界での貨客混載の試みはすでに利用している乗客との兼ね合いや、物流事業者との調整などから制約も多い。そこで、本研究は貨客混載を行うデマンドバスシステムを数理モデル化し、計算機シミュレーションを行うことで貨客混載の効果を明らかにすることを目的とする。この目的のため、汎用ソルバーを用いる場合でもある程度の大きさの入力まで扱える数理モデルを新たに提案する。しかしながら、提案モデルは NP 困難に属する問題であるため、大きな入力に対しては汎用ソルバーや効率的な厳密解法を用いたとしても現実的な時間で有用な解が得られない可能性がある。そこで、有用な解を実行時間内に有用な

¹ 名古屋大学大学院 情報学研究科
Graduate School of Informatics, Nagoya University
^{a)} kita@i.nagoya-u.ac.jp

解を導出する手法についても提案する。

2. 関連研究

デマンドバスを含む車両の配送問題は、古くから配送計画問題 (VRP: Vehicle Routing Problem) [3] として定式化され、研究が行われてきた。この配送計画問題はデポ (depot) と呼ばれる車両拠点から複数の車両が出発し、全ての顧客をちょうど一度ずつ訪問した後にデポに帰着する経路の集合の中で距離が最小となる経路を求める組合せ最適化問題である。デマンドバスの配送計画問題は、この配送計画問題にデポ以外の地点で顧客が乗車した後に下車をする事に関する制約や、乗客の上下車の時間枠の制約などを加えた Dial-a-ride 問題 (DARP)[4] として定式化され研究が行われている。

これに対して、Li ら (2014)[5] は Dial-a-ride 問題を基礎としてタクシーが人員と貨物の両方を運搬する状況を扱った数理モデルを Share-a-Ride 問題 (SARP) として定式化した。SARP が DARP と大きく異なる点は、乗客を乗せる車両に貨物も同時に積載する状況を扱っている点である。SARP では、利益を最大にするためにどの貨物を運搬するか取捨選択する問題を扱っている。また、Yu ら (2018) [6] は SARP を一般化したモデル (g-SARP) を提案した。g-SARP では SARP と比べて乗車時間に制約がない他、貨物を全て運ぶなどの変更が加えられている。

一方で、これらの定式化は Dial-a-ride 問題を基礎として複雑化したことから、汎用ソルバーを用いる場合等に計算時間が長く必要になる傾向がある。そこで、本研究では従来より簡単な問題となる、倉庫を用いた物流下で貨客混載を行うデマンドバスに対する配送計画問題を提案し、対応する数理モデルも紹介する。図 1 は、従来の研究で主に対象となっている状況と本研究で対象とする状況の違いを表した図である。

3. 提案モデル

本節では、倉庫を介して貨客混載を行うデマンドバスの配送計画問題を混合整数計画問題として定式化する。まず、3.1 節で提案モデルの前提条件について、3.2 節で数理モデルについて説明する。

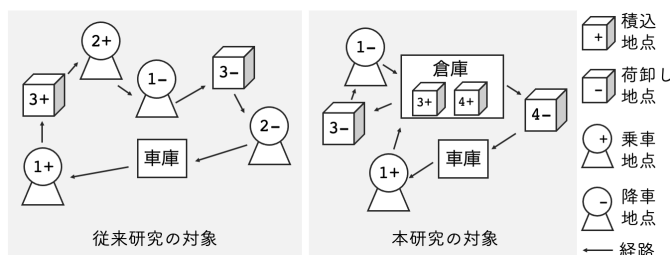


図 1 従来研究と本研究との扱う状況の比較

Fig. 1 Comparison of previous models and our model

表 1 提案モデルで用いる変数

Table 1 Notations for proposal model

変数	説明
n	乗客数
m	貨物数
σ	総停車回数. $\sigma := 2n + m$
k	車両数. 車両の集合を $K := \{1, 2, \dots, k\}$ と表記する.
V^{po}	乗車地点の集合. $V^{po} := \{1, 2, \dots, n\}$
V^{pd}	降車地点の集合. $V^{pd} := \{n + 1, n + 2, \dots, 2n\}$
V^{fd}	荷下ろし地点の集合. $V^{fd} := \{2n + 1, \dots, 2n + m\}$
V^{dp}	車両拠点 (デポ) の集合. $V^{dp} := \{0\}$
V^{wh}	倉庫の集合. $V^{wh} := \{\sigma + 1\}$
V	頂点集合. $V := V^{po} \cup V^{pd} \cup V^{fd} \cup V^{dp}$
d_{ij}^k	車両 k の頂点 i から頂点 j への移動費用
t_{ij}^k	車両 k の頂点 i から頂点 j への移動時間
q_i^c	リクエスト i の積載種類 $c \in \{p, f\}$ の数量
s_i	リクエスト i の処理時間
$[e_i, l_i]$	リクエスト i の処理時間枠
H_i	リクエスト i の乗車上限時間
U^{kc}	車両 k の積載種類 $c \in \{p, f\}$ の積載上限

3.1 前提条件

提案モデルで使用する表記を表 1 に記載する。また、提案モデルで扱う状況について、以下にそれぞれの項目ごとに説明する。

設計目的: 日本のデマンドバスシステムの多くは距離によらず運賃は一定となっており、また貨物の配送料についても距離によらない事が多いことから、走行に掛かる費用を最小化する事を目的とする。

運搬する対象: それぞれの車両は、貨物 (f) と人員 (p) を運搬する。また、貨物と人員の需要量 (q_i^c)、需要が発生する時間 ($[e_i, l_i]$)、需要が発生する位置は予め判明しているものとする。

乗客: 乗客は任意の地点で乗車した後に任意の地点で下車する。乗客は希望する乗車時刻と下車時刻を幅 ($[e_i, l_i]$) をもって指定でき、また乗車時間に上限 (H_i) を持つ事ができる。また、乗降に一定の時間 (s_i) が必要である。一回で乗降のする人数に制限はない。

貨物: 貨物は倉庫で積み込まれ、任意の地点でおろされる。荷下ろしを行う時間は、幅 ($[e_i, l_i]$) をもって指定する事ができる。一回の積み下ろしの個数に制限はない。倉庫での積み込みは個数に関わらず一定の時間 ($s_{\sigma+1}$) が必要であり、荷下ろしにも一定の時間 (s_i) が必要である。

倉庫: 倉庫には経路の途中で何度でも訪れる事ができ、倉庫では車両の貨物の積載上限 (U^f) まで荷物を積む事が可能である。別途制約がない限り、乗客を乗せた状態でも倉庫に訪れる事ができる。

車両: 車両は初めにデポから出発し、最後にデポに帰着する。人員と貨物のそれぞれに積載上限 (U^p, U^f) を設定する事ができる。

表 2 提案モデルでの設計変数とその説明
Table 2 Design variables for proposal model

設計変数	説明
x_{ij}^k	車両 k が頂点 i から j に移動したとき 1, それ以外の時 0
y_{ij}^k	車両 k が頂点 i から倉庫を経由して 頂点 j に移動したとき 1, それ以外の時 0
Q_i^{kc}	車両 k が頂点 i を訪れた後の積載物 c の積載量
B_i^k	車両 k が頂点 i でサービスを開始した時間
L_i^k	リクエスト i の乗車時間

サービス時間： 全ての車両がデポに戻る時間の限度はデポの時間枠 (l_0) によって制限する事ができる。また, 車両がそれぞれのリクエストのサービス開始時間 (e_i) よりも早く到着した場合, サービス開始時間まで待機しなければならない。

3.2 定式化

本節では, まず車両の性能が異なる場合を許す一般的な状況で倉庫を介して貨客混載を行うデマンドバスシステムの定式化について紹介し, 後に定式化についての具体的な説明を行う。

提案モデルで使用する設計変数とその説明を表 2 に示す。倉庫は複数回訪れる事ができ, その前後で時間や乗客数などを保存しなければいけないため, 他の頂点と同じように一つの頂点として扱う事が難しい。よって, 本研究での定式化では Koc ら (2011) [7] が用いたやり方と類似の手法を用いて設計変数 y_{ij}^k を導入する。本研究で提案する倉庫を介して貨客混載を行うデマンドバスシステムの定式化は以下の通りである。(なお, W_{ij}^{kc} は $W_{ij}^{kc} \geq \min\{U^{kc}, U^{kc} + q_i^c\}$ を満たす定数であり, M_{ij}^k は $M_{ij}^k \geq \max\{0, l_i + s_i + t_{ij} - e_j\}$ を満たす定数である。)

目的関数：

$$\text{minimize} \quad \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} [d_{ij}^k x_{ij}^k + (d_{i,\sigma+1}^k + d_{\sigma+1,j}^k - d_{ij}^k) y_{ij}^k] \quad (1)$$

制約条件：

$$\sum_{k \in K} \sum_{j \in V} x_{ij}^k = 1, \quad \forall i \in V \setminus V^{dp} \quad (2)$$

$$\sum_{j \in V} x_{0j}^k \leq 1, \quad \forall k \in K \quad (3)$$

$$\sum_{j \in V} x_{ji}^k - \sum_{j \in V} x_{ij}^k = 0, \quad \forall i \in V, \forall k \in K \quad (4)$$

$$\sum_{j \in V} x_{ij}^k - \sum_{j \in V} x_{i+n,j}^k = 0, \quad \forall i \in V^{po}, \forall k \in K \quad (5)$$

$$x_{ij}^k - y_{ij}^k \geq 0, \quad \forall i \in V, \forall j \in V, \forall k \in K \quad (6)$$

$$Q_j^{kp} \geq Q_i^{kp} + q_j^p - W_{ij}^{kp} (1 - x_{ij}^k) \quad \forall i \in V \setminus V^{dp}, \forall j \in V, \forall k \in K \quad (7)$$

$$Q_j^{kf} \geq Q_i^{kf} + q_j^f - W_{ij}^{kf} (1 - x_{ij}^k + y_{ij}^k)$$

$$\forall i \in V \setminus V^{dp}, \forall j \in V, \forall k \in K \quad (8)$$

$$Q_0^{kf} = U^f, \quad \forall k \in K \quad (9)$$

$$\max\{0, q_i^c\} \leq Q_i^{kc} \leq \min\{U^c, U^c + q_i^c\} \quad \forall i \in V, \forall c \in \{p, f\}, \forall k \in K \quad (10)$$

$$B_j^k \geq t_{0j}^k (x_{0j}^k - y_{0j}^k) + (t_{0,\sigma+1}^k + t_{\sigma+1,j}^k + s_{\sigma+1}) y_{0j}^k \quad \forall j \in V \setminus V^{dp}, \forall k \in K \quad (11)$$

$$B_j^k \geq B_i^k + s_i + t_{ij}^k - M_{ij}^k (1 - x_{ij}^k) + (t_{i,\sigma+1}^k + t_{\sigma+1,j}^k + s_{\sigma+1} - t_{ij}^k) y_{ij}^k \quad \forall i \in V \setminus V^{dp}, \forall j \in V, \forall k \in K \quad (12)$$

$$e_i \leq B_i^k \leq l_i, \quad \forall i \in V, \forall k \in K \quad (13)$$

$$L_i^k = B_{i+n}^k - (B_i^k + s_i), \quad \forall i \in V^{po}, \forall k \in K \quad (14)$$

$$t_{i,i+n}^k \leq L_i^k \leq H_i, \quad \forall i \in V^{po}, \forall k \in K \quad (15)$$

$$x_{ij}^k \in \{0, 1\}, \quad \forall i \in V, \forall j \in V \quad (16)$$

$$y_{ij}^k \in \{0, 1\}, \quad \forall i \in V, \forall j \in V \quad (17)$$

式 (2) は各リクエストが厳密に一度処理される制約を定義している。式 (3) と式 (4) は車両がデポから出発しデポに帰着する制約を定義し, また各頂点に入る車両と出ていく車両が同一である制約を定義している。式 (5) と式 (15) は各リクエストに対して, 乗車地点の後に降車地点を訪れる制約と, 乗車地点と降車地点に同一の車両が訪れる制約を定義している。式 (6) は倉庫を訪れる事ができる状態を制限している。式 (7) から式 (10) はそれぞれの頂点で積載上限を超えることを禁止する制約を定義している。式 (7) と式 (8) で違いが存在するのは, 積載人数は倉庫を訪れる前後で保存される一方で貨物数は倉庫を訪れる前後で異なるためである。式 (11) と式 (12) はそれぞれの頂点でサービスを開始する時間を定義し, 式 (13) でそれぞれのリクエストに対応した時間枠内で処理をする制約, 式 (14) と式 (15) で乗客の乗車時間が上限以下である制約を定義している。

4. 遺伝的アルゴリズムを用いた解法

3 節で提案したモデルは NP 困難に分類される問題であることから, 汎用ソルバーや効率的な厳密解法を用いたとしても実用的な時間内に有用な解を導出できない可能性がある。そこで, 本節ではヒューリスティック解法として遺伝的アルゴリズムを用いた求解を試みる。本研究で用いた遺伝的アルゴリズムは, 巨大近傍探索の手法を各世代ごとで適用確率を変化させながら使用しているという特徴を持つ。4.1 節では本研究で用いた遺伝的アルゴリズムの枠組みについて説明し, 4.2 節以降では遺伝的アルゴリズムの各プロセスの概要について説明を行う。

4.1 本研究で用いた遺伝的アルゴリズムの枠組み

本研究で提案したモデルは DARP を元に発展させている

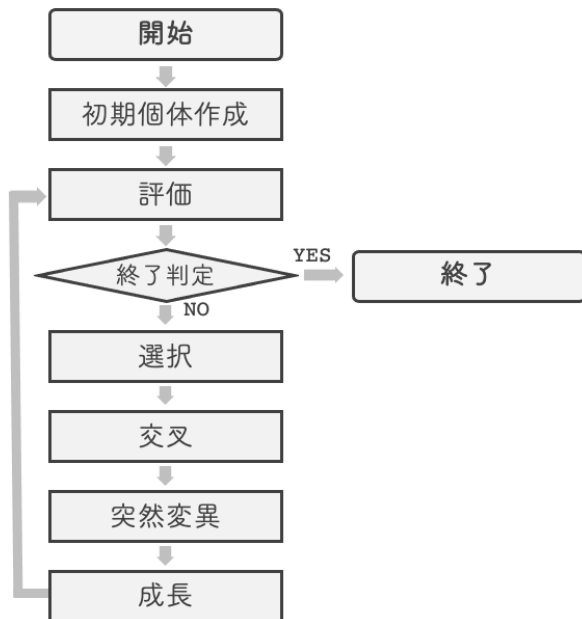


図 2 本研究で用いた遺伝的アルゴリズムの流れ
 Fig. 2 Genetic algorithm flow

ため、DARP に対して有効であったヒューリスティック手法が本モデルに対しても有効であると考えられる。DARP に関するレビュー論文 [8] によれば、他の手法と遺伝的アルゴリズムを組み合わせた手法 [9] がタブーサーチなどの他のヒューリスティック手法を単に用いた時と比べて良い解を導出している。よって、本研究でも巨大近傍探索 (LNS: Large Neighborhood Search) の手法を各世代で一定数の個体に対して適用し、それぞれの個体の解の質の向上 (成長) を行うようにした遺伝的アルゴリズムを用いて求解を行う。本研究で用いた遺伝的アルゴリズムの流れを図 2 に示す。

4.2 各プロセスの概要

終了条件

最良値が一定数の世代にわたって更新されない、または世代数が計算時間の上限に達した時、探索を終了する。

染色体表現

本研究で用いた染色体の表現は、経路に対応する頂点の車両数分の配列である。より具体的には、デポを除いた頂点の順序付きリストを車両数分内包する配列である。なお、貨物はそれぞれの荷下ろし地点に対応するダミーの積み込み頂点を導入する。

ここで、リクエスト i に対応する乗車地点を n_i^+ 、ダミーの積み込み地点を (n_i^+) 、降車 (荷下ろし) 地点を n_i^- と表現するとする。この時、2 台の車両が 2 組の乗客と 3 個の荷物を運搬する状況に対応する染色体の表現は次のようなものがある: $[[(3+), (5+), 2+, 5-, 2-, 3-], [1+, (4+), 1-, 4-]]$ 。この例の場合、1 台目の車両の経路は、(デポ) → (倉庫) → 乗客 2 の乗車地点 → 貨物 5 の荷下ろし地点 → 乗客 2 の降

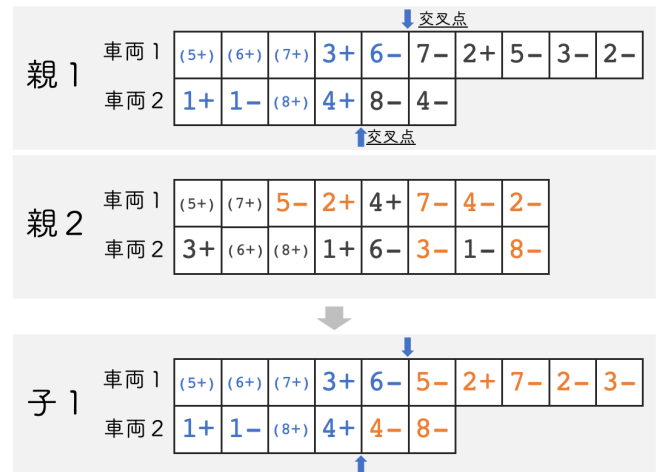


図 3 順序交叉のイメージ。子 1 は親 1 の経路をそのまま引き継ぐが、交叉点以降の頂点順序は親 2 の順序を適用する。

Fig. 3 Illustration of order crossover

車地点 → 貨物 3 の荷下ろし地点 → (デポ) である。

初期個体の生成

初期個体については、乗車 (積み込み) 地点を訪れた後に降車 (荷下ろし) 地点を訪れるという制約を満たす中でランダムに個体を生成する。

適応度

適応度は、違反した制約数に総走行距離を加えた数値の逆数とする。

選択

適応度の高い上位 n 個体を次世代の集団に残すというエリート保存と呼ばれる手法と、適応度に応じた確率で選ばれるルーレット選択と呼ばれる手法を用いる。

交叉

交叉は一点の順序交叉 (OX: Order Crossover) を用いる (図 3 参照)。

突然変異

突然変異プロセスでは、同一車両内で頂点が移動する突然変異か、車両をまたいで頂点が移動する突然変異が発生する。これらの突然変異が発生した後でも、頂点順序は守られるように変異を行う。

成長

成長プロセスでは、一部の個体に対して局所探索や巨大近傍探索の手法を適用し、解の質を向上させる。巨大近傍探索とは、経路からなんらかの基準で頂点を取り除いた後、再び何らかの基準を用いて頂点を経路に挿入する探索手法である。本研究では、評価方法と取り除く頂点が異なる 6 つの成長手法を用いる。

また、各世代で成長プロセスを適用する確率や各成長手法を用いる確率を、前世代で成長プロセスによって解の質が向上した割合をもとに変化させる。これは、計算時間の短縮や局所解に陥らないようにする事が目的である。

5. 数値実験

本節では、提案した数理モデルや遺伝的アルゴリズムの数値実験の結果を示す。まず5.1節で数値実験に用いた入力について説明し、5.2節で提案モデルと従来モデルとの数理ソルバーを用いた際の計算時間の違いについて説明を行う。5.3節では、4節で紹介した遺伝的アルゴリズムを用いた場合の計算結果について、5.4節では提案モデルを用いた貨客混載の効果のシミュレーション結果について説明する。

数値実験は、2.3GHz 8コア Intel Core i9-9880H を用いて行い、プログラムはPythonを用いて作成した。

5.1 数値実験条件

提案モデルに対応するようなベンチマーク用の入力は存在しない。そこで、本研究はOSMnx [10] を利用し、実世界の地図データを土台として数値実験用のデータを作成する。入力に用いた地図の例を、図4に示す。地図データに以下を満たすように乗車時間などの入力値を加え、数値実験の入力とする。作成した車両 k 台で乗客 n 組、貨物 m に対処する場合の入力名を人 $[n]$ 物 $[m]$ 車 $[k]$ と表記する。



図4 数値実験で用いた地図データ (名古屋市熱田区)
 Fig. 4 A map data used in numerical experiments

5.2 数理ソルバーを用いた時の提案モデルの計算時間

比較対象の従来モデルとして Yu らによって提案された g-SARP [6] を用いる。g-SARP のパラメータを $\alpha = \beta$, $\gamma_1 = \gamma_2 = \gamma_3$, $\gamma_4 = 0$ とし、貨物の積み込みをすべて倉庫の1か所で行った時、本研究で扱う問題と同等になる。

また、数理ソルバーとして並列化を有効にした CPLEX を使用する。

従来モデルと提案モデルの数理ソルバーを用いた際の計算時間は表3の通りである。結果より、本モデルでの計算時間は従来モデルのものより大幅に短縮できている事が示された。

5.3 遺伝的アルゴリズムを用いた時の計算結果

4節で提案した遺伝的アルゴリズムに巨大近傍探索の手

表3 従来モデルと提案モデルとの数理ソルバーの計算時間の比較 (単位: 秒)

Table 3 Computation time of CPLEX [sec].

入力名	従来モデル	提案モデル
人 [4] 物 [4] 車 [1]	2.88	0.50
人 [4] 物 [4] 車 [2]	12.76	2.12
人 [4] 物 [6] 車 [1]	94.49	3.32
人 [4] 物 [6] 車 [2]	297.47	22.62
人 [6] 物 [4] 車 [2]	255.19	51.13
人 [6] 物 [6] 車 [2]	—*	1604.62

* は1時間で最適解が求められなかった事を意味する。

法を取り入れる効果について検証を行う。

成長プロセスを有しない通常の遺伝的アルゴリズムと、提案手法である成長プロセスを有する遺伝的アルゴリズムの計算結果の比較を表4に示す。

サイズの大きな入力では成長プロセスを有する遺伝的アルゴリズムの方が良い解を導出できたことから、成長プロセスの有効性が示された。一方で、計算時間については成長プロセスを有する遺伝的アルゴリズムの方が長い時間必要になっている。

5.4 貨客混載の効果についてのシミュレーション

提案モデルを使用し、走行する環境を変化させる事で貨客混載の効果についてのシミュレーションを行う。本シミュレーションでは、5.1節で述べた入力に加え、名古屋市熱田区よりも面積の広い名古屋市千種区 [18.18km²] の地図データも用いる (図5, 図6参照)。

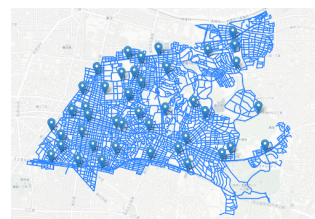


図5 地図データ (千種区)
 Fig. 5 Map data of Chikusa Ward

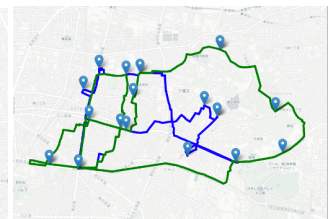


図6 走行経路の例
 Fig. 6 An example of driving routes

シミュレーションは貨客混載を行う車両2台を用いた場合を基準とし、以下の状況別に乗客を6組固定しながら貨物数を変化させる実験を10回ずつ行い、総走行距離の変化について検証を行う。走行経路は、提案モデルに対して

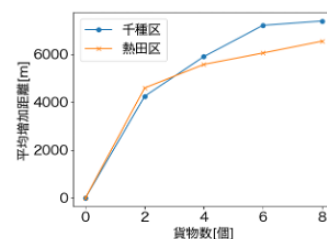


図7 状況1
 Fig. 7 situation1

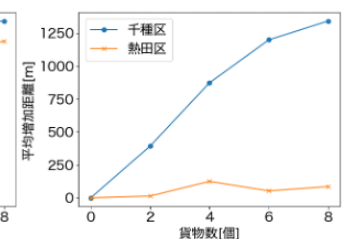


図8 状況2
 Fig. 8 situation2

表 4 遺伝的アルゴリズムの計算結果
Table 4 Computational results of genetic algorithms.

3*入力名	CPLEX		遺伝的アルゴリズム		提案手法	
	最適解	計算時間	最良解	平均 計算時間 *	最良解	平均 計算時間 *
人 [4] 物 [4] 車 [2]	16531.83	2.12	16531.83	1.47	16531.83	0.82
人 [6] 物 [6] 車 [2]	24307.56	1604.62	24307.56	2.50	24307.56	2.45
人 [8] 物 [8] 車 [2]	—	—	35971.10	4.06	35971.10	6.62
人 [10] 物 [10] 車 [2]	—	—	37127.21	7.86	35175.96	88.43
人 [12] 物 [12] 車 [3]	—	—	46976.506	6.98	42615.04	27.26
人 [14] 物 [14] 車 [3]	—	—	55543.72	17.44	48818.20	96.70

* は実行可能解を発見できた時を対象とした平均の値。計算時間の単位は秒 [sec]

数値ソルバーである CPLEX を用いて決定する。

- **状況 1**：乗客専用の車両 2 台と、貨物専用の車両 1 台を使用した場合（すなわち、貨客混載が行われていない場合）。
- **状況 2**：乗客の乗車中に貨物の積み下ろしを行わない制約を加えた場合。

シミュレーション結果は、図 7、図 8 の通りである。

シミュレーション結果より貨物用と乗客用にそれぞれに分けて車両を使用するより貨客混載を行った方が総走行距離が短くなる事が示された。加えて、同じ制約を課した場合でも走行地域によって貨客混載の効果に違いが出る事が示された。以上より、提案モデルを用いて貨客混載のシミュレーションが可能である事が示された。

6. 結論

本研究では、倉庫を用いた物流下で貨客混載を行うデマンドバスに対する配送計画問題を提起し、対応する数値モデルを提案した。また、提案モデルを実世界データに基づいた数値実験に適用して有効性を検討し、従来のモデルに比べて数値ソルバーを用いた際の計算時間が短くなる事を示した。加えて、ヒューリスティック解法として巨大近傍探索の手法を取り入れた遺伝的アルゴリズムを提案し、有用な解を導出できる事を示した。最後に、提案モデルを用いて貨客混載の効果について検証が可能である事を示し、検証の結果から貨客混載の効果が運用地域によって異なる可能性を示した。

今後解決すべき課題として次のことが考えられる。一つは需要の変化にリアルタイムで対応するようなアルゴリズムや数値モデルを開発する事である。日本では、都心部でのデマンドバスを中心に運行途中でもリアルタイムで予約を受け付けるシステムが広まりつつある。このような状況に対しても、貨客混載を行う方法や効果を検証する必要がある。このように、状況が変化する中でも即時に経路を導出できるようなオンラインアルゴリズムの開発は、今後の大きな課題である。

参考文献

- [1] 国土交通省 関東運輸局 交通政策部：関東運輸局管内におけるコミュニティバス・デマンド交通の実態及びバス待ち環境の先進事例に関する調査業務報告書，入手先 (<https://www.tb.mlit.go.jp/kanto/content/000166966.pdf>) (参照 2022-10-31).
- [2] 白石 悦二, 牧 幸洋, 吉武 哲信: 過疎地域における自家用有償旅客運送車両を用いた人流・物流サービス統合化に向けた事業者間調整に関する事例研究, . 交通工学論文集, 5 巻, 1 号, p. 11-19 (2019).
- [3] Kris Braekers, Katrien Ramaekers, Inneke Van Nieuwenhuysse: *The vehicle routing problem: State of the art classification and review*, Computers & Industrial Engineering, Volume 99, Pages 300-313, (2016).
- [4] Cordeau, J., & Laporte, G.: *The dial-a-ride problem (DARP): Variants, modeling issues and algorithms*, 4OR: A Quarterly Journal of Operations Research, 1, 89-101 (2003).
- [5] B. Li, D. Krushynskiy, H.A. Reijers, and T. Van Woensel: *The share-a-ride problem : people and parcels sharing taxis*, European Journal of Operational Research, 238:31-40 (2014).
- [6] Yu, V. F., S. S. Purwanti, A. P. Redi, C. C. Lu, S. Suprayogi, and P. Jewpanya: *Simulated Annealing Heuristic for the General Share-a-Ride Problem*, Engineering Optimization 50 (7): 1178-1197 (2018).
- [7] Koc C, Karaoglan I.: *A branch and cut algorithm for the vehicle routing problem with multiple use of vehicles*, Proceedings of the 41st international conference on computers & industrial engineering, Los Angeles, USA, pp 554-559 (2011).
- [8] Sin C. Ho, W.Y. Szeto, Yong-Hong Kuo, Janny M.Y. Leung, Matthew Petering, Terence W.H. Tou: *A survey of dial-a-ride problems: Literature review and recent developments*, Transportation Research Part B: Methodological, Volume 111, Pages 395-421, (2018).
- [9] Mohamed Amine Masmoudi, Kris Braekers, Malek Masmoudi, Abdelaziz Dammak: *A hybrid Genetic Algorithm for the Heterogeneous Dial-A-Ride Problem*, Computers & Operations Research, Volume 81, Pages 1-13, (2017).
- [10] Boeing, G.: *Simulated OSMnx: New Methods for Acquiring, Constructing, Analyzing, and Visualizing Complex Street Networks*, Computers, Environment and Urban Systems 65, 126-139 (2017).