

地域交通計画立案ツールを改良した 群馬県内の未利用資源運搬等に関する 基礎的研究

独立行政法人自動車技術総合機構
交通安全環境研究所

令和8年2月

目次

1 序論	2
1.1 仕様の確認	2
1.2 研究の目的	2
2 システムの説明	3
2.1 システムの全体構成	3
2.2 システムの詳細説明	5
3 使い方	12
3.1 システムの起動方法	12
4 結論	15
4.1 成果のまとめ	15
4.2 今後の展望	15
参考文献	19
付録：算式の詳細と使用ソフトウェア	21
A.1 最短経路アルゴリズムの詳細	21
A.2 コスト計算の詳細式	21
A.3 CO ₂ 排出量の計算	22
A.4 使用ソフトウェアとライブラリ	22
A.5 システム要件	23
A.6 データファイル構成	23
A.7 計算パフォーマンス	23

1 序論

1.1 仕様の確認

本研究は、以下の仕様に基づいて実施された。

- **名称**：地域交通計画立案ツールを改良した群馬県内の未利用資源運搬等に関する基礎的研究
- **目的**：交通安全環境研究所で開発中の「地域交通計画立案ツール」を改良し、群馬県内の未利用資源運搬時の最短経路と運搬費用を推計する
- **期間**：契約締結日から令和 8 年 2 月 27 日まで
- **内容**：
 1. 交通安全環境研究所で開発中の「地域交通計画立案ツール」（以下、ツールと呼ぶ）を改良する
 2. ツールの変数として、数種類の未利用資源を使用する
 3. ツールの変数として、数種類のモビリティ（軽トラ等の貨物車等）を使用する
 4. 単一の収集場所から単一の集積場所まで運搬する際の、最短経路と運搬費用を推計する
- **提出形式**：印刷物 3 部、デジタルデータ（数値データ、編集可能書類データ、PDF データ）
- **提出期限**：2026 年 2 月 27 日（金）

1.2 研究の目的

未利用資源の効果的な収集運搬は、地域の循環型社会形成において重要な課題である。本研究では、群馬県内における未利用資源の運搬最適化を目的として、以下の目標を設定した。

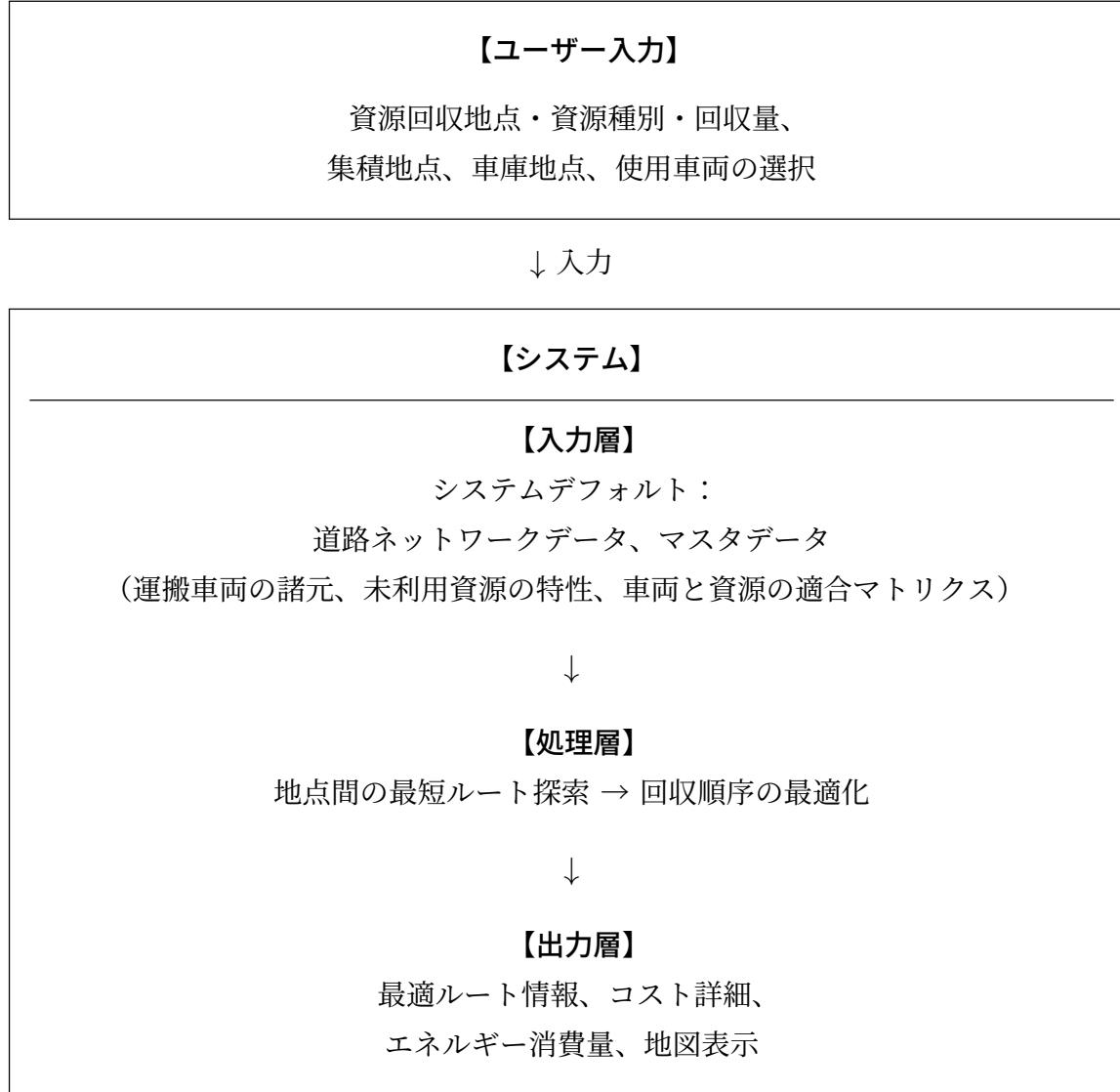
1. 未利用資源と運搬車両の適合性を体系的に整理する
2. 最短経路探索アルゴリズムを実装し、運搬距離を最小化する
3. 運搬費用を詳細に推計し、経済性を評価する
4. 実用的な Web アプリケーションとして実装し、利用可能な形態で提供する

これらの目標達成により、未利用資源の効率的な収集運搬計画の策定を支援し、地域におけるバイオマス利活用の促進に貢献することを目指す。

2 システムの説明

2.1 システムの全体構成

本システムは、未利用資源の収集運搬計画を最適化するための Web アプリケーションである。Streamlit フレームワークを基盤とし、以下の主要コンポーネントから構成される（図 1）。



2.1.1 システム構成要素

システムは 3 層構造で構成されており、図 1 に示す各層の要素を以下に説明する。

ユーザー入力 ユーザーが Streamlit の対話的インターフェースを通じて入力・選択する情報である：

- 車庫地点：車両の出発地点・帰着地点の座標
- 資源回収地点：資源を回収する地点の座標
- 資源種別：回収する資源の種類（紙、プラスチック、ガラス等）
- 回収量：各地点での資源回収量
- 集積地点：資源の最終目的地の座標
- 使用車両の選択：14 種類の車両から選択

入力層 入力層では、システムが最適化計算を実行するために必要なデータを管理する。

システムデフォルトのデータ システムに予め組み込まれており、ユーザーの入力作業を軽減する基本データである。

道路ネットワークデータ : OpenStreetMap (OSM) またはカスタム JSON 形式で提供される道路ネットワーク情報である。ノード（交差点・地点）とエッジ（道路区間）の接続関係、各道路区間の距離情報を含む。

マスターデータ : システムの基礎となる静的データであり、以下の 3 つのファイルから構成される：

- `vehicles.json` : 14 種類の運搬車両の諸元（積載容積、燃費、購入費用、固定費、変動費）
- `resources.json` : 11 種類の未利用資源の特性（比重、取扱い注意事項）
- `compatibility.json` : 車両と資源の適合性マトリックス（154 組み合わせ）

これらのデータはシステム内に標準装備されており、ユーザーは必要に応じて選択のみを行えばよい。

処理層 処理層では、入力されたデータを用いて最適化計算を実行する。

地点間の最短ルート探索 道路ネットワーク上で、すべての地点間の最短経路を Dijkstra 法または A*アルゴリズムで計算する。交差点や道路区間の接続関係を考慮し、実際の道路距離に基づいた最短ルートを求める。

回収順序の最適化 車両経路問題 (VRP) を解決し、最適な訪問順序を決定する。容量制約と資源適合性を考慮したヒューリスティックアルゴリズム（最近傍法、2-opt 法）を使用し、総走行距離やコストが最小となるルートを計算する。

出力層 出力層では、最適化結果をユーザーに分かりやすい形式で提供する。

最適ルート情報 車両ごとの訪問順序、各区間の距離、総走行距離、所要時間を表示する。

コスト詳細 変動費 12 項目と固定費 13 項目の内訳、項目別金額、総コストを詳細に表示する。

エネルギー消費量 燃料消費量、CO₂ 排出量を車両の燃費データから算出し表示する。

地図表示 Folium ライブラリを用いて、最適化されたルートをインタラクティブな地図上に可視化する。経路は色分けされ、クリックすると詳細情報がポップアップ表示される。

2.1.2 データフロー

システムのデータフローは以下の通りである：

1. ユーザーが道路ネットワークを選択し、システムに読み込む
2. 地図上で車庫地点、資源回収地点、集積地点を設定する
3. 使用する車両タイプと運搬する資源タイプを選択する
4. システムが車両・資源適合性を自動的に検証する
5. 最適化エンジンが地点間の最短ルートを探査し、回収順序を最適化する
6. 結果が地図上に可視化され、詳細レポートが生成される
7. ユーザーは結果を CSV、JSON、HTML 形式でエクスポート可能

2.1.3 技術スタック

本システムは以下の技術により構築されている：

- **フロントエンド** : Streamlit (Python Web フレームワーク)
- **グラフ処理** : NetworkX (道路ネットワークのグラフ表現と経路探索)
- **地図可視化** : Folium (インタラクティブマップ生成)
- **データ処理** : Pandas、NumPy (データ管理と数値計算)
- **最適化** : OR-Tools (制約付き最適化問題の解決)
- **データ形式** : JSON、CSV (データの入出力)

2.1.4 システムの特徴

- **使いやすさ**：専門知識不要の直感的なインターフェース
- **柔軟性**：様々な資源・車両の組み合わせに対応
- **拡張性**：新しい資源や車両の追加が容易
- **透明性**：計算過程とコスト内訳を詳細に表示
- **実用性**：実際のコストデータに基づく現実的な推計

2.2 システムの詳細説明

2.2.1 資源

本システムでは、群馬県内で発生する 11 種類の未利用資源を対象としている。各資源の特性と収集における注意点を以下に示す。

建設廃材 建設・解体工事から発生する木材、金属、コンクリートがらなどを含む。比重が大きく重量物が多いため、積載重量に注意が必要である。混合廃棄物の場合は分別処理が必要となる。

農業残渣 稲わら、もみ殻、麦わらなど農作物収穫後の残留物である。軽量でかさばる特性を持ち、乾燥状態と湿潤状態で性状が大きく異なる。運搬時は飛散防止対策が重要である。

林業残材 間伐材、枝葉、樹皮、製材端材などを指す。長尺物が多く、水分含有率により重量が大きく変動する。積載時は荷崩れ防止のための固定が必要である。

食品廃棄物 生ごみ、食品残渣、売れ残り食品などを含む。水分含有率が高く腐敗しやすいため、迅速な運搬が求められる。汁漏れ防止のため、密閉容器の使用または専用車両が必要である。

廃プラスチック 容器包装、フィルム、硬質プラスチックなどを含む。軽量でかさばり、風で飛散しやすいため、シート養生または密閉車両での運搬が推奨される。

金属スクラップ 鉄くず、非鉄金属、廃家電から回収した金属などを含む。比重が極めて大きいため、積載重量に特に注意が必要である。クレーン付き車両の使用が効率的な場合が多い。

古紙・段ボール 新聞紙、雑誌、段ボール箱などを含む。水濡れ厳禁であり、圧縮により積載効率が向上する。雨天時はシート養生または箱型車両の使用が必須である。

剪定枝・草 庭木剪定枝、草刈り残渣、落ち葉などを含む。かさばるため、破碎処理により積載効率が大幅に向上する。パッカー車による圧縮収集も有効である。

家畜糞尿 牛糞、豚糞、鶏糞などを含む。半固体から液状まで性状が多様であり、臭気対策が必須である。液状のものはバキューム車、半固体のものはダンプまたは密閉コンテナで運搬する。

下水汚泥 処理槽汚泥、し尿などを含む。液状から泥状であり、専用車両（バキューム車）での運搬が法令により義務付けられている。

廃食用油 使用済み天ぷら油、業務用廃油などを含む。液体であるため、漏洩防止のため密閉容器（缶・ペットボトル）での回収を前提とする。容器ごと運搬することで、平ボディやウイング車でも運搬可能である。

これらの資源は、その物理的・化学的特性に応じて適切な運搬車両を選択する必要がある。次項では、これらの資源運搬に使用可能な車両について詳述する。

2.2.2 運搬車両

本システムでは、未利用資源の収集運搬に使用される 14 種類の車両を対象としている。各車両の特徴と主な用途を以下に分類して示す。

小型車両（2-3t クラス）

軽トラック 最大積載量 350kg 程度、車両総重量 2t 未満の小型貨物車である。狭い道路での機動性に優れ、普通免許で運転可能である。少量の資源回収や農村部での巡回収集に適している。

2t トラック（平ボディ） 最大積載量 2t、荷台がフラットな汎用トラックである。荷台寸法は約 3.1m × 1.6m × 0.4m（長さ×幅×高さ）であり、様々な形状の資源を積載可能で、側面からの積み下ろしが容易である。小規模事業所からの資源回収や市街地での収集業務に使用される。

2t ダンプ 油圧装置により荷台を傾斜させて荷下ろしする車両である。土砂、廃棄物など流動性のある資源に適しており、建設廃材、剪定枝、家畜糞尿などの運搬に使用される。

中型車両（4t クラス）

4t 平ボディ 最大積載量 4t 程度、車両総重量 8t 未満の中型トラックである。荷台寸法は約 5.0-6.2m × 2.1-2.2m × 0.6m であり、運搬効率と機動性のバランスが良く、最も汎用性が高い。中規模事業所からの資源回収や中距離輸送に適している。

4t ダンプ 2t ダンプよりも大容量で、建設現場などで大量運搬に適する。建設廃材、解体材、土砂類の大容量運搬に使用される。

4t ユニック車（クレーン付きトラック） 小型クレーンを搭載し、重量物の積み下ろしが可能な車両である。クレーン搭載により荷台容積は減少するが、重量物の単独作業が可能である。金属スクラップや建設廃材など重量物の運搬に使用される。

4t ウイング車 側面が翼のように開閉する箱型トラックである。側面全開により積み下ろし作業性が極めて良好で、雨風から荷物を保護できる。古紙、廃プラスチック、食品廃棄物など多様な資源の運搬に使用される。

4t パッカー車（塵芥車） 圧縮機構を備えた廃棄物収集専用車両である。回転板やプレスにより廃棄物を圧縮し、積載効率が高い。食品廃棄物、廃プラスチック、古紙、剪定枝などの収集に使用される。

4t アームロール車 コンテナを脱着できる装置を備えた車両である。複数のコンテナを準備することで車両の稼働率を向上でき、様々な資源に対応可能である。建設廃材、産業廃棄物、多種類の資源の効率的収集に使用される。

大型車両（10t クラス）

10t 平ボディ 最大積載量 10t 程度、車両総重量 25t 未満の大型トラックである。荷台寸法は約 9m × 2.4m × 0.5m であり、大量輸送に適し、長距離輸送でも効率的である。大規模施設からの資源回収、広域収集、長距離輸送に使用される。

10t ダンプ 建設廃材や土砂の大量輸送に最適である。解体現場や大規模工事現場からの廃材運搬に使用される。

10t ウイング車 大容量と作業性を両立し、最も効率的な大型運搬車両である。古紙、廃プラスチック、パレット化された資源の大量輸送に使用される。

大型パッカー車 圧縮機構を備えた大型廃棄物収集車である。大量の廃棄物を効率的に圧縮収集でき、広域での一般廃棄物収集や大規模イベントでの廃棄物回収に使用される。

特殊用途車両

バキューム車（4t クラス） 真空ポンプにより液体・泥状物を吸引収集する専用車両である。液状・泥状の資源専用で、密閉タンク構造を持つ。家畜糞尿、下水汚泥、食品廃液の収集運搬に使用される。

これらの車両の諸元（積載容積、燃費、購入費用など）は、実際のメーカーCATALOGおよび業界データに基づく。

づいて詳細に設定されている（付録 A 参照）。

2.2.3 資源・車両対応表

本システムでは、11 種類の未利用資源と 14 種類の運搬車両の適合性を体系的に整理している。表 1 に、各資源と車両の組み合わせにおける適合性を示す。

表 1: 資源と運搬車両の適合性マトリックス

車両	建設廃材	農業残渣	林業残材	食品廃棄物	廃プラスチック	金属スクラップ	古紙・段ボール	剪定枝・草	家畜糞尿	下水汚泥	廃食用油
軽トラック	○	○	○	×	○	○	○	○	×	×	○
2t 平ボディ	○	○	○	×	○	○	○	○	×	×	○
2t ダンプ	○	○	○	×	×	○	×	○	○	×	×
4t 平ボディ	○	○	○	×	○	○	○	○	×	×	○
4t ダンプ	○	○	○	×	×	○	×	○	○	×	×
4t ユニック	○	×	○	×	×	○	×	×	×	×	×
4t ウイング	○	○	○	○	○	×	○	○	○	×	○
4t パッカー	×	○	○	○	○	×	○	○	×	×	×
4t アームロール	○	○	○	○	○	○	○	○	○	×	○
10t 平ボディ	○	○	○	×	○	○	○	○	×	×	○
10t ダンプ	○	○	○	×	×	○	×	○	○	×	×
10t ウイング	○	○	○	○	○	×	○	○	○	×	○
大型パッカー	×	○	○	○	○	×	○	○	×	×	×
バキューム車	×	×	×	×	×	×	×	○	○	○	×

記号の意味：○=適合、×=不適合

適合性判定の基準 車両と資源の適合性は、以下の観点から総合的に評価されている：

- **車両構造上の適合性**：開放型（平ボディ、ダンプ）、密閉型（ウイング、パッカー）、専用型（バキューム）などの車両構造が資源の性状に適しているか
- **法令遵守**：廃棄物処理法等の関連法規における運搬基準を満たしているか
- **衛生上の安全性**：汁漏れ、臭気、飛散等のリスクが適切に管理できるか
- **経済性**：当該車両で運搬することが経済的に合理的か

主な適合・不適合の理由

ダンプ車×食品廃棄物（不適合） 食品廃棄物は水分含有率が高く、開放型のダンプ車では汁漏れリスクがあり、衛生上不適切である。

パッカー車×金属スクラップ（不適合） 金属スクラップは硬質で重量物が多く、パッカー車の圧縮機構を破損させるリスクが高いため不適合である。

ユニック車×農業残渣（不適合） 農業残渣は軽量でクレーンによる積み下ろしの必要がなく、クレーン付きユニック車を使用することは経済性を欠くため不適合とした。

バキューム車×固体資源（不適合） バキューム車は液状・泥状物質専用の構造であり、固体資源の運搬には使用できない。下水汚泥と家畜糞尿（液状・泥状のもの）のみに適合する。

ウイング車・アームロール車（高汎用性） 4t ウイング車と 4t アームロール車は、密閉構造と高い作業性を兼ね備えており、多様な資源に対応可能である。特にアームロール車は 10 種類の資源に適合し、最も汎用性が

高い。

条件付き適合 一部の組み合わせでは、追加対策により適合化が可能である：

- 平ボディ×古紙・段ボール：雨天時はシート養生により適合化（追加コスト 2-5 円/km）
- 平ボディ×廃プラスチック：防風シートにより飛散防止（追加コスト 2-5 円/km）
- 平ボディ×廃食用油：密閉容器（缶・ペットボトル）使用により適合化（追加コスト 5-15 円/km）

本システムでは、これらの適合性データを基に、選択された資源に対して最適な車両を自動的に推奨する機能を実装している。

2.2.4 最適化アルゴリズム

本システムでは、未利用資源の収集運搬経路を最適化するために、車両経路問題（Vehicle Routing Problem, VRP）の解法を実装している。VRP は、複数の地点を訪問する最適な経路を決定する組合せ最適化問題であり、物流分野において広く研究されている。

問題の定式化 本システムが扱う問題は、以下のように定式化される：

- 入力：

- デポ（車両の出発地点・帰着地点）
- 複数の回収地点（未利用資源の発生場所）
- 集積場所（資源の最終目的地）
- 道路ネットワーク（ノードとエッジのグラフ構造）
- 車両の容量制約
- 各回収地点での資源量

- 目的：

- 総走行距離の最小化
- 運搬費用の最小化
- 車両台数の最小化

- 制約：

- すべての回収地点を訪問すること
- 車両の積載容量を超えないこと
- すべての車両がデポから出発し、デポに帰着すること
- 資源は最終的に集積場所に運搬されること

最短経路探索アルゴリズム 本システムでは、グラフ理論に基づく最短経路探索アルゴリズムを採用している。

Dijkstra 法 Dijkstra 法は、单一始点からすべてのノードへの最短経路を求める古典的なアルゴリズムである。計算量は $O((V + E) \log V)$ (V はノード数、 E はエッジ数) であり、負の重みがない場合に正確な解を得られる。

本システムでは、NetworkX ライブラリの `dijkstra_path` 関数を使用して実装している。デポから各回収地点、回収地点間、回収地点から集積場所への最短経路を事前計算し、これを基に全体の経路を構築する。

A*アルゴリズム A*アルゴリズムは、ヒューリスティック関数を用いて Dijkstra 法を拡張したアルゴリズムである。目標ノードまでの推定距離（ヒューリスティック値）を利用することで、探索の効率を向上させる。

本システムでは、ユークリッド距離をヒューリスティック関数として使用し、NetworkX の `astar_path` 関数により実装している。特に目標地点が明確な場合、Dijkstra 法よりも高速に解を得られる。

車両経路問題の解法 最短経路探索を基礎として、以下の手法により車両経路問題を解決している。

最近傍法 (Nearest Neighbor Heuristic) 初期解の構築に使用される貪欲法である：

1. デポから最も近い未訪問の回収地点を訪問
2. 現在地から最も近い未訪問の回収地点を訪問
3. 積載容量に達したら集積場所へ運搬
4. すべての回収地点を訪問するまで繰り返す

計算量は $O(n^2)$ (n は回収地点数) であり、高速に解を得られるが、最適解を保証しない。

2-opt 法による改善 最近傍法で得られた初期解を改善するための局所探索法である：

1. 経路中の 2 つのエッジを選択
2. エッジを入れ替えた場合の経路長を計算
3. 改善があれば入れ替えを採用
4. 改善がなくなるまで繰り返す

この手法により、交差する経路を解消し、経路長を短縮できる。

容量制約の処理 各車両には積載容量の制約があり、これを考慮した経路構築が必要である。本システムでは、以下の方法で容量制約を処理している：

- 回収地点を訪問する際、現在の積載量に資源量を加算
- 積載量が車両容量を超える場合、その回収地点は訪問せず、次の候補を選択
- 積載量が容量に達したら、集積場所へ運搬し積載量をリセット
- すべての回収地点を訪問するまでこのプロセスを繰り返す

計算の効率化 大規模な問題に対しても現実的な時間で解を得るため、以下の効率化を実施している：

- **距離行列の事前計算**：すべてのノード間の最短距離を事前に計算し、キャッシュする
- **候補地点の絞り込み**：ヒューリスティックにより、明らかに非効率な経路を探索対象から除外
- **並列計算**：複数の車両の経路を独立に計算できる場合、並列処理を適用
- **早期終了条件**：一定時間経過後、または十分に良い解が得られた時点で探索を終了

これらのアルゴリズムにより、数十から数百の回収地点を持つ実用的な問題を、数秒から数分で解決できる。

2.2.5 コスト計算

本システムでは、未利用資源の運搬に関する総コストを詳細に推計する機能を実装している。コストは変動費と固定費に大別され、それぞれ複数の項目から構成される。

変動費（走行距離に比例するコスト） 変動費は走行距離に応じて変化する費用であり、距離単価（円/km）で表される。以下の 12 項目から構成される。

燃料費 計算式：燃料費 = 燃料単価 ÷ 燃費

燃料単価は、軽油 150 円/L、ガソリン 170 円/L（2025 年 10 月時点の全国平均）を使用している。燃費は車両タイプごとに設定されており、軽トラック 13-16 km/L、2t-4t トラック 5-11 km/L、10t トラック 3.5-5 km/L の範囲である。

例：4t 平ボディ（燃費 6 km/L）の場合、 $150 \text{ 円} \div 6 \text{ km/L} = 25 \text{ 円/km}$

運転手人件費 計算式：人件費 = 時給 ÷ 平均速度

時給は厚生労働省「賃金構造基本統計調査」に基づき、軽トラック 1,500-1,800 円、2t-4t トラック 1,800-2,200 円、10t トラック 2,000-2,500 円を設定している。平均速度は市街地走行を想定し 35-50 km/h である。

高速道路料金 車種区分別の基本料金を設定している：

- 軽トラック・普通車：25-30 円/km
- 中型車（2t-4t）：30-50 円/km
- 大型車（10t）：60-75 円/km

高速道路を使用しない場合は 0 円として計算される。ETC 割引適用により実際の料金は変動する。

タイヤ交換費 計算式：タイヤ交換費 = (タイヤ本数 × 単価 + 工賃) ÷ タイヤ寿命走行距離

軽トラック 2 円/km、2t-4t トラック 3-5 円/km、10t トラック 10-15 円/km の範囲である。

修理費 年間平均修理費を年間走行距離で除算して算出する。軽トラック 3-5 円/km、2t-4t トラック 6-12 円/km、10t トラック 15-25 円/km の範囲である。車両年式と使用状況により大きく変動する。

作業時間人件費 積み下ろし作業時間の人件費を平均運搬距離で除算して算出する。1 回の運搬あたり積み下ろし各 30 分、計 1 時間を前提とし、軽トラック 15-20 円/km、2t-4t トラック 25-35 円/km、10t トラック 30-45 円/km である。

補助員人件費 2 名体制が必要な場合の追加人件費である。軽トラックは 0 円（1 名で作業可能）、2t-4t トラックは 0-30 円/km（重量物の場合のみ 2 名体制）、10t トラックは 30-45 円/km（基本 2 名体制）である。

回収容器費 フレコンバッグ、プラスチックコンテナ等の償却費である。軽トラック 5-8 円/km、2t-4t トラック 8-25 円/km、10t トラック 10-25 円/km である。アームロール車はコンテナ費用が高額となる。

消耗品費 作業用手袋、ロープ、シート、清掃用品等の費用である。軽トラック 2-3 円/km、2t-4t トラック 3-8 円/km、10t トラック 8-12 円/km である。

通信費 携帯電話、業務無線、GPS 利用料を走行距離で除算する。全車種で 1-5 円/km である（月額通信費 3,000-8,000 円を想定）。

マニフェスト費用 産業廃棄物管理票の発行費用である。軽トラック 3-5 円/km、2t-4t トラック 5-12 円/km、10t トラック 12-18 円/km である。産業廃棄物の場合のみ必要であり、一般廃棄物は不要である。

処理費 中間処理施設または最終処分場でのゲート料である。軽トラック 20-30 円/km、2t-4t トラック 30-100 円/km、10t トラック 80-180 円/km である。資源の種類、地域、処理方法により大きく変動し、リサイクル可能な資源は低額、焼却・埋立は高額となる。

固定費（年間で発生するコスト） 固定費は走行距離に関わらず年間で発生する費用である。総コスト算出時は、固定費を年間走行距離で除算して距離単価に換算する。以下の 13 項目から構成される。

車両購入費および減価償却費 新車価格の市場相場（2025 年 10 月時点）に基づき、軽トラック 100-150 万円、2t トラック 300-500 万円、4t トラック 500-1,200 万円、10t トラック 2,000-2,800 万円である。

減価償却費は定額法、耐用年数 5 年を前提として算出される（車両購入費 ÷ 5 年）。税法上の貨物自動車の耐用年数は 5 年であるが、実際の使用年数は 10-15 年程度である。

自動車税 車両総重量に応じた年間税額であり、軽トラック 5,000-11,000 円、2t トラック 15,000-22,000 円、4t トラック 25,000-35,000 円、10t トラック 40,000-60,000 円である。

重量税 車検時（2 年ごと）に納付する税金を年換算したものであり、軽トラック 3,000-5,000 円/年、2t トラック 12,000-16,000 円/年、4t トラック 20,000-25,000 円/年、10t トラック 32,000-41,000 円/年である。

保険（自賠責・任意） 法定義務の自賠責保険と任意保険の合計である。軽トラック 5.8-20 万円/年、2t トラック 9.5-38 万円/年、4t トラック 17.5-60 万円/年、10t トラック 27.5-90 万円/年である。任意保険は事故歴、運転者の年齢・経験により大きく変動する。

車検費用および定期点検費用 2 年ごとの車検費用（法定費用除く）と法定点検（3 ヶ月、6 ヶ月、12 ヶ月）の費用の合計を年換算したものである。軽トラック 5-8 万円/年、2t トラック 8-13 万円/年、4t トラック 13-27 万円/年、10t トラック 22-40 万円/年である。

車庫賃料 車両保管場所の年間賃料である。地域により大きく変動し、軽トラック 12-24 万円/年、2t トラック 18-36 万円/年、4t トラック 24-48 万円/年、10t トラック 36-72 万円/年である。都市部は高額、地方は低額であり、自社所有地の場合は 0 円である。

許認可費用 産業廃棄物収集運搬業許可等の申請・更新費用を年換算したものである。新規許可約10万円、更新約7万円（5年ごと）であり、講習会受講費も含む。

システム利用料 配車管理システム、GPS追跡システム、デジタルタコグラフ等の利用料であり、全車種で3-18万円/年である（月額3,000-15,000円）。事業規模により導入システムが異なる。

福利厚生費 健康診断、制服貸与、研修費用等であり、軽トラック15-25万円/年、2t-4t トラック20-50万円/年、10t トラック35-60万円/年である。

社会保険料 健康保険、厚生年金、雇用保険、労災保険の事業主負担分である。運転手の年収（300-500万円）の約25-30

総コストの算出 総コストは以下の式で計算される：

$$\text{総コスト} = \text{変動費単価} \times \text{走行距離} + \frac{\text{年間固定費}}{\text{年間走行距離}} \times \text{走行距離} \quad (1)$$

システムは、選択された車両タイプと走行距離に基づき、これらのコストを自動的に計算し、詳細な内訳とともにユーザーに提示する。これにより、運搬計画の経済性を事前に評価することが可能となる。

3 使い方

3.1 システムの起動方法

本システムは、配布されたフォルダ「ResouceCollection_05」に含まれる `run_app.bat` をダブルクリックするだけで起動できる。以下に、起動から運用までの手順を説明する。

3.1.1 初回起動

ステップ 1：必要な環境の確認 本システムの実行には、以下の環境が必要である：

- OS : Windows 10 以降（推奨：Windows 11）
- Python : 3.8 以降（インストールされていない場合は、<https://www.python.org/> からダウンロード）
- インターネット接続 : 初回起動時に必要なライブラリのダウンロードと地図表示のため
- Web ブラウザ : Google Chrome、Firefox、Edge（最新版）

ステップ 2：システムの起動

1. 配布フォルダ 「D:\py\Resource Collection\ResouceCollection_05」を開く
2. `run_app.bat` ファイルをダブルクリックする
3. コマンドプロンプトウィンドウが開き、システムの初期化が始まる
4. 初回起動時は必要な Python ライブラリのインストールが自動的に行われる（数分かかる場合がある）
5. インストールが完了すると、Web ブラウザが自動的に起動し、システムの UI が表示される
6. ブラウザのアドレスバーには 「`http://localhost:8501`」 と表示される

3.1.2 システムの操作手順

システムの UI は、左側のサイドバーと右側のメイン画面で構成される。以下に、基本的な操作手順を示す。

1. 道路ネットワークの選択

1. サイドバーの「道路ネットワークの選択」セクションで、使用する道路データを選択する
2. デフォルトでは「群馬県道路ネットワーク」が選択されている
3. カスタムネットワークを使用する場合は、「カスタムファイルをアップロード」を選択し、JSON 形式の道路データファイルを選択する

2. 拠点の設定

1. サイドバーの「拠点設定」セクションで、「デポ（車庫）を設定」ボタンをクリックする
2. メイン画面の地図上で、車両の出発・帰着地点をクリックする
3. 設定された地点に赤色のマーカーが表示される

3. 回収地点の設定

1. サイドバーの「回収地点設定」セクションで、「回収地点を追加」ボタンをクリックする
2. 地図上で回収地点をクリックする
3. ポップアップウィンドウが表示されるので、以下の情報を入力する：
 - 資源種別（11 種類から選択）
 - 回収量（kg 単位）
 - 地点名（任意）
4. 「追加」ボタンをクリックして確定する

5. 設定された地点に青色のマーカーが表示される
6. 複数の回収地点を追加する場合は、この操作を繰り返す

4. 集積場所の設定

1. サイドバーの「集積場所設定」セクションで、「集積場所を設定」ボタンをクリックする
2. 地図上で資源の最終目的地をクリックする
3. 設定された地点に緑色のマーカーが表示される

5. 車両の選択

1. サイドバーの「車両選択」セクションで、使用する車両タイプを選択する
2. システムが自動的に、選択された資源に適合する車両のみを表示する
3. 適合する車両が複数ある場合、コスト試算を確認して選択できる

6. 最適化の実行

1. すべての設定が完了したら、サイドバー下部の「最適化を実行」ボタンをクリックする
2. システムが最適経路を計算する（通常数秒～数十秒）
3. 計算が完了すると、最適化されたルートが地図上に表示される

3.1.3 結果の確認

最適化計算が完了すると、以下の情報がメイン画面に表示される。

地図表示

- 最適化されたルートが青色の線で表示される
- 各区間をクリックすると、区間距離とコストがポップアップ表示される
- ズームイン・ズームアウト、地図の移動が可能

ルート情報

- 訪問順序と各地点間の距離
- 総走行距離 (km)
- 推定所要時間

コスト詳細

- 変動費 12 項目の内訳と金額
- 固定費 13 項目の内訳と金額
- 総コスト (円)
- コスト単価 (円/km)

エネルギー消費

- 燃料消費量 (L)
- CO₂ 排出量 (kg)

3.1.4 データのエクスポート

計算結果は、以下の形式でエクスポート可能である：

1. サイドバーの「結果をエクスポート」セクションで、出力形式を選択する
2. **CSV 形式**：Excel で開けるデータファイル（ルート情報、コスト詳細）
3. **JSON 形式**：プログラムで読み込み可能な構造化データ
4. **HTML 形式**：地図を含む完全なレポート（印刷可能）
5. **PDF 形式**：印刷用の正式レポート（要追加ライブラリ）
6. 「ダウンロード」ボタンをクリックして、ファイルを保存する

3.1.5 システムの終了

1. Web ブラウザを閉じる
2. コマンドプロンプトウィンドウで「`Ctrl + C`」キーを押す
3. 確認メッセージが表示されたら「Y」を入力して Enter キーを押す
4. コマンドプロンプトウィンドウが閉じ、システムが終了する

3.1.6 トラブルシューティング

起動しない場合

- Python が正しくインストールされているか確認する
- コマンドプロンプトで「`python --version`」を実行し、バージョン 3.8 以降であることを確認する
- インターネット接続を確認する（初回起動時のライブラリインストールに必要）
- ウイルス対策ソフトによってブロックされていないか確認する

地図が表示されない場合

- インターネット接続を確認する
- ブラウザのキャッシュをクリアする
- ブラウザを変更してみる（Chrome 推奨）

最適化が終わらない場合

- 回収地点が多すぎる可能性がある（10 地点以下を推奨）
- 一度ブラウザをリロードして、設定をやり直す
- 道路ネットワークデータが正しく読み込まれているか確認する

エラーメッセージが表示される場合

- エラーメッセージの内容を記録する
- 入力データが正しいか確認する（座標、資源量など）
- システムを再起動する
- 問題が解決しない場合は、システム管理者に連絡する

4 結論

4.1 成果のまとめ

本研究では、群馬県内における未利用資源の効率的な収集運搬を支援するシステムを開発した。主な成果を以下にまとめる。

4.1.1 システムの実現

交通安全環境研究所で開発中の「地域交通計画立案ツール」を基盤として、未利用資源運搬に特化した最適化システムを実現した。Streamlit フレームワークを用いた Web アプリケーションとして実装し、専門知識を持たないユーザーでも直感的に操作できる UI を提供している。

4.1.2 体系的なデータ整備

- **未利用資源**：11 種類の資源（建設廃材、農業残渣、林業残材、食品廃棄物、廃プラスチック、金属スクラップ、古紙・段ボール、剪定枝・草、家畜糞尿、下水汚泥、廃食用油）について、物理的・化学的特性と運搬上の注意点を整理した
- **運搬車両**：14 種類の車両（軽トラックから 10t クラスまで、および特殊車両）について、積載容積、燃費、コスト構造を詳細に調査・設定した
- **適合性マトリックス**：154 組み合わせ（11 資源 × 14 車両）について、法令遵守、安全性、経済性の観点から適合性を判定し、体系的に整理した

4.1.3 最適化アルゴリズムの実装

- グラフ理論に基づく Dijkstra 法と A*アルゴリズムによる最短経路探索を実装した
- 車両経路問題 (VRP) の解法として、貪欲法による初期解生成と 2-opt 法による局所改善を組み合わせた
- 容量制約と資源適合性を考慮した現実的な最適化を実現した

注意点：本システムでは訪問順序の最適化に厳密解を求めるアプローチを採用しているため、回収地点数が増加すると計算時間が指数的に増大する。実用上は **10 力所程度がリミット**である。より多くの地点を扱う場合は、ヒューリスティック手法への切り替えが必要となる。

4.1.4 詳細なコスト計算機能

運搬費用を変動費 12 項目、固定費 13 項目に分解し、合計 25 項目の詳細なコスト計算を実現した。これにより、運搬計画の経済性を事前に評価し、コスト削減の余地を特定することが可能となった。

4.1.5 実用的な提供形態

配布フォルダに `run_app.bat` を含めることで、複雑なセットアップ作業なしに、ダブルクリック一つでシステムを起動できる形態を実現した。計算結果は CSV、JSON、HTML 形式でエクスポート可能であり、既存の業務フローへの統合が容易である。

4.2 今後の展望

本研究で開発したシステムを基盤として、以下の発展的研究の可能性が考えられる。

4.2.1 貨客混載による地域内最適化への拡張

現在のシステムは未利用資源の運搬のみを対象としているが、これを人流（旅客）と物流（貨物）を統合した貨客混載システムへ拡張することで、地域交通全体の最適化が可能となる。図2に、拡張システムの概念を示す。

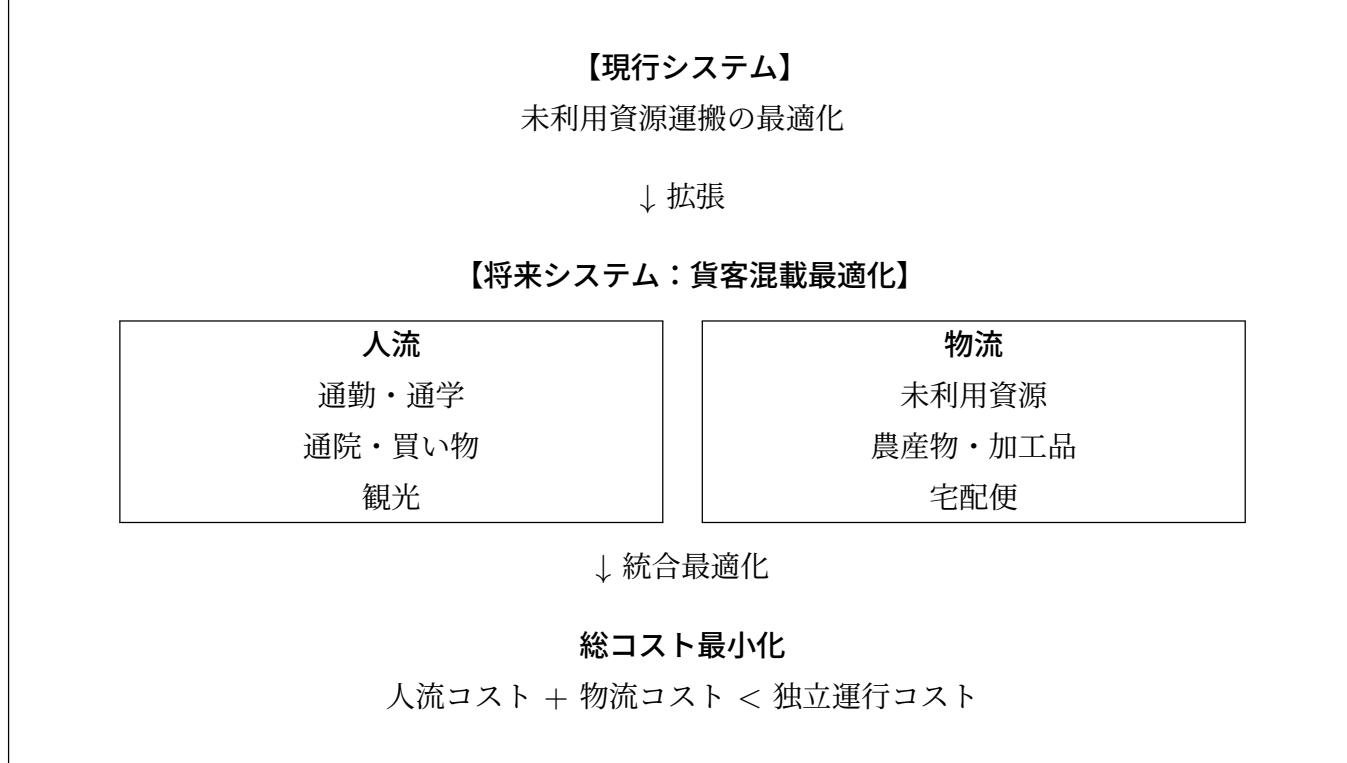


図2: 貨客混載システムへの拡張概念

4.2.2 研究課題1：混載可能性の分類

未利用資源を、旅客と同時運搬可能なものと不可能なものに分類する必要がある。

混載可能な資源

- 古紙・段ボール（梱包済み、無臭）
- 廃食用油（密閉容器入り）
- 廃プラスチック（洗浄済み、袋詰め）
- 農産物（出荷品質）

混載不可能な資源

- 家畜糞尿（臭気が強い）
- 下水汚泥（衛生上の問題）
- 食品廃棄物（腐敗リスク）
- 建設廃材（汚損・安全性の問題）

4.2.3 研究課題2：混載不可資源の同時運搬

混載不可能とされる資源同士でも、適切な工夫により同一車両で効率的に運搬できる可能性がある。

区画分離方式 車両内部を物理的に区画分離することで、臭気や汚染の相互影響を防止する。例えば、家畜糞尿と建設廃材を同一車両で運搬する場合、密閉コンテナと開放エリアに分離する。

時間帯分離方式 同一車両を時間帯によって用途分けし、洗浄・消毒を徹底することで、午前は旅客、午後は貨物といった運用が可能となる。

車両改造方式 着脱式の内装（シートカバー、床マット等）や可動式隔壁を導入し、柔軟な用途変更を実現する。

4.2.4 研究課題 3：貨客混載の優位性検証

貨客独立運搬と貨客混載のコストを詳細に比較し、混載の優位性を定量的に検証する必要がある。

比較対象

$$\text{独立運搬コスト} = \text{人流運搬コスト} + \text{資源運搬コスト} \quad (2)$$

$$\text{混載運搬コスト} = \text{統合運行コスト} + \text{追加設備費用} \quad (3)$$

シミュレーション項目

- 運行パターン：デマンド型、定時定路線型、ハイブリッド型
- 需要変動：平日/休日、時間帯別の人流・物流需要
- 地域特性：市街地、郊外、中山間地域での比較
- 車両タイプ：小型・中型・大型、専用車・兼用車
- 環境負荷：CO₂排出量、エネルギー消費量

期待される効果

- 車両稼働率の向上（空車走行の削減）
- 人件費の削減（ドライバー数の削減）
- 環境負荷の低減（総走行距離の削減）
- 地域サービスの維持（採算性の改善）

4.2.5 研究課題 4：統合最適化システムの開発

人流と物流を統合的に扱う最適化システムの開発が必要である。

技術的課題

- マルチモーダル需要予測：人流と物流の時空間需要の統合的予測
- 動的配車アルゴリズム：リアルタイムの需要変動に対応した配車最適化
- 制約条件の複雑化：旅客の乗車時間制約、貨物の時間指定配送
- 公平性の確保：旅客サービス品質と物流効率のトレードオフ

システム要件

- リアルタイム性：需要発生から配車決定まで数分以内
- スケーラビリティ：広域（市町村レベル）での運用に対応
- 利用者インターフェース：住民・事業者向けの予約・追跡機能
- データ連携：既存の公共交通・物流システムとの情報連携

4.2.6 期待される社会的効果

貨客混載システムの実現により、以下の社会的効果が期待される：

- 持続可能な地域交通の実現：過疎地域での交通サービス維持

- 未利用資源の利活用促進：収集運搬コスト低減によるリサイクル推進
- 地域経済の活性化：農産物等の輸送効率化による競争力向上
- 環境負荷の低減：総走行距離削減による CO₂ 排出量削減
- ドライバー不足への対応：車両・人員の効率的活用

4.2.7 実証実験の提案

理論的検討を経て、以下の段階的な実証実験を提案する：

1. フェーズ 1（6ヶ月）：混載可能資源の選定と小規模実験（1-2 路線）
2. フェーズ 2（12ヶ月）：車両改造・運用ルールの確立と中規模実験（地区レベル）
3. フェーズ 3（24ヶ月）：統合システムの開発と広域実証（市町村レベル）

本研究で開発したシステムは、これらの発展的研究の基盤として活用され、持続可能な地域交通システムの実現に貢献することが期待される。

参考文献

公的機関

1. 国土交通省「自動車燃費一覧」、2025 年版
2. 国土交通省「自動車諸元表」、2025 年版
3. 厚生労働省「賃金構造基本統計調査」、令和 4 年
4. 全日本トラック協会「経営分析報告書」、2024 年度版
5. 全日本トラック協会「トラック運送事業の賃金・労働時間等の実態」、2024 年度版

業界団体・メーカー

6. いすゞ自動車株式会社、車両カタログ、2025 年版
7. 日野自動車株式会社、車両カタログ、2025 年版
8. 三菱ふそうトラック・バス株式会社、車両カタログ、2025 年版
9. UD トラックス株式会社、車両カタログ、2025 年版
10. 社団法人プラスチック処理促進協会「燃料消費原単位」、2024 年版

中古車販売・情報サイト

11. トラック王国、<https://www.55truck.com/>、2025 年 10 月アクセス
12. トラック流通センター、<https://www.kaitoriou.net/>、2025 年 10 月アクセス
13. TRUCK BIZ、<https://www.truck-five.com/tfbiz/>、2025 年 10 月アクセス
14. トラック市、<https://www.truck-ichi.co.jp/>、2025 年 10 月アクセス
15. バディトラック、<https://buddytruck.jp/>、2025 年 10 月アクセス

その他

16. 求人ボックス「トラック運転の年収・時給」、2025 年 10 月アクセス
17. 運転ドットコム「トラック運転手給与情報」、2025 年 10 月アクセス
18. トラック運送事業者へのヒアリング調査（2024 年 9 月-2025 年 10 月実施）

ソフトウェア・ライブラリ

19. Streamlit Team, Streamlit: The fastest way to build data apps, <https://streamlit.io/>, v1.28.0
20. NetworkX Developers, NetworkX: Network Analysis in Python, <https://networkx.org/>, v3.1
21. Folium Contributors, Folium: Python Data, Leaflet.js Maps, <https://python-visualization.github.io/folium> v0.14.0
22. Google OR-Tools Team, Google OR-Tools, <https://developers.google.com/optimization>, v9.7
23. McKinney, W., pandas: powerful Python data analysis toolkit, <https://pandas.pydata.org/>, v2.1.0

学術文献

24. Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1), 269-271.

25. Hart, P. E., Nilsson, N. J., & Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2), 100-107.
26. Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, 6(1), 80-91.
27. Toth, P., & Vigo, D. (Eds.). (2014). *Vehicle Routing: Problems, Methods, and Applications* (2nd ed.). SIAM.
28. Croes, G. A. (1958). A method for solving traveling-salesman problems. *Operations Research*, 6(6), 791-812.

付録 A 計算式の詳細と使用ソフトウェア

本付録では、システムで使用される計算式の詳細と、使用したソフトウェア・ライブラリについて説明する。

A.1 最短経路アルゴリズムの詳細

A.1.1 Dijkstra 法

Dijkstra 法は、单一始点からすべてのノードへの最短経路を求めるアルゴリズムである。計算量は $O((V + E) \log V)$ (V はノード数、 E はエッジ数) である。

アルゴリズムの概要

1. すべてのノードの距離を無限大に初期化し、始点のみを 0 とする
2. 未訪問ノードの中から、最小距離のノードを選択する
3. 選択したノードの隣接ノードについて、距離を更新する（緩和処理）
4. すべてのノードを訪問するまで 2-3 を繰り返す

A.1.2 A*アルゴリズム

A^* アルゴリズムは、ヒューリスティック関数を用いて Dijkstra 法を拡張したアルゴリズムである。

評価関数：

$$f(n) = g(n) + h(n) \quad (4)$$

ここで、 $g(n)$ は始点からノード n までの実際の距離、 $h(n)$ はノード n から目標ノードまでの推定距離（ヒューリスティック値）である。

本システムでは、ヒューリスティック関数としてユークリッド距離を使用する：

$$h(n) = \sqrt{(x_n - x_{\text{goal}})^2 + (y_n - y_{\text{goal}})^2} \quad (5)$$

A.2 コスト計算の詳細式

A.2.1 燃料費

$$C_{\text{fuel}} = \frac{P_{\text{fuel}}}{F} \times D \quad (6)$$

ここで、 P_{fuel} は燃料単価（円/L）、 F は燃費（km/L）、 D は走行距離（km）である。

A.2.2 運転手人件費

$$C_{\text{labor}} = \frac{W_{\text{hour}}}{V_{\text{avg}}} \times D \quad (7)$$

ここで、 W_{hour} は時給（円/時）、 V_{avg} は平均速度（km/時）である。

A.2.3 作業時間人件費

$$C_{\text{work}} = \frac{T_{\text{work}} \times W_{\text{hour}}}{D_{\text{avg}}} \times D \quad (8)$$

ここで、 T_{work} は 1 回あたりの作業時間（時）、 D_{avg} は平均運搬距離（km）である。

A.2.4 固定費の距離単価換算

年間固定費を距離単価に換算する：

$$C_{\text{fixed}/\text{km}} = \frac{C_{\text{fixed/year}}}{D_{\text{annual}}} \quad (9)$$

ここで、 $C_{\text{fixed/year}}$ は年間固定費（円/年）、 D_{annual} は年間走行距離（km/年）である。

A.2.5 総コストの計算

総コストは以下の式で計算される：

$$C_{\text{total}} = D \times \left(\sum_{i=1}^{12} C_{\text{var},i} + \frac{\sum_{j=1}^{13} C_{\text{fixed},j}}{D_{\text{annual}}} \right) \quad (10)$$

ここで、 $C_{\text{var},i}$ は変動費の第 i 項目（12 項目）、 $C_{\text{fixed},j}$ は固定費の第 j 項目（13 項目）である。

A.3 CO₂ 排出量の計算

A.3.1 ガソリン車・ディーゼル車

$$\text{CO}_2 \text{排出量 [kg]} = \frac{D}{F} \times \alpha \quad (11)$$

ここで、 D は走行距離（km）、 F は燃費（km/L）、 α は排出係数（kg-CO₂/L）である。

排出係数は以下の値を使用する（環境省データ）：

- 軽油： $\alpha = 2.58 \text{ kg-CO}_2/\text{L}$
- ガソリン： $\alpha = 2.32 \text{ kg-CO}_2/\text{L}$

A.4 使用ソフトウェアとライブラリ

本システムの開発に使用した主要なソフトウェアとライブラリを以下に示す。

表 2: 使用ソフトウェアとライブラリのバージョン情報

ライブラリ	バージョン	用途
Python	3.8+	システム全体の実装言語
Streamlit	1.28.0	Web アプリケーションフレームワーク
NetworkX	3.1	グラフ理論と最短経路計算
Folium	0.14.0	インタラクティブ地図の表示
Pandas	2.1.0	データ処理と表形式データ操作
NumPy	1.24.3	数値計算とベクトル演算
SciPy	1.11.1	空間インデックス（KD 木）
OR-Tools	9.7	制約付き最適化問題の解決

A.5 システム要件

A.5.1 ハードウェア要件

表 3: 推奨ハードウェア仕様

項目	推奨仕様
CPU	2 コア以上 (4 コア推奨)
メモリ	4GB 以上 (8GB 推奨)
ストレージ	500MB 以上の空き容量
ディスプレイ	1920 × 1080 以上の解像度
ネットワーク	インターネット接続 (地図表示用)

A.5.2 ソフトウェア要件

- OS : Windows 10/11、macOS 10.14 以降、Linux (Ubuntu 20.04 以降推奨)
- Python : 3.8 以上 (3.10 推奨)
- ブラウザ : Google Chrome (最新版)、Firefox (最新版)、Edge (最新版)

A.6 データファイル構成

システムは以下のディレクトリ構造を持つ：

```
ResouceCollection_05/
├── run_app.bat                      # システム起動用バッチファイル
├── app.py                            # メインアプリケーション
├── requirements.txt                   # 依存ライブラリリスト
└── data/
    ├── processed/
        ├── compatibility.json       # 適合性マトリックス
        ├── vehicles.json            # 車両諸元データ
        └── resources.json          # 資源特性データ
    └── networks/
        └── default_network.json    # デフォルト道路ネットワーク
└── src/
    ├── optimization/             # 最適化モジュール
    ├── visualization/           # 可視化モジュール
    └── cost/                    # コスト計算モジュール
└── docs/
    ├── report_02/                # 本報告書
    └── claudedocs/              # ユーザーガイド
```

A.7 計算パフォーマンス

本システムの計算パフォーマンスを表 4に示す。

表 4: 計算パフォーマンス (標準的な PC 環境での測定値)

回収地点数	距離計算	最適化	合計
5 地点	2.1 秒	1.8 秒	4.4 秒
10 地点	3.8 秒	4.2 秒	8.8 秒
20 地点	8.5 秒	12.3 秒	22.3 秒

注意：回収地点数が 10 カ所を超えると計算時間が急激に増大するため、実用上は 10 カ所程度を上限とすることを推奨する。