

コネクテッド自動運転車の駐車：車両制御、駐車割り当て、マルチエージェントシミュレーション

Xu Shen*, Yongkeun Choi*, Alex Wong[†], Francesco Borrelli*, Scott Moura[†], Soomin Woo^{‡§}

*Mechanical Engineering, University of California, Berkeley, USA

[‡]Computer Science, University of California, Berkeley, USA

[†]Civil and Environmental Engineering, University of California, Berkeley, USA

[§]Corresponding Author, soomin.woo@berkeley.edu

概要-本論文では、コネクテッドカーとオートメーテッドカー(CAV)の車両に対する駐車効率を最適化するための新しいアプローチを紹介する。本論文では、個々のCAVのための階層的な経路計画と衝突回避機能を備えた、新しい複数車両駐車シミュレータを紹介する。このシミュレータは、低レベルの車両制御から高レベルの駐車割り当てに至るまで、駐車における主要な意思決定プロセスを捉えるように設計されており、大規模な地上車両の駐車戦略を効果的に評価することを可能にする。集合的コストを最小化するために、異なる戦略的駐車場割り当てを定式化し、比較する。提案するフレームワークは様々な目的関数を最適化するように設計されているが、自動車のエネルギー消費量と温室効果ガス排出量の削減に密接に関係するため、実験では総駐車時間を選択する。提案手法の有効性を、実世界の駐車場ダイナミクスのデータセットに対する実証評価により検証し、駐車場時間を最大43.8%大幅に短縮することを実現した。この改善は、運転自動化の相乗効果、共有インフラ状態データの活用、歩行者交通の排除、最適な駐車場割り当てのリアルタイム計算に起因する。

索引用語-自動駐車、駐車効率、コネクテッド自動運転車、車車間シミュレーション、ニューラルネットワーク、車両制御、車両接続性

I. INTRODUCTION

A. Background

本論文では、駐車場におけるコネクテッドカーとオートメーテッドカー(CAV)の駐車コストの最適化の問題を解決することを試みる。INRIX Researchの調査[1]によると、アメリカ人は駐車を探すために年間平均17時間無駄になり、その結果、無駄な時間、燃料、排出量は総額730億ドルになる。さらに、駐車場に関連する遅延によって誘発されるフラストレーションやストレスは、生活の質の低下につながり、都市住民の幸福に影響を与える可能性がある。現代生活のさまざまな側面の中で、私たちの時間と注意力を必要とするものの中で、駐車場は見過ごされがちだが、交通の効率的な流れ、ひいては生活の質に直接影響する重要な側面である。利用可能な駐車場の旋回に費やされる時間とエネルギーは、駐車場構造からの出入りの遅れによって複合化され、都市環境における全体的な移動コストのかなりの部分を構成している。

この効率性の低下は、環境的・社会的な影響だけでなく、かなりの経済的コストにもつながる。駐車場を求めてローミングする車が増えれば増えるほど、温室効果ガスの排出量と燃料消費量が増加し、大気の水質が悪化し、都市のカーボンフットプリントが増加する。

この差し迫った問題に対処するため、駐車場インフラ、資源配分、価格設定メカニズムの再構築を目的とした様々な技術の開発が進められている[2, 3, 4, 5, 6]。しかし、駐車場のインフラを変革するにはコストがかかる。さらに、価格設定と配分のソリューションは、異なる駐車場間の交通のバランスをとるのに役立ってきたが、利用可能な場所を求めて往復を続けたり、時間と資源を浪費したりすることが多い。

本論文では、駐車方法を変革するために、車両の自動化と通信技術を実験する。自律走行車は、正確な駐車操作を実行し、駐車効率をかつてないほど向上させる大きな可能性を示している。自動駐車ソリューションは、学界と産業界の両方から大きな関心を集めている。自動駐車車の複雑さについて深く掘り下げ、経路計画アルゴリズム[7, 8, 9, 10, 11]やセンサーフュージョン技術[12, 13]などの様々な側面を検証する研究努力がなされている。これらの試みは、自動駐車車の理論的基礎に関する貴重な洞察に貢献した。

同時に、テスラのようなパイオニアを含む大手自動車メーカーは、こうした理論的概念を具体的な製品へと変貌させ始めている。これらの製品は、自律走行駐車技術の限界を押し広げるだけでなく、自律走行車のフリートが一体となって都市交通に革命を起こす可能性を高めた。単一の自律走行車の検討から相互接続された車両の構想へと移行するにつれ、このパラダイムシフトの重要性が明らかになり、V2X(Vehicle-to-Everything)技術の最近の進歩がこの移行を実現可能にしている[14, 15]。駐車場の割り当てと自動運転車の制御を最適化することで、駐車場を探すコストを大幅に削減し、最終的に通勤者にとって貴重な資源を節約し、都市交通エコシステム全体を強化することができる。

CAV技術の文脈における自動駐車に関する文献は、2つの側面に分けられる。最初のファセットは、主に地域の駐車場割り当てに焦点を当てている。

駐車場の際に、地域の紛争を解決しようとする経路計画・制御技術を採用している[16, 17, 18]。これらの取り組みは、個々の車両と地域の紛争解決を優先し、車両群全体の最適化を軽視している。我々の研究では、この焦点を個々の車両にとどまらず、CAVの全車両に対する駐車効率の最適化に努めている。この貢献は、CAVフリートに合わせた効率的な駐車管理戦略の理解を深めるものである。

逆に、2つ目のファセットは、駐車場の利用を最適化し、都市部における交通渋滞を緩和することを目的としたマクロスケールに集中している。既存の文献では、車両全体を監視キューとしてモデル化することが多い[19, 20, 21, 22]。しかし、このような単純化は、個々の運動学的制約や車体の形状を見落としており、駐車操作の自動化に内在する課題に対処する能力を妨げている。タイトスペース、制約のあるレイアウト、歩行者や障害物を回避するためのナビゲーションの精度の必要性から、安全な駐車操作のための高度な制御システムが求められ、駐車プロセスに遅延が生じる。したがって、複雑な環境における駐車システムを理解し、最適化するための専用のアプローチが必要である。我々の先行研究[23]では、グリッドベースの衝突回避アプローチを導入し、小規模駐車場設定におけるフリート駐車場ヒューリスティックを探索した。しかし、このプロトタイプは、進入車両のみを持つ小規模駐車場に制約されており、より複雑な構造と多様な交通行動を持つ環境に適用可能なシステムを設計するための十分な余地が残されている。本研究では、個々の車両の複雑な運動学を学習し、ロット内の複数の車両の効率的な駐車性能を達成する、スポット割り当てのためのニューラルネットワーク戦略を考案する。

文献の相違は、専用駐車場シミュレータの開発を促している。微視的な駐車シミュレーションは、PTV VissimやCARLA[24, 25]のような既存の交通シミュレータの拡張を活用することが多い。一方、マクロな都市規模の駐車場に関する文献では、待ち行列モデルを用いたカスタムシミュレータの作成が頻繁に行われている。本研究では、独自の駐車シミュレータを導入し、CAVフリート向け自動駐車の詳細をそれぞれ柔軟に修正・分析できるようにする。注目すべきは、他のマイクロ駐車場シミュレータとは異なり、人口密度の高い駐車場シナリオにおいて、完全なCAVフリートの影響を評価するために、実際の人間の駐車場操作データを組み込んでいることである。

B. Contribution

本論文では、自動化された車両駐車と制御のための完全なシステムを設計し、以下の主要な貢献を提供する：

- 1) 階層型複数車両駐車シミュレータ：我々の知る限り、複数車両駐車シナリオをシミュレートするための包括的なフレームワークを開発したのは我々が初めてである。このシミュレータは、個々の車両の経路計画と追跡の低レベル制御、効率的な衝突回避のための中レベルアルゴリズム、車両フリートの駐車場の高レベル割り当てなど、様々なレベルでのアルゴリズムの統合を可能にする。

- 2) スポット割り当て戦略の設計と分析：駐車場のスポット割り当てのための複数の戦略を考案し、新しいニューラルネットワーク法を含む。提案するシミュレータと駐車場から記録された実際の運転データを用いて、多様な駐車場と交通シナリオにおける様々な戦略を、人間の行動ベースラインに対して分析する。
- 3) 実世界での応用に向けた洞察 1)車両の自動化と駐車場インフラとの通信により、駐車場時間を大幅に短縮できること、2)スポット割り当てのためのニューラルネットワーク戦略を提案することで、大型車両の場合、人間の運転のベースラインと比較して、駐車場時間を最大43.8%短縮できることを示す。この実証的検証は、我々のアプローチの実世界での適用可能性と利点を強調するものである。

C. 研究の概要

自動化と通信による効率的な車両駐車の問題を分析するために、単車制御、複数車両シミュレーションフレームワーク、車両割り当て戦略の3つの大まかな開発ステップを踏む。図1は、本原稿の各セクションに対応するシミュレーションフレームワークと制御戦略の概要を説明したものである。

シミュレーションの枠組みとして、自律走行車が駐車場をナビゲートし、他のエージェントと対話できるクローズドループ環境をシミュレートし、適切な車車間情報(V2I)と車車間情報(V2V)にアクセスできるようにする。我々のマルチエージェントシミュレータは、駐車場をナビゲートする際に、独立した車両の集合を管理するシステムと見なすことができる。シミュレーションは以下のように進められる：シミュレータはシミュレーションシナリオを選択し、駐車場に関する情報(スポット位置や駐車車両など)を適切に収集する。そして、その場で一連の車両を生成し、それらの車両の中心的なインフラとして機能する。生成された各車両はシミュレータから次の行動を与えられ、分散化された方法でナビゲーションを計算する。

この原稿は、フレームワークの説明においてボトムアップのアプローチをとり、セクションIIIで最初に我々の単一車両制御を説明する。車両はまず経路を計画し(セクションIII-A)、ロットの通路を通るルート(「巡航」)と通路からその場所への進入方法(「操縦」)を決定する。そして、巡航と操縦に異なるコントローラを用いた経路(セクションIII-B)が導かれる。また、V2V通信(セクションIII-C)を用いて、車両が経路をたどりながら衝突を回避する。車両は駐車場を同様に退去させるが、逆順である。

セクションIVでは、マルチピークルシミュレーションのフレームワークを示し、ピークルが走行し、相互作用するシミュレーション環境を説明する。スポット占有マネージャは、複数の車両が同じスポットに進入したり、スポット内で出車車両と衝突したりしないようにする(セクションIV-A)。特定のタイムスタンプと位置で車両を生成した後(セクションIV-B)、車両に高レベルの動作を順番に実行するように「タスクプロファイル」を作成する(セクションIV-C)。

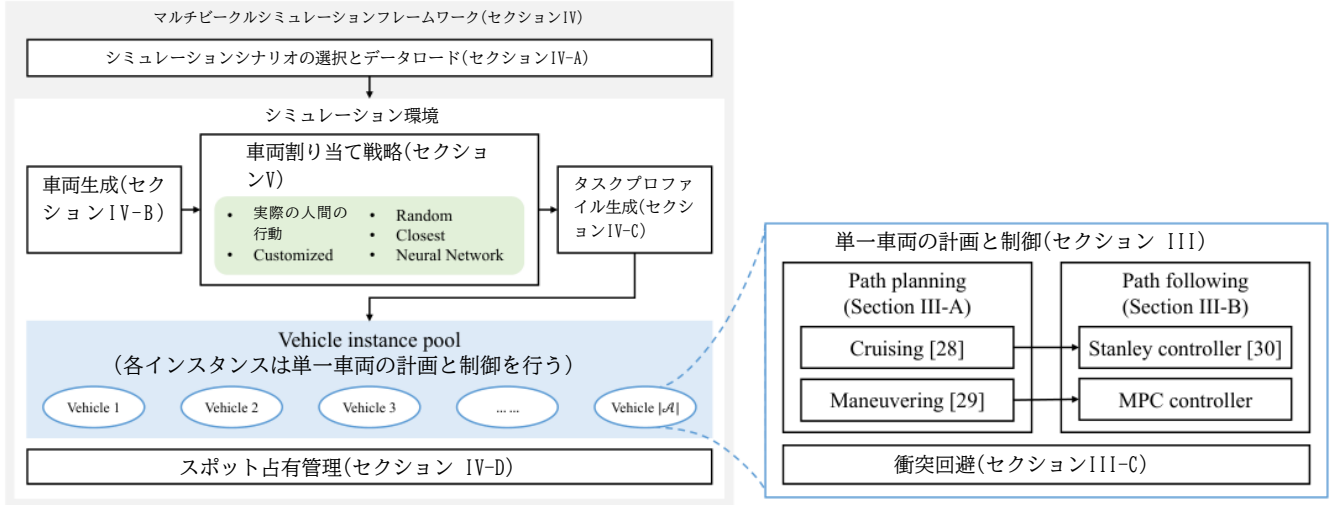


図1: シミュレータフレームワークと制御戦略の概要: 数字は本原稿のデザインを詳しく説明した部分を示す。引用しない限り、この図の各アルゴリズムとモジュールは新たに提案されたものである。

セクションVでは、駐車場を車両に割り当てるための様々な戦略を考案し、総走行時間での性能を評価する。図1には示されていないが、我々は提案されたシミュレーションフレームワークを検証し、セクションVIで割り当て戦略の性能を広範囲に分析するためにそれを使用する。

II. METHODOLOGY

本節では、自動駐車問題の背景、通信・コンピューティング技術の前提条件、人間が運転する車両を備えた実際の駐車場からの駐車データについて説明する。

A. 問題の定義

駐車場における自律走行車の駐車台数の目的関数値を最適化することを目的とし、その関数は柔軟に設計することができる。我々は、実世界の駐車場における大型フリートの移動をシミュレートし、モーションコントロール設計と異なるスポット割り当て戦略の効果を分析することで、この問題に対処する。駐車場には、駐車場 S 、静止車両 0 、移動車両 A 、および座標 $(x_e, y_e) \in \mathbb{R}^2$ を持つ入口 e が含まれる。 S はすべての駐車場の集合である。駐車場 $s \in S$ は、その中心 $(x^{[s]}, y^{[s]}) \in \mathbb{R}^2$ と幅と長さ $(w^{[s]}, l^{[s]}) \in \mathbb{R}^2$ で定義される。その $[s]$ バウンディングボックスは B_{spot} で定義される。本論文では、走行レーンに垂直な駐車場のみを扱うため、各駐車場の方位角は無視する。 0 はすべての静的車両の集合であり、ここでは障害物と呼ぶ。これらは実験時間中ずっと動かないので、静止画として扱われる。障害物 $0 \in 0$ は、その中心 $(x^{[0]}, y^{[0]}) \in \mathbb{R}^2$ 、その幅と長さ $(w^{[0]}, l^{[0]}) \in \mathbb{R}^2$ 、その方位角 $\phi^{[0]} \in \mathbb{R}$ で定義される。

A はすべての移動車両の集合である。これらは実験時間中に移動する車両である。時刻 t における車両 $i \in A$ の状態 $z^{[i]}(t)$ は、その中心位置 $(x^{[i]}(t), y^{[i]}(t)) \in \mathbb{R}^2$ 、

幅と長さ $(w^{[i]}, l^{[i]}) \in \mathbb{R}^2$ 、方位角 $\phi^{[i]}(t) \in \mathbb{R}$ 、速度 $v^{[i]}(t)$ である。そのバウンディングボックスは $B^{[i]}$ で定義される。また、各車両は駐車場に入る時間 $T_{start}^{[i]}$ を持つ。簡潔にするため、車両を不連続な集合に分類する:

A_{enter} は入口 e を起点として駐車スペースで終了する車両、 A_{exit} は駐車スペースを起点として入口 e を終了する車両、 A_{other} はスイッチングスポットやアンロードなど他のタイプの動作をする車両である。時刻 t における各車両 $i \in A$ について、決定した経路に沿って車両を駆動する加速度 $a^{[i]}(t)$ と操舵角 $\delta^{[i]}(t)$ からなる制御指令を求める。

時刻 t における車両 $i \in A$ の総コストを $f^{[i]}(t)$ とすると、目的は全車両の総コストを最小化することである。

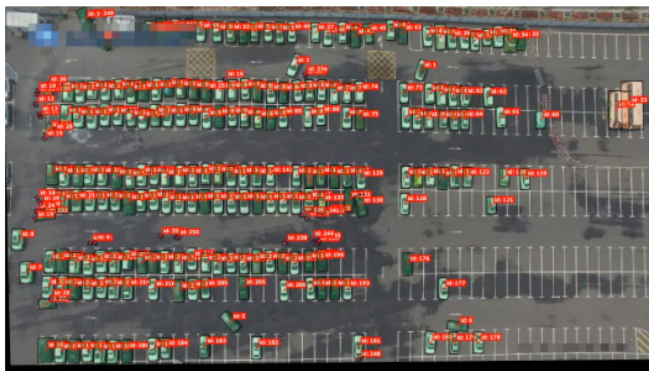
$$\sum_{i \in A_{enter}} \int_{T_{start}^{[i]}}^{T_{end}^{[i]}} f^{[i]}(t) dt, \quad (1)$$

[i] ここで、 $T_{end}^{[i]}$ は車両 i が操縦を終了し、駐車場で停止する時間である。一般的なコスト $f^{[i]}(t)$ は、時間、乗客の快適性、エネルギー消費、排出量など、駐車効率の指標を反映することができるが、これらに限定されない。本稿の残りの部分では、時間は駐車効率の素直な表現であり、エネルギー消費と相関があるため、時間、すなわち $f^{[i]}(t) = 1$ に注目する[1]。

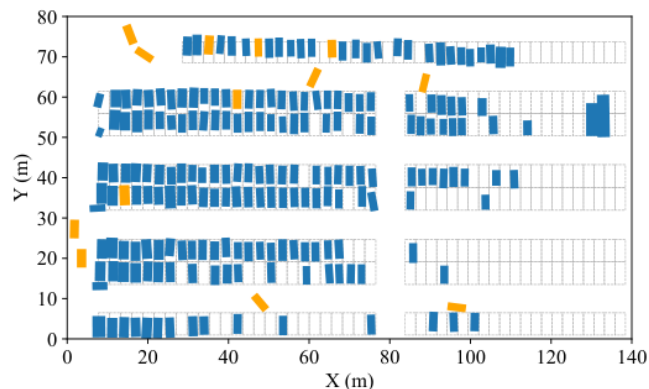
駐車場では車両が複雑な挙動や相互作用をするため、この最適化問題に対する厳密な解法は困難である。そこで、直接解法ではなく、車両運動制御装置、衝突回避ルール、駐車場割り当てアルゴリズムを設計し、この目的値を減少させるシミュレーション研究を行う。

B. Assumptions

駐車場コントローラを開発するために、通信と情報の利用可能性について以下のように仮定する: まず、



(a) 注釈付きビデオからのサンプルフレーム



(b) 処理されたデータの可視化

図2: ドラゴンレイクパーキング(DLP)データセット[26, 27].

Vehicle-to-Infrastructure(V2I)通信を仮定し、各車両は走行レーンや駐車場の位置を含む駐車場の地図に関する情報を持っている。第二に、車両間(V2V)通信を想定し、各車両はロット内の他の車両に関する情報(静的および移動の両方)と、自律航法に関する情報(加速、ブレーキ、基準軌道)を持っている。V2Vは、情報を直接渡す、駐車場の中央インフラを利用する、センサーや予測モデルから情報を収集するなど、様々な方法で実現することができる。特に車両数が増加するにつれて計算負荷が大きくなるため、車両の運動制御コマンドに対する集中コンピューティングは想定していない。さらに、ダイナミックな駐車場の変更に対してロバストでない可能性があり、駐車場に進入する予期せぬ車両が、すべての車両の最適経路を劇的に変更する可能性がある。したがって、各車両は分散化された方法で独立して独自の軌道を計算し、中央のインフラは情報ハブとしてのみ機能すると仮定する。

C. Data

マルチエージェントシミュレーションの開発では、既存のドラゴンレイクパーキング(DLP)¹データセット[26, 27]と連携し、駐車場マップ、人間主導の車両の軌跡、可視化機能を利用する。

¹<https://sites.google.com/berkeley.edu/dlp-dataset>

現実世界のデータを使用することで、駐車場の形状、密度、各駐車場における車両の位置など、現実的な駐車場環境をモデル化する上で極めて重要な利点が得られる。また、ペースラインを人間が運転する駐車行動でモデル化することができ、それに対して我々の提案するフレームワークを評価することができる。

DLPデータセットは、世界最大の駐車場公開データセットの一つである。3.5時間かけて収集された、混雑した駐車場内の車両、歩行者、自転車の注釈付きデータが含まれている。図2aは、アノテーションされたデータのサンプルである。シミュレーションでは、ロットマップと人間駆動車の軌跡のみを利用する。

DLPデータには、ロットの寸法、各駐車場の位置と向き、ロットの走行レーン(「通路」)を定義するウェイポイントのセットなど、ドラゴンレイク駐車場のマップに関する情報が含まれている。人間が運転する車両の行動は、時間の経過とともに一連のフレームとして記録される。各フレームには、その時刻にロット内の全車両のサイズ、位置、方位角、速度、加速度が含まれる。車両を静的障害物と非静的車両に分類するためにデータを処理した。図2bは、処理されたデータのサンプルフレームを視覚化したもので、駐車場は黒い破線で描かれ、障害物は青い長方形で描かれ、車両は黄色の長方形で描かれている。

III. シングル・バイク・プランニングと制御設計

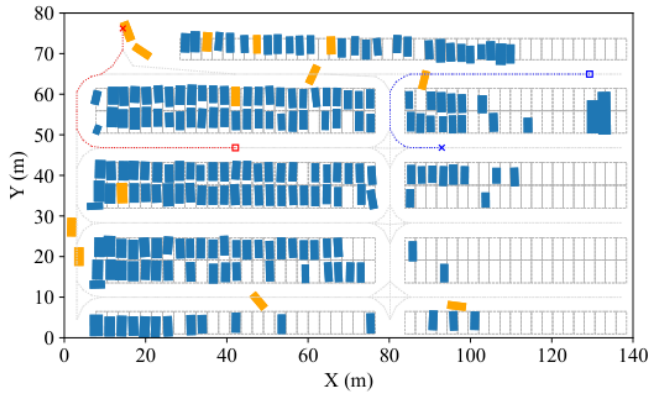
本節では、単一車両が巡航(すなわち、ロット内のある場所から別の場所への移動)し、駐車場への出入りを操作するための効率的な制御方法について述べる。セクションIII-Aでは、ロットを通り抜ける効率的な経路を計画し、セクションIII-Bでは、フィードバック制御器を用いて計画した経路をたどる。セクションIII-Cでは、車両とその周囲に基づくルールによる衝突回避を示す。

A. 経路計画

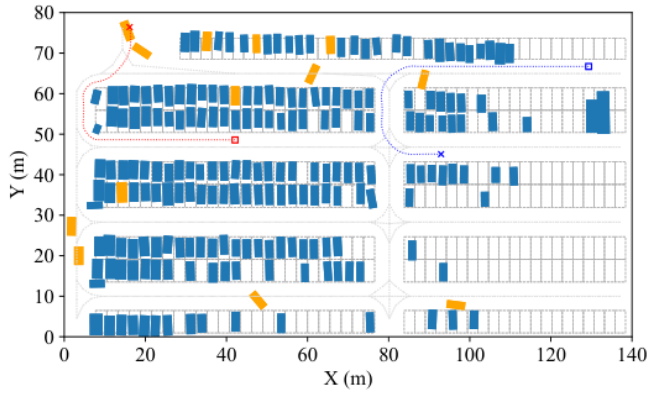
巡航経路を計画するために、A*探索アルゴリズム[28]を使用する。駐車場への移動経路の出入りを計画するために、最適化ベースの衝突回避(OBCA)[29]を使用する。以下、2つのプランニングアルゴリズムについて簡単に説明する。

1) クルージング: 巡航のA*アルゴリズムでは、ロットの通路上のウェイポイントを使用する。図3aはDLPデータの2つのサンプルA*パスを示しており、赤のパスはスポットから入り口へ、青のパスはスポット間に向かう。1つの複雑さは、同じ通路を走行する2台の車両が同じ方向を走行すると衝突するため、車両が通路の中央を直接走行できないことである。そこで、長さ N_{ref} の経路 p (すなわち図3aの経路)を探索した後、車線中央からオフセット $\Delta_{off} \in R$ を適用し、2方向に走行する車両が干渉しないようにし、最終的に車両がたどる基準経路 p^* とする。各点 $(x_{ref}, y_{ref}, \psi_{ref}) \in p$ に対して、 p^* の対応点は(2)で与えられる。図3aのパスを $\Delta_{off} = 1.75$ で可視化したものを図3bに示す。

$$(x_{ref} - \sin(\psi_{ref})\Delta_{off}, y_{ref} + \cos(\psi_{ref})\Delta_{off}, \psi_{ref}) \quad (2)$$



(a) サンプルA*の軌跡



(b) オフセット付きサンプルA*軌道

図3: 巡航のための経路計画四角いマーカーは開始位置を、十字のマーカーは終了位置を示す。

2) 操縦: 車両が意図した場所に移動し、駐車したいと望むと、意図的に通路を出てその場所に入る必要があり、道路沿いの既存のウェイポイントでは達成できない作業である。その代わりに、車両がA*端子からスポットにどのように移動するかをモデル化する新しい経路を作成しなければならない。一般的に、駐車場周辺のスペースは狭いので、車両が追従するためには、動的に実行可能で衝突のない経路が必要である。

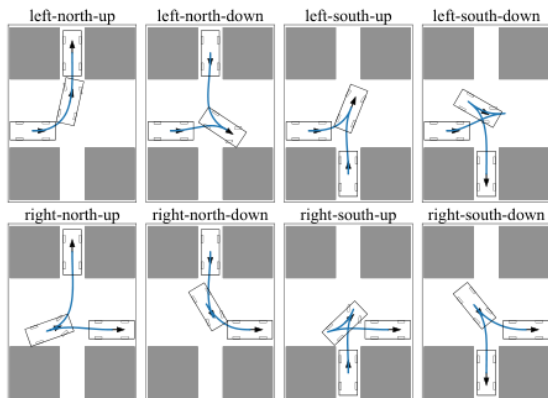


図4: パーキングマヌーバの軌跡

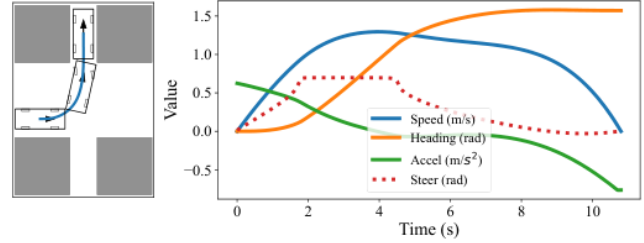


図5: 「左-右-上」マヌーバの軌跡とそれに対応する状態入力プロファイルの例。

あるロットのすべての駐車場は同じような向きであるため、そのスポットに入るか出るかの経路はあらかじめ決めておくことができ、そのスポットとは無関係である。つまり、各駐車場についてオフラインで一度だけ計算することができる。DLPデータに対して、図4に示すような8組の経路を計算し、これを「パーキングマヌーバ」と呼ぶ。各マヌーバは3つの基準で構築されている。まず、スタート地点(左または右)で、車両がどの側で操縦を開始するかを考える。次に、駐車場への方角を考慮して、選ばれた場所(北または南)である。第三は、マヌーバ終了時に車両がどの方向に向かなければならないかを考慮した最終方位(上か下か)である。

与えられたマヌーバの経路を計算するために、マヌーバ開始時と終了時の車両の状態をそれぞれ z_0 と z_f と定義し、車両ダイナミクスとして運動学的自転車モデル $\dot{z} = f(z, u)$ を用いる:

$$\dot{z} = f(z, u) := \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v \cos(\psi) \\ v \sin(\psi) \\ \frac{v}{l_{wb}} \tan(\delta) \\ a \end{bmatrix}, \quad u = \begin{bmatrix} a \\ \delta \end{bmatrix}. \quad (3)$$

次に、以下の軌道計画問題を解くことにより、最適経路とそれに対応する制御指令を求める:

$$\min_{z, u, T} J = \int_{t=0}^T c(z(t), u(t)) dt \quad (4a)$$

$$\text{s.t. } \dot{z}(t) = f(z(t), u(t)), \quad (4a)$$

$$z(t) \in \mathcal{Z}, u(t) \in \mathcal{U}, \quad (4b)$$

$$z(0) = z_0, z(T) = z_f, \quad (4c)$$

$$\text{dist}(\mathbb{B}(z(t)), \mathbb{B}_{\text{obs}}^{[o]}) \geq d_{\min}, \forall o, \quad (4d)$$

ここで、状態 z と入力 u は動作限界 \mathcal{Z} と \mathcal{U} の下で制約される。時刻 t における車体を $B(z(t))$ とし、全ての障害物 $o \in \mathcal{O}$ から安全距離 d_{\min} を離すよう求める。ステージコスト $c(-, -)$ は、作動量、エネルギー消費量、時間消費量を符号化することができる。最適解 $\{z^*, u^*\}$ は与えられたマヌーバの最適経路である。図5は、サンプルの軌跡とそれに対応する最適入力を示している。

各駐車操作は、車両が左側からスポットに近づいていると仮定しているにもかかわらず、これら8つの操作は、可能性のある駐車シナリオ²に対応するのに十分である。

車両が右から接近している場合は、既存の操作の1つをミラーリングするだけである。また、ある車両が通路にその場を抜ける軌跡を計算するには、逆方向以外は同じ駐車操作でその場に入ることができる。この時点から、通路からスポットに入る行為を「インパーキング」、通路にスポットから出る行為を「アンパーキング」と呼ぶことにする。両者とも操縦の傘に収まる。

B. 経路追従

ウェイポイントのセットを定義して車両の経路を計画した後、その経路をたどるアルゴリズムとして、巡航経路のスタンレーコントローラと操縦経路のMPCコントローラの2つを使用する。

1) スタンレーコントローラ: A*探索によって返されたような、ウェイポイントの集合によって定義される長い軌道をたどる方法はたくさんある。最良のコントローラの1つは、リアルタイムで急速に変化する軌道を追跡するために構築されたスタンレーコントローラ[30]である。ウェイポイントのセットが与えられると、車両の加速度 a と操舵角 δ を決定するための制御則を定義する:

$$a(t) = k_p(v(t) - v_{ref}), \quad (5a)$$

$$\delta(t) = \psi(t) + \arctan \frac{ke_{lat}(t)}{v(t)}, \quad (5b)$$

ここで、各車両について、 v は速度、 v_{ref} は基準速度、 e_{lat} は横方向の変位誤差、 ψ は基準軌道の最も近いセグメントに対する車両の方位誤差である。パラメータは、車両がブレーキしていない場合は $k=0.5$ 、 $k_p=1$ 、車両がブレーキしている場合は $k_p=5$ とした。また、最大入力は $|a| \leq 10 \text{ m/s}^2$ と $|\delta| \leq 40^\circ$ である。各車両は、式(2)で定義されるオフセット基準軌道に従って、駐車場の通路をナビゲートするためにこのコントローラを使用する。

2) モデル予測制御: スタンレーコントローラは、経路追従のための効率的な制御を可能にするが、衝突回避を扱わない。通路を走行している間、この問題はセクションII-Cで説明したような衝突回避アルゴリズムによって解決できる。マヌーバについては、静的な障害物(駐車車両)が存在するため、より正確な制御が望まれる。計算コストを合理的に保ちながら、駐車と退出操作の経路追従を最適化するために、モデル予測制御(MPC)を使用する。MPCは、限られた時間地平にわたって常に最適制御問題を解くことにより、制約を満たしながら、可能な限り近い基準軌道をたどることを目的とする。我々の問題では、操縦車両のMPC軌道は対応するオフライン駐車操縦から取られ、その制約は他の車両を避けるためである。MPCの定式化は以下の通りである:

DLP データの場合、駐車スポットによっては、近傍のウェイポイントの性質上、出入りに異なる駐車操作が必要になる場合がある。簡単のため、これらの場所に車両を駐車させない。

$$\begin{aligned} \min_{\mathbf{z}, \mathbf{u}} \quad & J = \sum_{k=0}^{N-1} (x_{k|t} - x_{ref}(k|t))^2 + (y_{k|t} - y_{ref}(k|t))^2 \\ \text{s.t.} \quad & z_{k+1|t} = z_{k|t} + f(z_{k|t}, u_{k|t})\Delta_t, \\ & z_{k|t} \in \mathcal{Z}, u_{k|t} \in \mathcal{U}, \\ & z_{0|t} = z(t), \\ & \text{dist}(\mathbb{B}(z_{k|t}), \mathbb{B}_{obs}^{[o]}) \geq d_{min}, \forall o, \\ & \forall k = 0, \dots, N-1, \end{aligned} \quad \begin{aligned} (6a) \\ (6b) \\ (6c) \\ (6d) \end{aligned}$$

ここで、 t は駐車操作開始からの現在の時間、 f は運動学的自転車モデル(3)、 Δ_t は時間ステップサイズ、状態 $z = [x, y, \phi, v]^T$ と入力 u は操作限界 \mathcal{Z} と \mathcal{U} の下で制約される。前回と同様に、時刻 k における車体を $B(z_{k|t})$ とし、全ての障害物 $o \in \mathcal{O}$ から安全距離 d_{min} だけ離すように求める。ステージコストは駐車操作の対応部分までの二乗距離のみに依存し、 z で与えられる。 $x_{ref}, y_{ref}, \phi_{ref}, v_{ref}$ 。衝突回避制約は式(6d)のように記述され、 $\text{dist}(-, -)$ は2つの凸多角形間の符号付き距離を記述する。微分可能な再定式化は付録Aに含まれている。式(6)を解いた後、最初の入力 $u_{0|t}^*$ を車両に適用する。

限られた時間軸である1秒を選択したにもかかわらず、駐車MPC問題を解くために必要な計算量は、スタンレーコントローラよりも大幅に多い。これは現実の分散型ハードウェア実験では問題にならないが、シングルマシンシミュレーションでは減速の原因となる。このことを考慮すると、大規模なシングルマシンシミュレーションにおいて、オフラインで計画された駐車操作に沿って車両を直接テレポートすることも許容される。オフラインマヌーバは運動学的に実行可能で衝突のないものであることが計算されるため、テレポテーションは現実的なシミュレーション結果を導くことができる。

C. 衝突回避

駐車場における衝突回避は、標準的な道路操作とは異なることが多い。交通信号や停止標識がなく、車両の進行方向は想定外の操作であり標準化されていない。したがって、衝突事故の回避はより複雑になる可能性がある。我々は、スムーズで高速な交通流を促進するために、分散化されたルールベースの衝突回避アルゴリズムを考案した。アルゴリズムは以下の原則で動作する:

- 1) 車両は、当面、他の車両と衝突する可能性がある場合、降伏を検討する;
- 2) At an intersection, the vehicle that is further into the intersection region has higher priority;
- 3) 通路を巡航する車両は、駐車操作を開始する予定の車両よりも優先順位が高い。逆に、ある車両がその場の操縦の間にいる場合、その車両はその場の操縦の間にいる場合、その車両はその周囲の車両よりも高い優先順位を持つことになる。

したがって、車両がその場で運転するかしないかを決定する前に、その操縦をブロックする車両がなくなるまで、

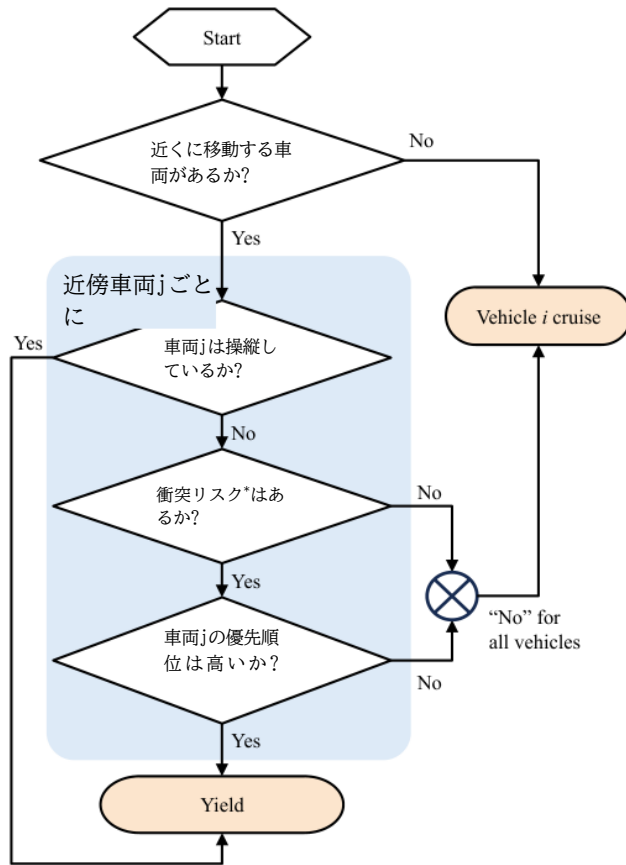


図6: 巡航時の車両 $i \in A$ の衝突回避決定ルール * 衝突リスクはアルゴリズム1で評価する ** 優先度の計算については図14、アルゴリズム2は付録Bで詳しく説明する。

常に待機することになる。一度開始すれば、完了するまでマヌーバは中断されない。巡航中の車両については、衝突回避決定ルールを図6に示す。他の車両に譲歩した場合は、図7のルールに従って、譲歩を続けるか巡航を開始するかを決定する。車両が巡航しているとき、 $[i]$ は式(5)の $v_{ref} = v_{cruise}$, $k_p = 1$ とすることで制御される。逆に、 $v_{ref} = 0$, $k_p = 5$ とすることで降伏が得られる。

IV. 複数車両のシミュレーションフレームワーク

単一車両制御フレームワークを開発した後、車両群全体をテストするための環境を作成する。これは、中央の占有管理者を維持し、車両が駐車場で矛盾した選択をしないようにすること(セクションIV-A)、駐車場で車両を産卵すること(セクションIV-B)、一連の高レベル行動、すなわち「タスクプロファイル」を割り当てること(セクションIV-C)を含む。

A. スポット占有管理

同じ駐車場まで走行することで車両が衝突したりデッドロックに入らないようにするため、集中型インフラは占有データベース $o = \{0, 1\}^{|S|}$ を維持する。我々はルールに従う

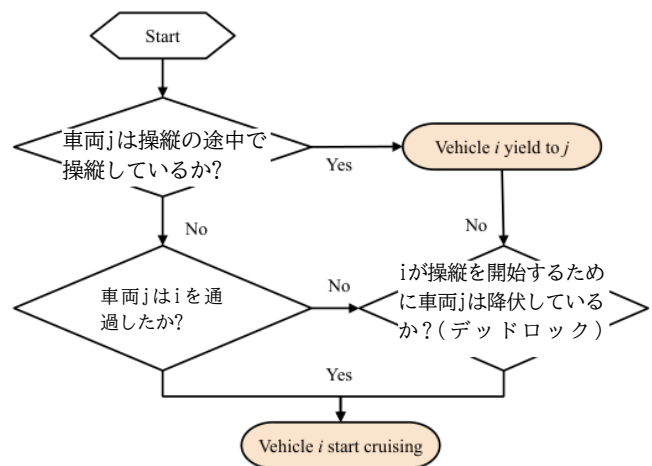


図7: 車両 $i \in A$ が $j \in A/\{i\}$ に対して降伏しているときの衝突回避決定ルール* 「jが通過したi」の基準については、付録の図15とアルゴリズム3で詳しく説明する。

シミュレーションの間中、スポットを再利用することは可能であるが、車両が同じスポットに割り当てられないようにデータベースを更新するために、以下を参照してください:

- 1) As the initial state of the simulation, for each $s \in S$, $\mathbb{O}^{[s]} = 1$ if there is an obstacle $o \in \mathcal{O}$ such that $(x^{[o]}, y^{[o]}) \in \mathbb{B}_{spot}^{[s]}$ or an existing vehicle $i \in \mathcal{A}$ such that $(x^{[i]}, y^{[i]}) \in \mathbb{B}_{spot}^{[s]}$;
- 2) When a vehicle $i \in \mathcal{A}_{enter}$ is assigned to park in spot $s \in S$, then $\mathbb{O}^{[s]}$ is set to 1;
- 3) When a vehicle $i \in \mathcal{A}_{enter}$ finishes its maneuver out of the spot s , $\mathbb{O}^{[s]}$ is set to 0.

B. Vehicle Spawning

シミュレーション環境における産卵機には2つのアプローチがある:

- 1) 事前に定義されたタイムスタンプと位置を持つ;
- 2) 入口または空き駐車場のいずれかに、ランダムな時間間隔で配置する。

最初のアプローチでは、事前に定義された時間と位置は、DLPデータセットまたは任意の手動設計から得ることができる。第二のランダムアプローチでは、平均 $\lambda_{arrival}$ を持つ指数分布を用いて、各車両到着間の時間をモデル化する。これは、到着率とロット内の渋滞レベルを制御できるようにしながら、ある程度のランダム性で車両を生成する。時刻 $T_{start}^{[i]}$ に車両 $i \in \mathcal{A}_{enter}$ が産卵されると、次の進入車両 $j \in \mathcal{A}_{enter}$ は $T_{start}^{[j]} = T_{start}^{[i]} + T_{interval}$ で産卵される。出口車両もこの分布で産卵される。場所的には、進入車両は入口 e で産卵され、退出車両はラビダ時刻 $T_{start}^{[i]}$ に他の車両が入るのを塞いでいるため、車両 i が生成できない場合、車両 i は待ち行列に配置される。入口が明らかになると、待ち行列の最初の車両 $[i]$ が生成され、 T_{start} は現在時刻に変更される。

C. タスクプロファイル

運轉行動の高レベル抽象化のシーケンスを記述するために、「タスクプロファイル」を定義する。各車両 $p^{[i]}$, $i \in A$ のタスクプロファイルは、IDLE, CRUISE, PARK, UNPARK を含むタスク $\tau^{[i]}$ のリストを含む。例えば、ロットに入り、公園に入り、待ち、そしてロットから出る車両は、表Iに示すように、各タスクに適切なパラメータを持つタスクプロファイル[CRUISE, PARK, IDLE, UNPARK, CRUISE]を持つことになる。各車両は、シミュレーション中にそのタスクプロファイルを順番に実行する。タスクプロファイルの定義により、車両は運轉の異なる段階に対して異なる計画・制御アルゴリズムを効果的に切り替えることができる。

V. 車両アライメント戦略

本節では、駐車場の混雑を回避し、駐車場の効率をさらに向上させるために、駐車場を車両群に割り当てる様々な戦略を提案する。入口に駐車場を配置することで、最適な駐車場を持つことのメリットを享受できるだけでなく、目的地に到着するまでの最適な経路をたどることができる。複数のタイプのスポット割り当てアルゴリズムを検証する。戦略には2つのベースラインが含まれる。まず、駐車車両の現状維持のための「人間選択」戦略である。第二に、カスタマイズされたソリューション戦略であり、手で解かれ、提案された戦略の目標性能を表す(ただし、自動的に解くのは非常に難しい)。なお、非占有スポットの集合を S_{empty} $= \{s \mid s \in S, s^s = 0\}$ と定義する。

A. 人間の選択(現状維持ベースライン)

このベースライン割り当て戦略は、現状を反映して、DLPデータセットの人間のドライバーによって選択された駐車場を再現する。駐車場が選択され、人間のドライバーによって運轉される場合、自動運轉制御を適用する。シミュレーションで車両がロットに入ったとき、その車両がデータセットに駐車していたスポットを割り当てる。この場合、歩行者はシミュレートされないことに注意してください。

B. カスタマイズ解法(アルゴリズムベースライン)

このカスタマイズされたソリューション戦略は、人間の駐車データが利用できないシナリオにおいて、他の戦略のパフォーマンスを比較するためのアルゴリズムベースラインとして機能する。この戦略では、スポット割り当ての順序は、限られたシナリオのセットに対して手でカスタマイズされる。Shenら[23]が明らかにしたように、車両を適切な距離で間隔をあけることで、車両同士をブロックすることなく、車両が並行して駐車操作を行うことができ、車両全体の時間を節約することができる。したがって、Customized Solution戦略では、連続する2つの割り当てごとに、入口に近接したまま、互いに対して十分な距離 Δp を持つことを確認する。3列の空き駐車場と $|A_{enter}| = 27$ が与えられたとき、図8は $\Delta p = 5$ のカスタマイズ解戦略を示している。

この戦略は、異なる駐車場構成に適応するために人間の専門知識に依存するため、自動化されたアルゴリズムに変換することは困難であることに注意してください。さらに、このヒューリスティックは、

割り当てパターンを混乱させる既存の車がある場合、スポット選択がすぐに非自明になる可能性があるため、部分的に占有された駐車場の自動化が特に困難である。

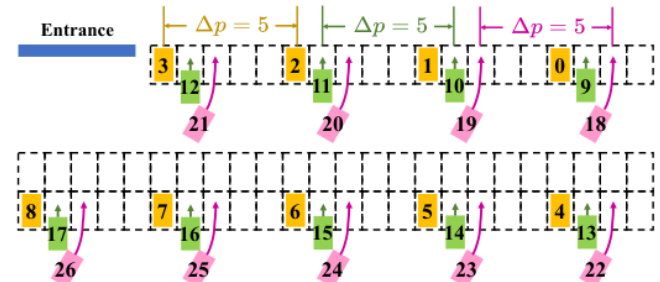


図8: カスタマイズされたソリューションの割り当て戦略の説明図。駐車場にはダッシュ線が付されている。車両は、割り当てられた駐車場に向けられた長方形の実線ボックスで表される。数字は到着順を表し、すなわち数字0は最初の車両、数字26は最後の車両である。オレンジ色、緑色、紫色で示したように、すべての連続した割り当てに対して、最小距離を $\Delta p = 5$ に維持する。

C. Random

すべての非占有スポットの集合 S_{empty} からランダムにスポットが割り当てられる。整数 i を1からロットの空点数までの離散一様分布($|S_{empty}|$)からランダムにサンプリングし、 S_{empty} の i 番目のスポットを使用する。

D. Closest

このポリシーは、到着した車両を、入口から最も近い未占有の場所に割り当てる、すなわち。

$$\hat{s} = \underset{s \in S_{empty}}{\operatorname{argmin}} (x^{[s]} - x_e)^2 + (y^{[s]} - y_e)^2, \quad (7)$$

ここで、 (x_e, y_e) は入口の座標である。

E. Neural Network

車両の駐車時間は、駐車場の位置やリアルタイムの交通量など、複数の要因に影響される可能性がある。この戦略は、この複雑なマッピング $f_\theta(-): \mathbb{R}^7 \rightarrow \mathbb{R}$ をパラメータ θ を持つニューラルネットワークで近似し、最も少ない時間しか必要としないスポットに車両を割り当てることを目的とする。

各候補スポット $s \in S_{empty}$ に対して、ネットワークへの入力

$$X^{[s]} \in \mathbb{R}^K = \left[x^{[s]}, y^{[s]}, L^{[s]}, N_{path}^{[s]}, N_{neighbor}^{[s]}, \lambda_{enter}, N_{queue} \right]^T, \quad (8)$$

ここで、 $x^{[s]}, y^{[s]}$ はそれぞれスポット s の (x, y) 座標、 $L^{[s]}$ は入口からスポット s までの計画基準経路 p の長さ、 $[s] [s] N_{path}, N_{neighbor}$ は経路 p に沿ってスポット s を移動する車両の数(車両を減速させる可能性のある所定のスポットへの交通情報を追加する可能性がある)、

表1: 各タスクプロファイルに必要なパラメータ

| Task name | 最大走行速度, v | クルーズするスポットと公園に s_{target} | Coordinates to cruise to, (x_{target}, y_{target}) | Idle task duration, ΔT_{idle} | Time to end idle task, T_{next} |
|-----------|-------------|-----------------------------|--|---------------------------------------|-----------------------------------|
| IDLE | n/a | n/a | n/a | optional | optional |
| CRUISE | required | optional | optional | n/a | n/a |
| PARK | n/a | required | n/a | n/a | n/a |
| UNPARK | n/a | required | n/a | n/a | n/a |

λ_{enter} は平均車両到着率、 N_{queue} は入口から待ち行列を作成した車両数である。ニューラルネットワークの出力 $f_{\theta}(X^{[s]}) \in \mathbb{R}$ はスポット s に駐車する時間を表す。ニューラルネットワークの目的は駐車時間をできるだけ正確に予測することなので、損失関数は次のように定義される。

$$\ell = \frac{1}{|D|} \sum_{t, X \in D} (t - f_{\theta}(X))^2, \quad (9)$$

ここで、データセット D は、第III節と第IV節で説明したシミュレーションを用いて収集され、各シミュレーションのスポット s 、 λ_{enter} 、 λ_{exit} 、 λ_{enter} を変更する。

ニューラルネットワークは、2つの隠れ層を持つ単純な多層パーセプトロン(MLP)である。隠れ層のサイズは84と10で、活性化にはReLUが使われる。最適化はAdamを用い、学習率0.01で行う。学習後の最適パラメータ θ^* により、スポット割り当て戦略は以下のように与えられる:

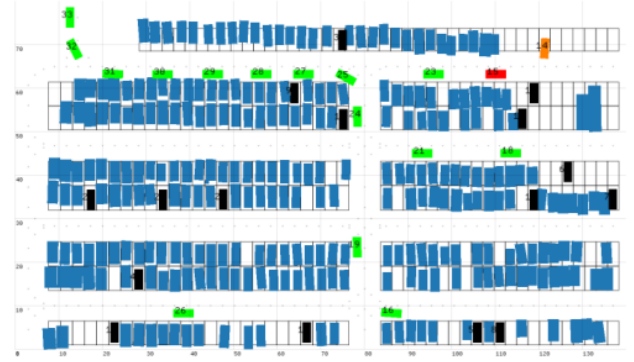
$$\hat{s} = \underset{s \in S_{empty}}{\operatorname{argmin}} f_{\theta^*}(s). \quad (10)$$

VI. マーケティング・アライメント戦略の評価

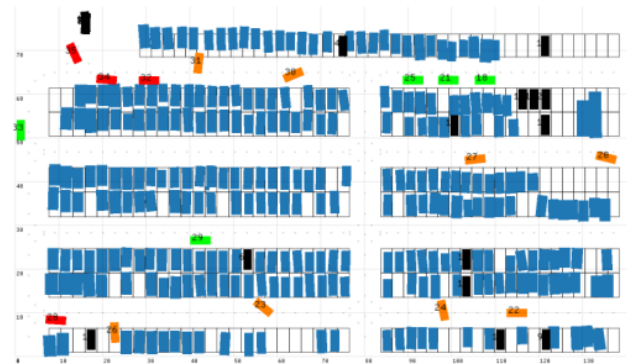
本節では、セクションVのシミュレータと様々な割り当て戦略の性能を示す。その結果、駐車場における車両の自動化は走行時間を大幅に短縮し、ニューラルネットワーク法は提案された戦略の中で最も走行時間を短縮することが示された。DLPデータセットの記録車両が駐車場に占有されている場合と、車両の出入り速度が異なる空き駐車場がある場合の2つのシミュレーションシナリオを示す。すべてのシミュレーションは、MSI Stealth 15M ラップトップと第12世代Intel i7-1280P 2.00 GHz CPU で計算されている。シミュレータはPython 3.8でプログラムされている。

A. シミュレーションシナリオ: 占有を伴う駐車場

このシナリオは、現実世界における駐車場の現状と、車両自動化と我々の割り当て戦略の最も現実的な比較である。DLPデータセットから記録された車両に従って、シミュレーション全体を通して駐車場がある程度占拠されていると仮定する。時間の入出庫など、実際のドライバーの行動を考慮する。運転自動化と割り当て戦略の影響を、実際の人間のドライバーの行動のベースラインと比較してテストする。



(a) 操縦よりもロットに入る車両の割合が多い場合のシミュレーションスナップショット。



(b) ロット内を操縦する車両の割合が進入する車両よりも多い場合のシミュレーションスナップショット。

図9: シミュレーションのスナップショット。

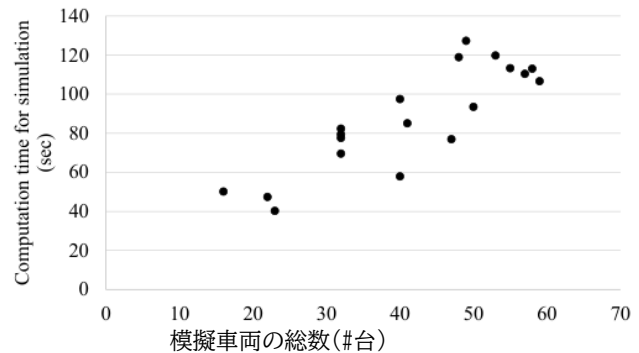


図10: シミュレーション車両総数に対するシミュレータの計算時間。この図には、DLPデータセットのシーン12からシーン30が含まれている。

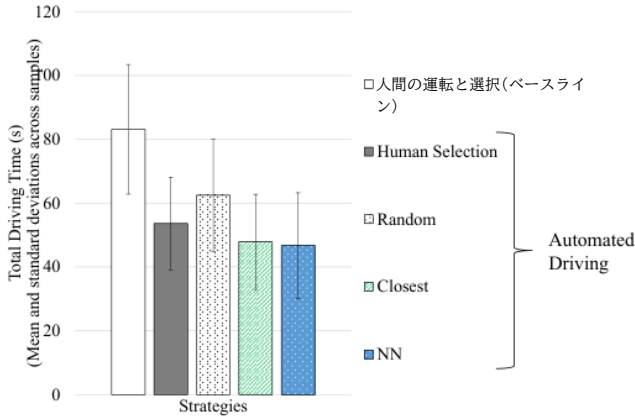


図11:DLPデータセットから記録された車両によると、車両占有率が高いロットにおける提案戦略の運転時間。自動運転は、人間の運転に比べ、運転時間を大幅に短縮することができる。このシナリオでは、Closest戦略とNeural Network戦略がRandom戦略を上回る。ニューラルネットワーク戦略は、総運転時間の平均をベースラインから43.8%削減する。

1) *Methodology*: We use the vehicles \mathcal{A} and obstacles \mathcal{O} from scenes 12-30 of the DLP dataset. The entering vehicles A_{enter} はデータセットに現れると同時に入口で生成され、出口車両 A_{exit} はデータセットに記録されたのと同時に駐車場から操縦を開始する。各車両 $i \in \mathcal{A}$ について、データセット $[i]$ におけるその軌跡の最大速度を、自律走行車の基準巡航速度 v_{cruise} とする。人間のドライバーは頻繁に停止するため、平均速度は通常非常に低く、最高速度は各ドライバーの希望する速度をよりよく反映する。ここで評価した割り当て戦略は、人間選択、ランダム、最接近、ニューラルネットワークである。人間選択戦略は、実際の人間のドライバーがロット状態を一人称で観察するという制限のある行動を反映しているため、ベースラインとして機能する。カスタマイズされたソリューション戦略は、部分的に占有されている駐車場では適用できない。各シーンと各割り当て戦略について、シミュレーションを10回繰り返した。

2) 結果 Fig. 9はシミュレーションの途中にある2つのスナップショットである。車両は、異なる状態を表すために異なる色で表示されている：青は車両が静止していること、黒は車両が駐車または運転を終えていること、緑は運転中であること、赤はブレーキをかけていること、オレンジは車両が駐車場所の出入りを目的とした駐車操作を行っていることを意味する。シミュレーションの完全な動画もオンラインでアップロードされています³。図10は、DLPデータセットのうち、人間選択割り当て戦略と自律走行コントローラを使用した場合のシーン12から30の計算時間を報告する。例えば、あるシーン(DJI 0012)は元データの32台の車両をカバーし、シミュレータはシーン全体を計算するのに79.4秒かかる。シミュレーションする車両が増えると、計算時間が長くなる。

³<https://youtu.be/qUWVLN-5RSU>

表 II: 空のロット実験のパラメータセット

| Parameter set | 進入車両数 | 出口車両の数 | 平均入出庫率(秒) |
|---------------|-------|--------|-----------|
| 1 | 30 | 0 | 8 |
| 2 | 15 | 15 | 8 |
| 3 | 15 | 15 | 12 |
| 4 | 10 | 20 | 8 |
| 5 | 10 | 20 | 12 |

なお、録画映像の長さは1シーンあたり平均431.7秒、標準偏差は49.7秒である。

図11では、人間の運転と選択、人間の選択、ランダム、最接近、およびニューラルネットワークによる駐車場の性能を示している。棒グラフは平均運転時間を示し、エラーバーは平均値から標準偏差を足したり引いたりしている。自動運転(単車制御、ルールベースの衝突回避)により、運転時間が大幅に短縮されることがわかる。この時間短縮は、効率的な経路計画、滑らかな速度プロファイル、効率的な衝突回避ルール、シミュレーションにおける歩行者の省略によるものである。この結論は、人間のドライバーが駐車場の周りを回る時間を減らして利用可能な場所を見つけ、衝突回避のために頻繁にストップ&ゴーをし、熟練しない駐車操作に苦戦するDLPデータセットを観察したことから得られた。割り当て戦略の中で、Random戦略が最も運転時間が長い。ClosestとNNは、(自動運転による)人間選択よりわずかに良いだけであるが、最高のパフォーマンスを発揮する。以下では、駐車場の運用に関する様々なシナリオにおいて、これらの戦略をより詳細に比較する。

B. シミュレーションシナリオ：空のロット

このシナリオでは、提案された割り当て戦略を比較することを深く掘り下げる。シミュレーションでは、空き駐車場から出発し、様々な車両挙動シナリオを生成し、どの戦略で一貫して運転時間を短縮するかを実験する。

1) 方法論：この実験では、空の駐車場から開始し、表IIで定義されたパラメータセットに基づいて、30台の車両の新しい交通シナリオを作成する。パラメータセット1は、すべての車両が急速に駐車場に入るシナリオをモデル化したものである。パラメータセット2と3は、同数の車両が出入りするが、車両の出入り速度が異なる状況をシミュレートする。パラメータセット4と5は、進入車両数が退出車両数より少ないシナリオを示し、それぞれが明確な進入率と退出率を持つ。進入車両 A_{enter} は、指数関数的に時間間隔が分布する入口で生成され、退出車両 A_{exit} は、指数関数的に開始時間が分布する走行車線の端の周辺でランダムに生成される。すべての車両 $i \in \mathcal{A}$ は、同じ参照 $[i]$ 巡航速度 $v_{cruise} = 5m/s$ を持つ。ここで評価した割り当て戦略には、カスタマイズソリューション、ランダム、クローズド、ニューラルネットワークが含まれる。カスタマイズされたソリューション戦略は、交通渋滞が発生せず、同時に駐車操作を行うことができる場合、大域的最適解に最も近い場合、ベースラインとして機能する。

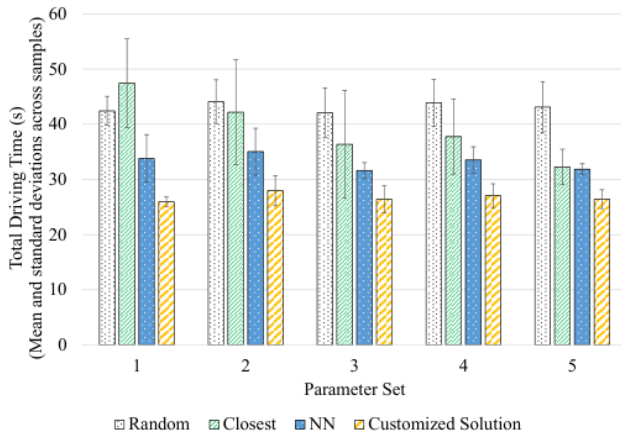


図12: 様々な入退出速度を持つ空き駐車場における提案戦略の運転時間(パラメータセットは表IIに記載)。ニューラルネットワーク戦略は、ランダム戦略とクローズド戦略を上回り、最も理想的な割り当てを表す我々のベースラインであるカスタマイズソリューション戦略に近い。

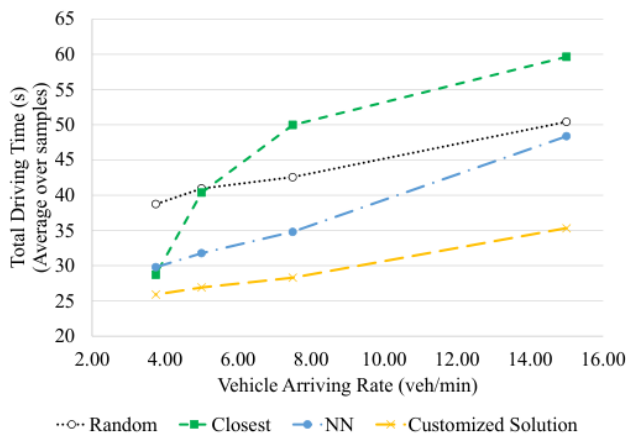


図13: 車両到着率を変化させ、30台の進入車両と10台の退出車両を持つ空駐車場における提案戦略の走行時間。ニューラルネットワーク法は、最も理想的な割り当てを表す我々のベースラインであるCustomized Solution戦略に最も近い。

DLPデータセットには空き駐車場のシナリオがないため、Human Selectionの結果を出すことはできない。各シーンと各割り当てポリシーについて、10回の繰り返しシミュレーションを実行する。

さらに、車両到着率(3.75台、5台、7.5台、15台/分)を用い、30台の進入車両と10台の退出車両を固定数として感度分析を行った。これにより、駐車場への車両の流入が異なり、スポット割り当て戦略が車両のラッシュを処理できるかどうかテストされる。各ケースを10回繰り返し、総走行時間の平均値を示す。

2) 結果 Fig. 12 は、表 II のパラメータセットを変化させた場合の戦略のパフォーマンスを示している。Random戦略は、車両をロットの周囲に均等に分散させるため、パラメータセット間で同様のパフォーマンスを示す。

しかし、この戦略では、途中で駐車場があるにもかかわらず、入口から非常に遠い車両を割り当てることがある。Closest戦略は、車両をできるだけ入口に近づけることで、混雑が激しくなり、駐車場の入口に長い待ち行列が発生する。この戦略は、ロット密度と進入車両数が減少するにつれて、より良いパフォーマンスを示す。

ニューラルネットワーク法は、カスタマイズされたソリューション戦略に最も近い、優れたパフォーマンスを発揮する。一般的に入口に近い場所に車両を配置するが、ロットの通路では比較的均等である。これにより、入口付近での混雑が解消され、他の車両の駐車を待つ時間が短くなる。カスタマイズされたソリューション戦略は、各シナリオに合わせたカスタマイズされたソリューションであり、入口に近い車両を割り当てることが、同時に駐車できるように各通路に十分に広がっていることに注意してください。したがって、カスタマイズされたソリューション戦略は、自動的に解くことは難しく、実用的ではないが、すべてのパラメータで最高のパフォーマンスを発揮する。

次に、図13は、割り当て戦略が、ロット内の車両のフローの違いによってどのように異なるパフォーマンスを示すかを示している。車両がより高い速度で到着すると、車両のラッシュが混雑を招き、すべての割り当て戦略の総走行時間が増加する。カスタマイズされたソリューション戦略は、できるだけ混雑を避けるためにソリューションを手動で調整したため、最短の運転時間となる。図12の結果と同様に、ニューラルネットワークはCustomized Solution戦略に忠実に従っている。Closest戦略は、到着率が低い場合には良好なパフォーマンスを示すが、進入車両の割合が増加するにつれて走行時間が増加する。ランダム割り当ては、入口付近の空きスポットの機会を利用しないため、到着率が低いと比較的性能が低下する。

VII. 結論と今後の課題

本研究では、駐車場における複数のコネクテッド自律走行車を効率的にシミュレートするための新しいフレームワークを提示し、自動化と通信技術を通じて、車両の駐車効率を最適化するための戦略を開発し、その性能を実験することを可能にする。このフレームワークは、個人レベルの車両制御と衝突回避アルゴリズムから、駐車車両を同時にスポットに割り当てるための上位戦略まで、幅広いシミュレーションレベルを包括的にカバーしている。さらに、車両を効率的に駐車場に割り当てるための様々な戦略を設計する。戦略の目的関数は一般化できるかもしれないが、我々は、車両の総走行時間を短縮し、駐車インフラの効率を高め、エネルギーのような資源を節約する可能性のある戦略を実験している。マルチピークルシミュレータと実際の駐車場データセットを用いて、スポット割り当て戦略の性能を分析する。我々は、車両の自動化と通信が総走行時間を大幅に短縮できること、そしてニューラルネットワークに基づく我々のスポット割り当て戦略が駐車時間を最大43.8%短縮し、他の戦略を凌駕することを推論する。

今後、駐車場自動化の理解と計画について、多くのことが期待される。まず、これらのシミュレーションに基づく実験結果を、

自律走行車を用いた実世界での実験で検証することができる。シミュレーションからフィールドワークへの移行は、ハードウェア・イン・ザ・ループやシミュレーション・イン・ザ・ループ形式の実験など、段階的に行うことができる。特に、駐車場のシナリオを一度にテストするために、複数の自律走行車を獲得することは(経済的、技術的、あるいは安全的に)困難であるためである。

第二に、自律走行車は市場に参入するため、自動化技術や通信技術の様々なレベルの車両と混在する可能性が高い。自律走行車が実世界で人間主導の車両と混在している場合に生じる可能性のある多くの問題を研究することは極めて重要である。自律走行車、人間主導車、非連結車、歩行者と車両の交差点には、広範なモデリングと制御が必要となる。

第三に、自動車は温室効果ガスの排出を削減し、持続可能なエネルギー源を利用するために電化される。車両のバッテリー情報はデジタル化され、充電動作(主に駐車場で発生する)は最適化と制御によって制御可能になったため、モビリティエネルギーの効率的な運用のための絶大な機会が得られた。したがって、電気自動車充電のインフラ効率やサービスの質の向上、再生可能エネルギーの利用の最大化、施設コストの削減など、さまざまな目的のために、充電インフラの運用を駐車場インフラに統合することができる。

ACKNOWLEDGMENT

本書で紹介する研究成果は、米国高等研究計画局(ARPA-E)の助成を受けたものである。DE-AR0000791のエネルギー省。

APPENDIX A REFORMULATION OF COLLISION AVOIDANCE CONSTRAINTS

ここでは、衝突回避制約(6d)の微分可能な再定式化を示す。

$$\text{dist}(\mathbb{B}(z_{k|t}), \mathbb{B}_{\text{obs}}^{[o]}) \geq d_{\min}, \forall o.$$

According to [29, 7, 31], (6d) is equivalent to

$$\forall o \in \mathcal{O}, \exists \lambda^{[o]} \geq 0, \mu^{[o]} \geq 0, s^{[o]} : \|s^{[o]}\| \leq 1, \quad (11a)$$

$$-g(z_{k|t})^\top \lambda^{[o]} - b^{[o],\top} \mu^{[o]} \geq d_{\min}, \quad (11b)$$

$$G(z_{k|t}) \lambda^{[o]} + s^{[o]} = 0, \quad (11c)$$

$$A^{[o]} \mu^{[o]} - s^{[o]} = 0, \quad (11d)$$

ここで、 $G(z_{k|t})$, $g(z_{k|t})$ は車両状態が $z_{k|t}$ のときの車両本体多角形 $B(z_{k|t})$ を記述し、 $A^{[o]}$, $b^{[o]}$ は静的障害物の多角形 $\mathbb{B}_o^{[o]}$ を記述する。

APPENDIX B

ルールベースの衝突回避の擬似コード

ここでは、セクションIII-Cで紹介したルールベースの衝突回避アルゴリズムのヘルパー関数を示す。

アルゴリズム1は、ある車両*i*があるタイムステップ数以内に衝突する車両を決定する。これは、計画された軌道を使用してすべての車両の動きをシミュレートし(5~9行目)、ポリトープを比較することによって行われます(1014行目)。 $d_{\text{crashcheck}}$ は衝突の可能性をチェックする半径、 $t_{\text{lookahead}}$ は衝突をチェックするために前方にシミュレートするタイムステップ数である。

アルゴリズム1 衝突リスク評価

```

1:   ▷ Determine any vehicle  $j$  will collide with
    $i$  within  $t_{\text{look\_ahead}}$  time steps if all vehicles continue on
   their current trajectories
2:   ▷ Constants:  $d_{\text{collision\_check}}, t_{\text{look\_ahead}}$ 
3:  $s_{\text{crash}} \leftarrow \emptyset$ 
4:  $s_{\text{nearby}} \leftarrow$  vehicles within  $d_{\text{collision\_check}}$  of  $i$ 
5: for  $t = 1, 2, \dots, t_{\text{look\_ahead}}$  do
6:   Simulate  $i$  moving forward one time step
7:   for  $j$  in  $s_{\text{nearby}}$  do
8:     Simulate  $j$  moving forward one time step
9:     for  $j$  in  $s_{\text{nearby}}$  do
10:      if  $\mathbb{B}^{[i]} \cap \mathbb{B}^{[j]} \neq \emptyset$  then ▷ The vehicle bodies of  $j$ 
      and  $i$  intersect
11:         $s_{\text{crash}} \leftarrow s_{\text{crash}} \cup j$ 
12: return  $s_{\text{crash}}$ 

```

アルゴリズム2は、2台の車両*i*と*j*が衝突する可能性のある場合の優先度を決定する。これは、2つの車両中心間のベクトルによって形成される角度を使用して行う。これは、どの車両が他方より遠くに進んでいるかの尺度として使用することができ、したがって優先順位を持つべきであるからである。このアルゴリズムの視覚化を図14に示す。

アルゴリズム2 優先度評価

```

1: 車両  $i$  と  $j$  の間に衝突リスクがある場合、どの車両が優先
   されるかを決定する。
2:  $\psi_{ij} \leftarrow$  heading angle formed by vector from  $i$  to  $j$ 
3:  $\psi_{ji} \leftarrow$  heading angle formed by vector from  $j$  to  $i$ 
4:  $\psi_{i,\text{diff}} = \psi_{ij} - \psi_i$ , adjusted by factors of  $2\pi$  so  $|\psi_{i,\text{diff}}| < \pi$ 
5:  $\psi_{j,\text{diff}} = \psi_{ji} - \psi_j$ , adjusted by factors of  $2\pi$  so  $|\psi_{j,\text{diff}}| < \pi$ 
6: if  $|\psi_{i,\text{diff}}| > |\psi_{j,\text{diff}}|$  then
7:   return  $i$  has priority
8: else
9:   return  $j$  has priority

```

アルゴリズム3は、車両*i*が他の車両*j*を「通過」しているかどうかを判定するもので、*j*の各前隅にその後隅のそれぞれが形成するベクトルが、*i*の方位から90度以上離れていると定義する(2~6行目)。また、オプションのバッファ距離 d_{buffer} があり、車両*i*の中心がある距離だけ*j*から離れていなければならない(7行目)。このアルゴリズムの視覚化を図15に示す。

REFERENCES

[1] Graham Cookson, Bob Pishue, 米国、英国、ドイツにおけるパーキングペインの影響、技術報告書、

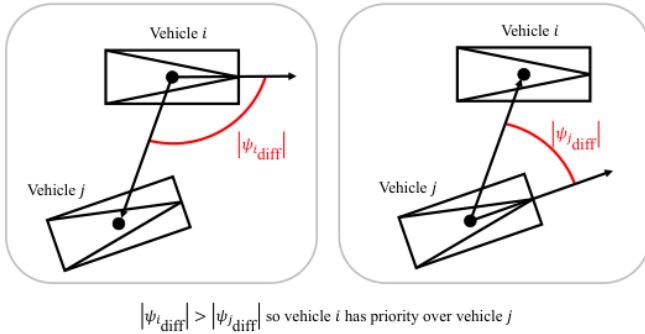


図14: アルゴリズム2の可視化(車両iが車両jより優先される場合)

アルゴリズム3 パッセージ評価

- 1: 車両iが車両jを経由して走行したかどうかを判断する。 with d_{buffer} を用いたオプション基準
- 2: for $c_1 \in i$ の後角 do
- 3: for $c_2 \in j$ の前角 do
- 4: $\psi_{ij} \leftarrow$ angle formed by vector from c_1 to c_2
- 5: if $|\psi^{[i]} - \psi_{ij}| < \frac{\pi}{2}$ then
- 6: return False
- 7: if d_{buffer} is provided and $\sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} < d_{buffer}$ then $\triangleright i$ is not d_{buffer} away from j
- 8: return False
- 9: return True

INRIX Research, 2017. URL: <https://www2.inrix.com/research-parking-2017>.

- [2] A. O. Kotb, Y. C. Shen, X. Zhu, Y. Huang, IParker-A New Smart Car-Parking System Based on Dynamic Resource Allocation and Pricing, IEEE Transactions on Intelligent Transportation Systems 17 (2016) 2637–2647. Publisher: IEEE.
- [3] Y. Geng, C. G. Cassandras, New 'smart parking' system based on resource allocation and reservations, IEEE Transactions on Intelligent Transportation Systems 14 (2013) 1129–1139. Publisher: IEEE ISBN: 9550121011.

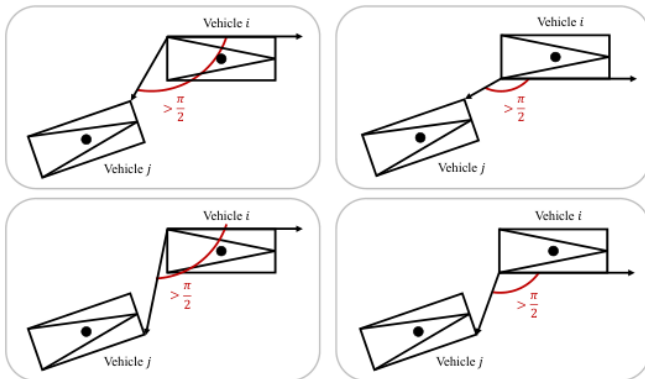


図15: 車両iが車両jを通過したアルゴリズム3の可視化

- [4] H. Banzhaf, F. Quedenfeld, D. Nienhuser, S. Knoop, J. M. Zollner, High density valet parking using k-deques in driveways, IEEE Intelligent Vehicles Symposium, Proceedings (2017) 1413–1420. ISBN: 9781509048045.
- [5] H. Banzhaf, D. Nienhuser, S. Knoop, J. Marius Zollner, The future of parking: A survey on automated valet parking with an outlook on high density parking, IEEE Intelligent Vehicles Symposium, Proceedings (2017) 1827–1834. ISBN: 9781509048045.
- [6] G. Serpen, J. Debnath, Design and performance evaluation of a parking management system for automated, multi-story and robotic parking structure, International Journal of Intelligent Computing and Cybernetics 12 (2019) 444–465. Publisher: Emerald Publishing Limited.
- [7] X. Zhang, A. Liniger, A. Sakai, F. Borrelli, Autonomous Parking Using Optimization-Based Collision Avoidance, in: Proceedings of the IEEE Conference on Decision and Control, volume 2018-Decem, IEEE, 2019, pp. 4327–4332. URL: <https://ieeexplore.ieee.org/document/8619433/>. doi:10.1109/CDC.2018.8619433, iSSN: 07431546.
- [8] D. Kim, K. Huh, Neural Motion Planning for Autonomous Parking (2021). ArXiv: 2111.06739.
- [9] B. Li, K. Wang, Z. Shao, Time-Optimal Maneuver Planning in Automatic Parallel Parking Using a Simultaneous Dynamic Optimization Approach, IEEE Transactions on Intelligent Transportation Systems 17 (2016) 3263–3274. Conference Name: IEEE Transactions on Intelligent Transportation Systems.
- [10] B. Li, Z. Shao, A unified motion planning method for parking an autonomous vehicle in the presence of irregularly placed obstacles, Knowledge-Based Systems 86 (2015) 11–20.
- [11] J. M. Kim, K. I. Lim, J. H. Kim, Auto parking path planning system using modified Reeds-Shepp curve algorithm, 2014 11th International Conference on Ubiquitous Robots and Ambient Intelligence, URAI 2014 (2014) 311–315. Publisher: IEEE ISBN: 9781479953325.
- [12] J. K. Suhr, H. G. Jung, Sensor Fusion-Based Vacant Parking Slot Detection and Tracking, IEEE Transactions on Intelligent Transportation Systems 15 (2014) 21–36. Conference Name: IEEE Transactions on Intelligent Transportation Systems.
- [13] J. K. Suhr, H. G. Jung, K. Bae, J. Kim, Automatic free parking space detection by using motion stereo-based 3D reconstruction, Machine Vision and Applications 21 (2010) 163–176.
- [14] J. E. Siegel, D. C. Erb, S. E. Sarma, A survey of the connected vehicle landscape—architectures, enabling technologies, applications, and development areas, IEEE Transactions on Intelligent Transportation Systems 19 (2018) 2391–2406.
- [15] J. Guanetti, Y. Kim, F. Borrelli, Control of connected and automated vehicles: State of the art and future challenges, Annual Reviews in Control 45 (2018) 18–40. ArXiv: 1804.03757 Publisher: Elsevier.
- [16] T. Kessler, A. Knoll, Multi vehicle trajectory coordination for automated parking, in: 2017 IEEE Intel-

- ligent Vehicles Symposium (IV), 2017, pp. 661–666. doi:10.1109/IVS.2017.7995793.
- [17] Y. Li, K. H. Johansson, J. Martensson, A hierarchical control system for smart parking lots with automated vehicles: Improve efficiency by leveraging prediction of human drivers, 2019 18th European Control Conference (ECC) (2019) 2675–2681. ISBN: 978-3-907144-00-8.
- [18] Y. Li, N. Li, H. E. Tseng, I. Kolmanovsky, A. Girard, D. Filev, A Game Theoretic Approach for Parking Spot Search with Limited Parking Lot Information, in: 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), 2020, pp. 1–6. doi:10.1109/ITSC45102.2020.9294257.
- [19] H. Tavafighi, K. Poolla, P. Varaiya, A Queuing Approach to Parking: Modeling, Verification, and Prediction (2019) 1–28. ArXiv: 1908.11479.
- [20] R. Arnott, J. Rowse, Modeling Parking, Journal of Urban Economics 45 (1999) 97–124.
- [21] S. Choi, J. Lee, Optimal planning of parking infrastructure and fleet size for Shared Autonomous Vehicles, Transportation Research Part E: Logistics and Transportation Review 176 (2023) 103213.
- [22] C. Micus, J. Smeets, M. Böhm, H. Krcmar, Customer Integration in Product Development using Big Data: An Example of using Fleet Data to determine Parking Behavior, PACIS 2022 Proceedings (2022).
- [23] X. Shen, X. Zhang, F. Borrelli, Autonomous parking of vehicle fleet in tight environments, in: 2020 American Control Conference (ACC), IEEE, 2020, pp. 3035–3040.
- [24] PTV VISSIM, 2023. URL: <https://www.ptvgroup.com/en-us/products/ptv-vissim>.
- [25] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, V. Koltun, Carla: An open urban driving simulator, in: Conference on robot learning, PMLR, 2017, pp. 1–16.
- [26] X. Shen, M. Lacayo, N. Guggilla, F. Borrelli, ParkPredict+: Multimodal Intent and Motion Prediction for Vehicles in Parking Lots with CNN and Transformer, arXiv:2204.10777 [cs, eess] (2022).
- [27] X. Shen, M. Pan, V. Govindarajan, F. Borrelli, Dragon Lake Parking Dataset, 2023. URL: <https://datadryad.org/stash/dataset/doi:10.5061/dryad.tht76hf5b>. doi:10.5061/DRYAD.THT76HF5B.
- [28] P. Hart, N. Nilsson, B. Raphael, A formal basis for the heuristic determination of minimum cost paths, IEEE Transactions on Systems Science and Cybernetics 4 (1968) 100–107.
- [29] X. Zhang, A. Liniger, F. Borrelli, Optimization-based collision avoidance, IEEE Transactions on Control Systems Technology 29 (2021) 972–983.
- [30] G. M. Hoffmann, C. J. Tomlin, M. Montemerlo, S. Thrun, Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing, in: 2007 American Control Conference, 2007, pp. 2296–2301. doi:10.1109/ACC.2007.4282788.
- [31] R. Firoozi, L. Ferranti, X. Zhang, S. Nejadnik, F. Borrelli, A distributed multi-robot coordination algorithm for navigation in tight environments, 2020. arXiv:2006.11492.

Xu Shen Xu Shen カリフォルニア大学バークレー校の博士候補生兼研究員。清華大学機械工学科卒業、修士課程修了。カリフォルニア大学バークレー校機械工学科卒業。研究テーマは、シングルエージェントとマルチエージェントの行動予測、意思決定、計画、制御など、制約の厳しい環境における自律走行車の安全性と効率性。

Yongkeun Choi Yongkeun (Eric) Choi カリフォルニア大学バークレー校博士研究員。カリフォルニア大学バークレー校で機械工学の学士号と博士号を取得。博士課程在籍中は、ARPA-E NEXTCARプロジェクトに従事。と博士研究員として勤務している。

Alex Wong Alex Wongは、サンフランシスコ・ベイエリアを拠点とするIXLラーニングのソフトウェアエンジニアである。カリフォルニア大学バークレー校でコンピュータサイエンスの学士号と修士号を取得。以前は、カリフォルニア大学バークレー校機械工学科のモデル予測制御ラボで、自律走行車の挙動に焦点を当てて研究していた。

Francesco Borrelli (Fellow, IEEE) イタリア、ナポリのナポリ・フェデリコ2世大学でコンピュータサイエンス工学のラウレア学位を取得、1998年にチューリッヒ工科大学(スイス、チューリッヒ)で博士号を取得。現在、カリフォルニア大学バークレー校機械工学科教授。予測制御の分野で100以上の論文を執筆し、Constrained Optimal Control of Linear and Hybrid Systems (Springer Verlag)という本の著者でもある。研究テーマは、制約付き最適制御、モデル予測制御、および先進的な自動車制御とエネルギー効率の高い建物運搬への応用。2008年、IEEE自動車制御技術委員会委員長に就任。2009年全米科学財団CAREER賞、2012年IEEE制御システム技術賞受賞。2017年、国際自動制御連盟(IFAC)理事会から産業業績賞を受賞。

スコット・J・ムーラ(IEEE会員)理学士号取得。2006年にカリフォルニア大学バークレー校(UCバークレー校)(米国カリフォルニア州バークレー市)で学位を取得し、M. S. 2008年と2011年にそれぞれ米国ミシガン州アナーバーのミシガン大学から博士号を取得。

2011年から2013年まで、カリフォルニア大学サンディエゴ校(米国カリフォルニア州ラホヤ)の博士研究員。2015年から2021年まで、清華・バークレー深圳研究所(中国・深圳)の主任研究員(PI)。2013年、フランス・パリのMINES ParisTech客員研究員。現在、カリフォルニア大学バークレー校のクレア・アンド・ハイ・ウェン・シェン基金特別教授、エネルギー・制御・応用研究所(eCAL)所長、PATH教授、工学科学講座長を務める。研究テーマは、バッテリー、電動化車両、クライメートテックの制御、最適化、データサイエンス。

Moura博士は、米国機械学会(ASME)ダイナミックシステム・制御部門若手研究者賞、全米科学財団CAREER賞、キャロル・D・ソック特別大学院生メンター賞、0・ヒューゴ・シュック最優秀論文賞、UC学長ポストドクトラルフェローシップ、全米科学財団(NSF)大学院研究フェローシップを受賞した。

Soomin Woo Soomin Woo 韓国ソウルのコンクク大学スマートビークル工学科助教授。自動化、電化、通信を通じた持続可能なエネルギーとモビリティシステムに焦点を当てた研究を行っている。カリフォルニア大学バークレー校で博士号を取得。土木環境工学の学位を取得し、ARPA-E NEXTCARプロジェクトの博士研究員として勤務。韓国科学技術院(KAIST)土木環境工学科で学士号および修士号を取得。