

# インフラ型車両制御システムにおける エッジサーバ間予測ハンドオーバー手法

鈴木 有也<sup>1,a)</sup> 佐々木 健吾<sup>1</sup> 牧戸 知史<sup>1</sup>

**概要:** 本研究では、次世代移動通信システム 5G が普及する将来において、都市レベルの大域的な車両制御を行うクラウドと、交差点等の局所的な車両制御を行うエッジサーバが車両を統括制御するインフラ型車両制御システムの実現可能性を検討している。インフラ型車両制御における課題のひとつに、エッジサーバ間の車両制御ハンドオーバーにおいて、ハンドオーバー時間の増加時に車両制御の乱れを抑制することが挙げられる。提案手法では、車両制御プログラムの内部変数を予測して共有することにより、模擬システムにおいて、ハンドオーバー時間の増加時に、コースアウト率の増加を 6.8[%] から 1.0[%] に抑えられることを確認した。

## A Prediction Handover Method between Edge Servers on Infrastructure-based Vehicle Control System

NAOYA SUZUKI<sup>1,a)</sup> KENGO SASAKI<sup>1</sup> SATOSHI MAKIDO<sup>1</sup>

### 1. はじめに

近年、自動運転に関する研究開発が世界各地で精力的に行われている。トヨタ自動車は、自動運転技術に関する研究開発の考え方とアプローチを自動運転白書として公開している [1]。Google の自動運転車の開発部門から分社化した Waymo では、市街地において自動運転車のテスト走行を継続し、2017 年 11 月時点で 400 万マイルを達成している [2]。また、NVIDIA は、LIDAR、レーダ、複数のカメラ等から得られるセンシング情報を統合することで、車両の周囲の状況を高精度に把握することが可能な自動運転車向けの AI 車載コンピュータを精力的に開発している [3]。

本研究では、自動運転の一形態として、図 1 に示すようなインフラ型車両制御システムによるインフラ型自動運転の実現可能性を検討している [4][5][6]。インフラ型自動運転とは、インフラに配置されている ITS アプリケーション（以降、単にアプリケーションと略す）が、複数の車両の位置・速度等を格納したダイナミックマップ（Dynamic

Map, 以降、DM と略す）[7][8] を参照しながら車両群を統括制御する自動運転の形態である。インフラ型自動運転は、Google 等が開発を進めているような LIDAR、レーダ、カメラ等のセンサを車両に搭載して自律的に走行する自動運転では困難な、多数の車両の統括制御を行うことが可能となる。それにより、たとえば、車両の流れを信号で制御する従来の交差点ではなく、車両を停止させずに互いに他車両をすり抜けて走行するような交差点 [9] を実現できる。また、アプリケーションと DM は、クラウド、およびエッジサーバ [10] に分散して配置されるために、広範囲に分布する複数車両の位置・速度・経路等を用いて最適な車両制御を行うことが可能となる。

本研究では、インフラ型車両制御システムの技術として、ハンドオーバー時の車両挙動の乱れを低減するエッジサーバとエッジサーバ間（以降、単にエッジサーバ間と略す）における車両制御のハンドオーバー手法を提案する。この手法は、次世代移動通信システム 5G が普及する将来において、エッジサーバが配置される基地局が増加し、それら基地局のセルが重なり合うことが想定される場合、エッジサーバ間において制御主体（車両制御を行う実体）を移行する手

<sup>1</sup> 株式会社 豊田中央研究所  
Toyota Cenral R&D Labs., Inc.

<sup>a)</sup> naoya@mosk.tytlabs.co.jp

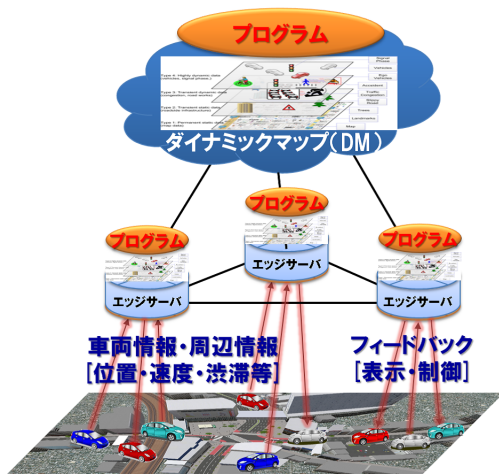


図 1 インフラ型車両制御システム

Fig. 1 Infrastructure-based vehicle control system

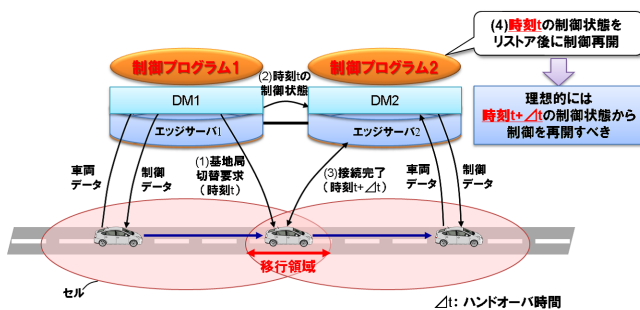


図 2 エッジサーバ間ハンドオーバー

Fig. 2 Edge-to-edge handover

法である。これにより、あるエッジサーバのセルから他のエッジサーバのセルに車両が移動した場合にも、移行先のセルを担当するエッジサーバから車両制御を続行できる。しかしながら、ハンドオーバー時の通信途絶時間が増加すると、移行先のエッジサーバで車両制御の適切な引継ぎが困難になり車両制御の乱れが顕著になる。その車両制御の乱れの低減が、インフラ型車両制御システムの重要課題のひとつに挙げられる。本稿では、エッジサーバ間のハンドオーバー時の通信途絶時間が増加しても車両挙動の乱れを低減する予測ハンドオーバー手法を提案し、インフラ型車両制御システムの模擬システムであるマイクロプロトタイプを用いて評価する。本稿の構成は以下のとおりである。第2章で従来手法の課題について述べ、第3章で提案手法を説明する。第4章で評価を行い、第5章でまとめと今後の展開を述べる。

## 2. 従来手法の課題

インフラ型車両制御システムにおける重要課題のひとつとして、エッジサーバ間ハンドオーバー時の車両制御の乱れを低減するハンドオーバー手法の確立が挙げられる。従来手法 [4] においては、無線ハンドオーバー時における車両との通信途絶時間（以降、ハンドオーバー時間と略す）がきわめ

て短いことを前提とした場合、制御プログラムの制御状態（制御プログラムの内部変数）を移行元と移行先で共有することにより、制御処理のシームレスな引継ぎ、すなわちアプリケーションハンドオーバーが実現できることを確認している。一方で、移行先が制御を再開する際に、移行元から受信した制御状態をそのまま用いて車両制御を再開する場合、ハンドオーバー時間が増大すると、本来あるべき制御状態とのかい離が生じ、制御処理のシームレスな引継ぎが行われずに車両制御が乱れることがある。

その原因は以下のとおりである。図 2 は、エッジサーバ1とエッジサーバ2にそれぞれ配置されているDMと制御プログラムによって、エッジサーバ間ハンドオーバーを行い、エッジサーバが配置されている基地局のセル間を移動する車両がシームレスに制御される様子を示す。従来手法では、エッジサーバが配置されている基地局のセルが重複する領域（以降、移行領域と呼ぶ）に車両が進入した場合に、以下の手順でハンドオーバーを行う。

- (1) 時刻  $t$  でエッジサーバ1が基地局の切り替えを要求
- (2) 時刻  $t$  の制御状態をエッジサーバ1から2に送信
- (3) 時刻  $t + \Delta t$  で車両がエッジサーバ2への接続を完了
- (4) 時刻  $t$  の制御状態からエッジサーバ2が制御を再開

$\Delta t$  はハンドオーバー時間を指す。理想的には、(4)において、時刻  $t + \Delta t$  の制御状態からエッジサーバ2が制御を再開すべきである。この  $\Delta t$  が微小な場合は、時刻  $t$  の制御状態を用いてもハンドオーバー後の車両制御に大きな影響を与えない。反対に、 $\Delta t$  が増大すると、すなわちハンドオーバー時間が増加すると、本来あるべき制御状態とのかい離が生じ、ハンドオーバー後の車両制御が大きく乱れることとなる。上記のようなハンドオーバー時間の増加は、移動局の高速移動時、および台数増加時に発生することが報告されている [11]。文献 [11] によると、たとえば、LTE の場合、半径 1[km] のセル内に 30 個の移動局が存在し、各移動局が 50[km/h] で移動すると、ハンドオーバーの失敗率が 0.30[%] であるのに対し、120[km/h] で移動すると、失敗率は 0.89[%] にまで高まる。移動局の個数が増加して 100 個になった場合は、ハンドオーバーの失敗率が 50[km/h] のときに 0.87[%]、120[km/h] のときに 2.28[%] まで高まる。ハンドオーバーに失敗した移動局は、再接続を試みるため、高速移動時、ならびに接続台数増加時には、ハンドオーバー時間が増加する移動局の割合が必然的に高まる。インフラ型車両制御システムの通信インフラとしてもっとも有力な 5G においては、周波数利用効率を向上させる技術、広い周波数帯域幅を有効活用する技術等により大容量化が実現される予定である [12][13]。しかしながら、IoT の一層の浸透により、車両を含めた多数のモノがインターネットに接続され、それらが高速に移動しうることを想定すると、上記のようなハンドオーバー時間の増加が局所的に起こり得ることは否定できない。このような状況下でインフラ型車両

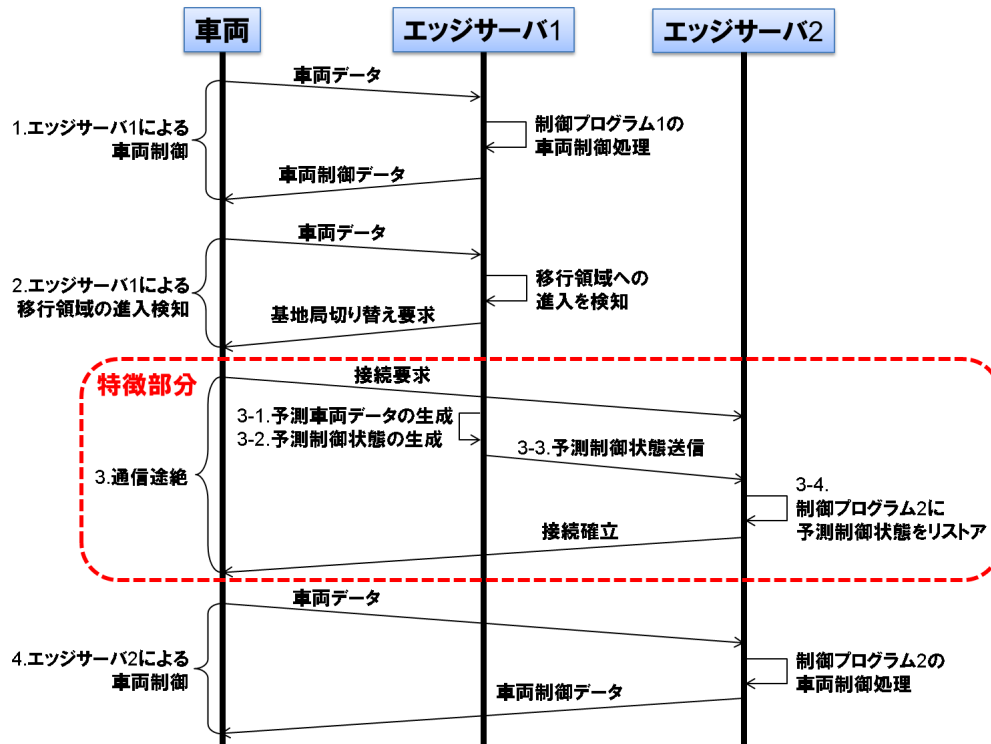


図 3 提案手法の特徴

Fig. 3 The characteristic of our method

制御システムを実現するためには、ハンドオーバー時間の増加が引き起こされた場合においても、車両制御の乱れを極力低減する必要がある。よって、本研究では、ハンドオーバー時間の増加時においても車両制御の乱れを低減するハンドオーバー手法の確立に取り組む。

### 3. 提案手法

図 3 のシーケンス図により提案手法の特徴を示す。シーケンス図は、4つの部分から構成される。エッジサーバ1による車両制御（同図手順1）、エッジサーバ1による移行領域の進入検知（同図手順2）、通信途絶（同図手順3）、およびエッジサーバ1による車両制御（同図手順4）である。本研究の特徴は、同図手順3の通信途絶部分にある。第2章で述べたように、ハンドオーバー時間の増加時にハンドオーバー後の車両制御に乱れが生じる要因は、移行元から受信した制御状態をそのまま用いて車両制御を再開すると、本来あるべき制御状態とのかい離が生じ、制御処理のシームレスな引継ぎが行われないことにある。そこで、本研究では、車両が移行領域に進入し、基地局切り替えのための通信途絶が発生した際に（同図手順3）、移行元のエッジサーバにおいて、本来受信すべき車両データ（位置・速度等）を予測する（同図手順3-1）。この予測された車両データを予測車両データと呼ぶ。次に、その予測車両データを用いて、本来あるべき制御状態を予測し（同図手順3-2）、その予測した制御状態（以降、予測制御状態と呼ぶ）を移

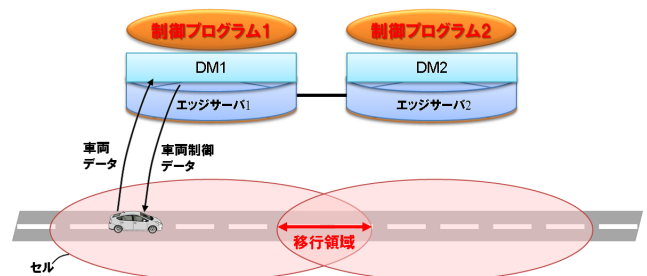


図 4 予測ハンドオーバー (1)

Fig. 4 Predict handover(1)

行先のエッジサーバに送信する（同図手順3-3）。その予測制御状態を制御プログラム2にリストアした後（同図手順3-4）、エッジサーバ2が車両制御を再開することで（同図手順4）、本来あるべき制御状態とのかい離を低減し、車両制御の乱れを抑制する。以降、この提案手法を予測ハンドオーバー手法と呼ぶ。

予測ハンドオーバー手法の詳細を模式図の流れで示す。図4～図8は、エッジサーバ1とエッジサーバ2にそれぞれ配置されているDMと制御プログラムによって、予測ハンドオーバー手法を用いてエッジサーバ間ハンドオーバーを行い、車両をシームレスに制御する様子を示す。エッジサーバ1とエッジサーバ2が車両を制御できる領域は、各エッジサーバが配置されている基地局のセル内に限定される。また、セルの大きさは固定とし、かつエッジサーバ1が配置されている基地局のセルとエッジサーバ2のセルが

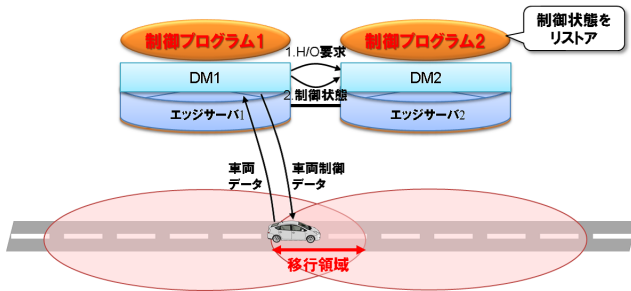


図 5 予測ハンドオーバ (2)  
 Fig. 5 Predict handover(2)

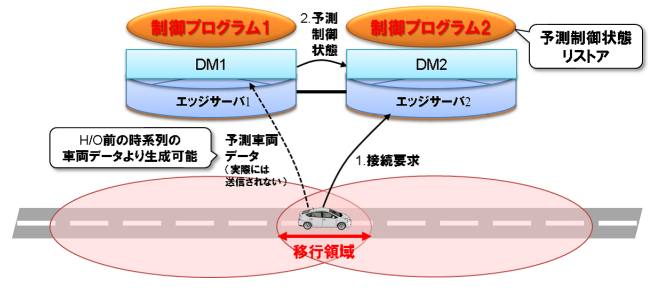


図 7 予測ハンドオーバ (4)  
 Fig. 7 Predict handover(4)

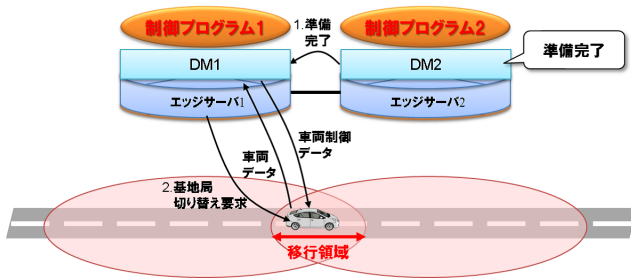


図 6 予測ハンドオーバ (3)  
 Fig. 6 Predict handover(3)

重複している領域が存在することを前提とする。この領域が移行領域であり、双方のエッジサーバは、車両が移行領域に存在する間にハンドオーバを開始し完了させなければならないこととする。以降、「エッジサーバ X が配置されている基地局のセル」のことを、単に「エッジサーバ X のセル」と呼ぶ。

図 4 は、制御プログラム 1 が車両を制御している状態を示す。エッジサーバ 1 のセル内に車両が存在する場合、DM1 が車両から周期的 (例. 100[ms]) に車両データを受信し、制御プログラム 1 に渡す。制御プログラム 1 は、渡された車両データから車両制御データ (アクセル量・ブレーキ量・ステアリング量) を計算し、車両に送信することで車両を制御する。移行領域に車両が進入するまで、制御プログラム 1 は車両を制御し続ける。

図 5 は、エッジサーバ 2 へのハンドオーバ要求と制御状態の送信を示す。移行領域に車両が進入した場合、DM1 は、車両データ中の位置の解析によってそれを検知し、移行先の DM2 にハンドオーバ (図では略して H/O) を要求する (同図手順 1)。DM1 は、DM2 に制御プログラム 1 の制御状態を送信する (同図手順 2)。車両制御データは、制御プログラム 1 が並行して車両に送信し続けることにより、車両制御を中断しないようにする。DM2 は、受信した実行状態を制御プログラム 2 にリストアすることで車両制御の引継ぎに備える。制御プログラム 1 は、並行して車両制御データを車両に送信し続ける。

図 6 は、車両に対する基地局の切り替え要求を示す。DM2 は、制御プログラム 2 の実行準備が完了したことを

DM1 に通知する (同図手順 1)。DM1 は、DM2 の実行準備の完了通知を受信後、車両に対してエッジサーバ 2 のの基地局に切り替えるよう要求する (同図手順 2)。車両制御データは、制御プログラム 1 が並行して車両に送信し続ける。

図 7 は、車両によるエッジサーバ 2 への接続要求、およびエッジサーバ 1 の予測車両データの生成と予測制御状態の送信を示す。車両は、基地局切り替え要求をエッジサーバ 1 から受信後、エッジサーバ 1 との接続を切断し、エッジサーバ 2 に対して接続要求を送信する (同図手順 1)。この間、車両制御データは車両に送信されず、車両は、直前に受信した車両制御データと同一のデータを受信し続けているものとして動作する。前述したように、ハンドオーバ時間が長くなればなるほど、ハンドオーバ後の車両制御に悪影響を与える。なぜならば、図 5 でエッジサーバ 2 が受信した制御状態は、あくまでもハンドオーバ直前の車両を制御するための制御状態であり、ハンドオーバ時間が増加した場合、そのハンドオーバ時間が経過した後の車両を制御するための本来あるべき制御状態とかが離れが発生するからである。本手法では、その制御状態のかが離れを低減するために、ハンドオーバ時間の経過中に、本来あるべき制御状態を予測してハンドオーバ後の車両制御に用いる。すなわち、ハンドオーバしない場合に車両から本来受信するであろう車両データを DM1 で予測して予測車両データを取得し、その予測車両データを入力として制御プログラム 1 を動作させ予測制御状態を得る。その予測制御状態をエッジサーバ 2 に送信することで (同図手順 2)、エッジサーバ 2 は、本来あるべき制御状態に近い、かが離れの少ない制御状態を用いて車両制御を再開できる。予測車両データは、DM1 に蓄積されているハンドオーバ前の時系列の車両データより生成する。また、予測車両データの生成と予測制御状態の送信は、車両データの周期的な受信タイミング (例. 100[ms]) と同じタイミングで、エッジサーバ 2 が車両の制御を再開するまで継続して行う。制御プログラム 2 は、受信した予測制御状態をリストアし、車両との接続が確立した直後から制御を再開できるよう準備する。車両とエッジサーバ 2 との接続には、高速移動時、または接続台数増加時において時間を要する可能性が否定できない。車両がエッジサーバ 2 と接続を確立するまで、エッジサー

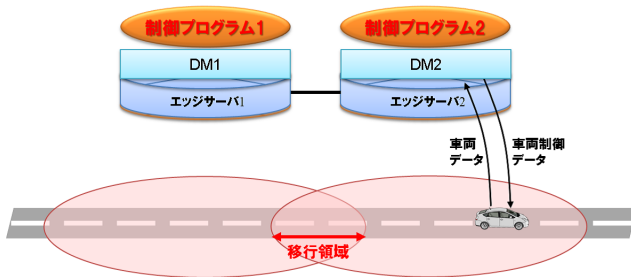


図 8 予測ハンドオーバー (5)  
 Fig. 8 Predict handover(5)

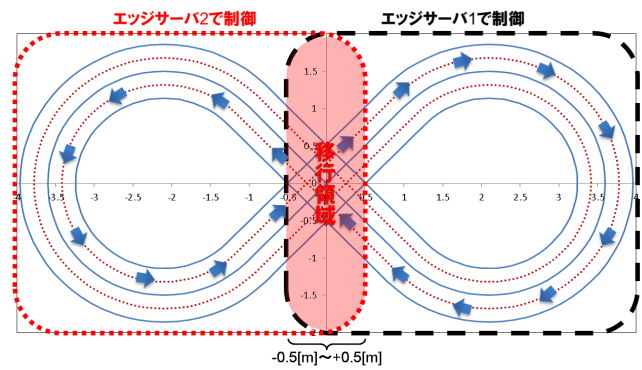


図 10 制御領域と移行領域

Fig. 10 Control region and handover region

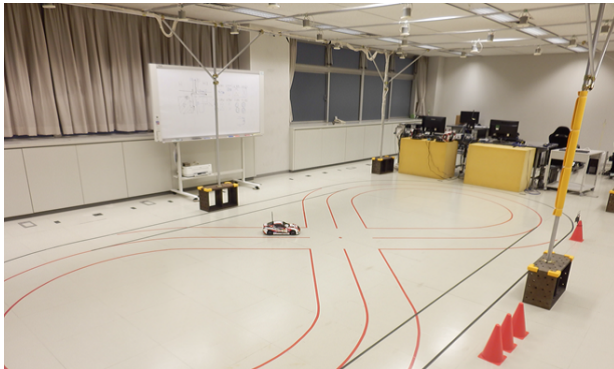


図 9 マイクロカープロトタイプ  
 Fig. 9 Microcar prototype

バ 1 における予測車両データの生成と予測制御状態の送信を続けることによって、ハンドオーバー時間の増加時に適切な予測制御状態を制御プログラム 2 にリストアし続けるようにする。

図 8 は、エッジサーバ 2 が車両を制御している状態を示す。エッジサーバ 2 との接続を確立した車両は、新たに接続したエッジサーバ 2 に対して車両データを送信する。適切な予測制御状態をリストアした制御プログラム 2 が、受信した車両データを入力として車両制御データを計算し車両に送信することで、制御の乱れを抑えつつ、車両制御を再開する。

## 4. 評価

### 4.1 評価方法

本手法の有効性を評価するために、インフラ型車両制御システムの模擬システムとしてマイクロカープロトタイプ [4] を使用する。マイクロカープロトタイプの外観を図 9 に示す。マイクロカープロトタイプは、市販の 1/10 スケールのラジコンを改造したマイクロカーを複数台走行させることができるプラットフォームである。車両データ (位置, 速度, および車両角度) の周期的なセンシングが可能で、かつ外部から車両制御データ (アクセル量, ブレーキ量, ステアリング量) を送信することによってマイクロカーを制御できる。

本評価における各エッジサーバの制御領域と移行領域

を図 10 に示す。8 の字コースを左右に分割し、右領域をエッジサーバ 1 が、左領域をエッジサーバ 2 が制御する。各エッジサーバから他方のエッジサーバへのハンドオーバーは、移行領域で行われる。マイクロカープロトタイプには、PC を 2 台接続してエッジサーバを模擬し、それぞれで車両制御プログラムを動作させる。また、マイクロカーの位置により、各エッジサーバに車両データが適切に配信されるようにする。たとえば、右側をマイクロカーが走行している場合は、エッジサーバ 1 にのみ車両データが送信され、エッジサーバ 2 には送信されないようにする。同様に、移行領域を走行している場合は、双方のエッジサーバに車両データが送信され、左側を走行している場合は、エッジサーバ 2 にのみ車両データが送信され、エッジサーバ 1 には送信されないようにする。各エッジサーバでは、Click modular router [14] が動作し、車両から周期的に送信されるパケットを処理する。すなわち、受信した車両データを含むパケットを入力として、車両制御データを計算し、マイクロカープラットフォームに車両制御データを含むパケットとして出力する。

加えて、ハンドオーバー時間を模擬するために、車両データをエッジサーバに配信する際に無送信時間を設定できるようにする。この無送信時間は 100 [ms] 間隔で設定でき、ハンドオーバー時間の増減を模擬することができる。

上記のマイクロカープロトタイプを用いて、走行軌跡、コースアウト率、およびコースの中心線からのずれを、制御状態予測あり (本手法) と制御状態予測なし (従来手法) で比較し、予測ハンドオーバー手法の有効性を評価する。制御状態予測のための予測車両データは、ハンドオーバー直前の車両の位置・速度・車両角度がハンドオーバー時間中も維持されると仮定することによって算出する。

### 4.2 評価結果

#### 4.2.1 走行軌跡

制御状態予測なしと制御状態予測ありのそれぞれにおいてハンドオーバー時間が走行軌跡に与える影響を図 11 に示

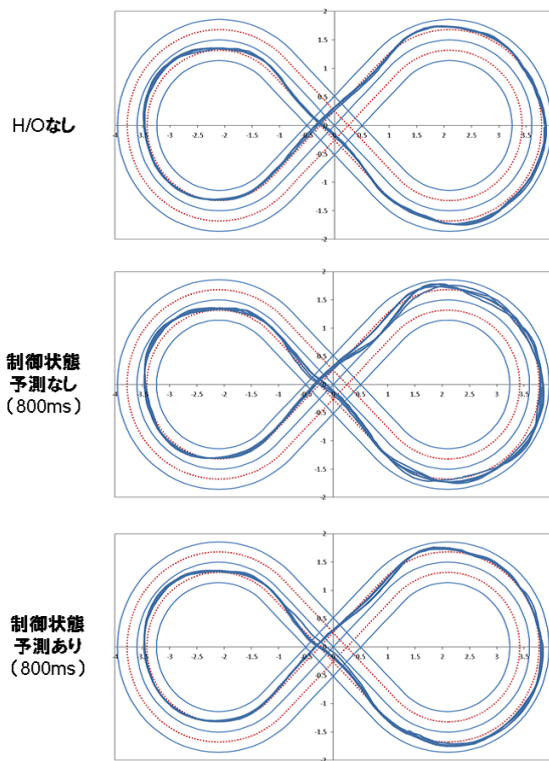


図 11 マイクロカーの走行軌跡  
Fig. 11 Microcar orbit

す。これらの走行軌跡は、ハンドオーバ時間を 800[ms] に設定し、8 の字コースの 5 周走行分を重ねて示している。比較対象として、ハンドオーバを行わずに単一のエッジサーバで制御した場合 (H/O なし) を掲載している。制御状態予測ありの場合、制御状態予測なしに比べて、走行軌跡の乱れが抑制されていることがわかる。制御状態予測ありは、ハンドオーバ時間中の制御状態の予測により、適切な制御状態から車両の制御を引き継げるため、ハンドオーバ時間が増加したとしても、良好な制御が実現できていると考える。

#### 4.2.2 コースアウト率

制御状態予測なしと制御状態予測ありのそれぞれにおいて、ハンドオーバ時間を増加させた場合のコースアウト率を図 12 に示し、制御の乱れを数値化する。縦軸がコースアウト率で、横軸がハンドオーバ時間である。コースアウト率とは、5 周分の走行軌跡のうち、マイクロカーの車体がコースからはみ出た割合で、コースアウト率が増加するという事は、制御の乱れが大きいことを示す。図 12 より、制御状態予測なしの場合は、ハンドオーバ時間が増加するにつれてコースアウト率が高まり、ハンドオーバ時間が 800[ms] の時には 11.0[%] まで悪化する。これは、ハンドオーバを行わずに制御した場合 (ハンドオーバ時間が 0[ms]) と比較して、6.8[%] の増加となる。一方で、制御状態予測ありの場合、ハンドオーバ時間が増加してもコースアウト率は大きく変化せず、ハンドオーバ時間が 800[ms]

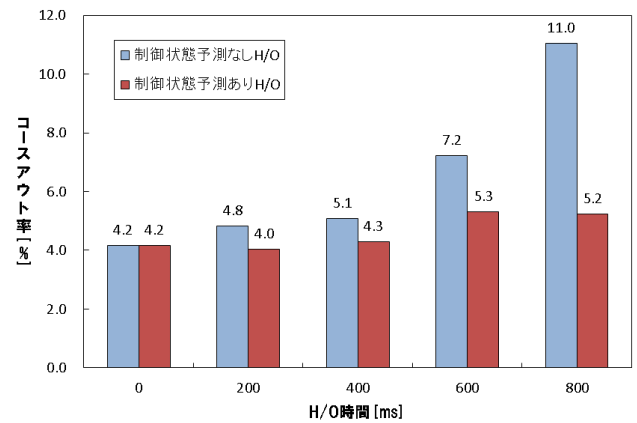


図 12 コースアウト率  
Fig. 12 Course out rate

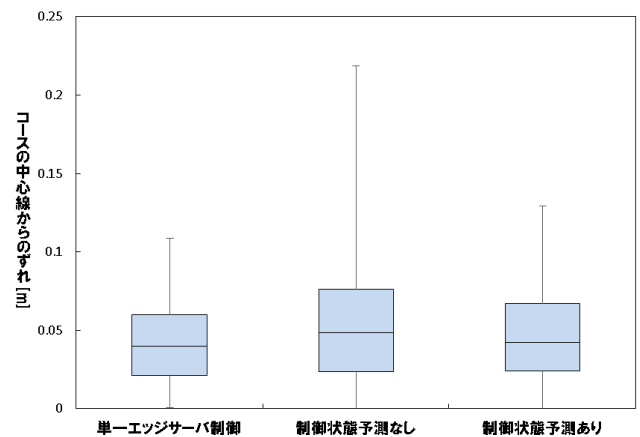


図 13 コースの中心線からのずれ  
Fig. 13 Deviation from the course center line

の時に 5.2[%] にとどめられている。これは、ハンドオーバを行わずに制御した場合と比較して 1.0[%] の増加となる。ハンドオーバ時間が 800[ms] の時と比較すると、制御状態予測ありは、制御状態予測なしと比較して、11.0[%] から 5.2[%] とコースアウト率を半減することができている。これらのことから、ハンドオーバ時間中の制御状態の予測が、ハンドオーバ時間の増加による制御の乱れを大きく低減できることがわかる。

#### 4.2.3 コースの中心線からのずれ

図 13 において、ハンドオーバ時間が 800[ms] のときのコースの中心線からのずれを箱ひげ図で比較する。箱ひげ図は、長方形とその両側に出たひげで、下から最小値、第 1 四分位点、中央値、第 3 四分位点、および最大値を表現する。単一エッジサーバ制御では、中心線からのずれが最大 0.11[m] なのに対し、制御状態予測なしの場合は、0.22[m] となり、単一エッジサーバ制御と比較して 2.0 倍のずれが認められる。一方、制御状態予測ありの場合は、最大 0.13[m] となり、単一エッジサーバ制御と比較して 1.2 倍のずれにとどめることができ、ずれを 41[%] 低減できている。これ

らのことから、制御状態予測ありは、制御状態予測なしと比較して安定した制御の引継ぎを実現できることがわかる。

#### 4.3 評価のまとめ

マイクロカーの走行軌跡に関して、制御状態予測ありを用いる場合は、制御状態予測なしと比較して、ハンドオーバー時間を増加させても走行軌跡の乱れを抑制できることを確認した。このことは、コースアウト率からも確認でき、制御状態予測ありの場合、ハンドオーバー時間が増加してもコースアウト率は大きく変化せず、ハンドオーバー時間が800[ms]の時には、制御状態予測なしでは11.0[%]のコースアウト率のところを、制御状態予測ありでは5.2[%]に抑制できることを確認した。また、コースの中心線のずれに関して、ハンドオーバー時間が800[ms]のときに、制御状態予測なしの場合は、最大0.22[m]のずれが生じるのに対し、制御状態予測ありの場合は、最大0.13[m]となり、ずれを41[%]低減できることを確認した。以上より、本研究が提案する予測ハンドオーバー手法により、ハンドオーバー時間の増加時においても車両制御の乱れを低減できることを確認した。

### 5. まとめと今後の展開

本稿では、インフラ型車両制御システムにおいて、ハンドオーバー時間の増加時に発生する車両制御の乱れを低減する予測ハンドオーバー手法を提案し、マイクロカープロトタイプで評価した。マイクロカーの走行軌跡、コースアウト率、およびコースの中心線からのずれを評価することにより、本手法を用いることでハンドオーバー時間の増加時においても車両制御の乱れを低減できることを確認した。今後の展開としては、実車に適用し、ハンドオーバー時間の増加が車両制御にどの程度影響するかを評価することが挙げられる。

#### 参考文献

- [1] トヨタ自動車：自動運転白書「トヨタにおける自動運転への取り組み—ビジョン、戦略、開発」(2017).
- [2] Waymo: On the road, Waymo (online), available from <https://waymo.com/ontheroad> (accessed 2018-02-01).
- [3] NVIDIA: NVIDIA DRIVE PX, NVIDIA (online), available from <https://www.nvidia.com/en-us/self-driving-cars/drive-px/> (accessed 2018-02-01).
- [4] 鈴木有也, 佐々木健吾, 牧戸知史: インフラ型車両制御システムにおけるクラウド—エッジサーバ間のハンドオーバー手法, 情報処理学会研究報告, 高度交通システムとスマートコミュニティ (ITS) 研究会第68回研究発表会 (2017).
- [5] 鈴木有也, 佐々木健吾, 佐藤健哉, 高田広章: クラウド型自動運転を指向した並列ストリーム型ダイナミックマップ, 情報処理学会論文誌, Vol. 59, No. 1, pp. 88–101 (2018).
- [6] Sasaki, K., Suzuki, N., Makido, S. and Nakao, A.: Vehicle control system coordinated between cloud and mobile edge computing, *In proc. of 2016 55th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pp. 1122–1127 (2016).
- [7] ETSI: Intelligent transport systems(ITS); Vehicular communications; Basic set of applications; Local dynamic map(LDM), EN 302 895, European Telecommunications Standards Institute (2014).
- [8] 渡辺陽介, 高木建太郎, 手嶋茂晴, 二宮芳樹, 佐藤健哉, 高田広章: 協調型運転支援のための交通社会ダイナミックマップの提案, DEIM Forum 2015 (2015).
- [9] Dresner, K. and Stone, P.: A multiagent approach to autonomous intersection management, *Journal of Artificial Intelligence Research*, Vol. 31, No. 1, pp. 591–656 (2008).
- [10] ETSI: Mobile Edge Computing - A key technology towards 5G, Introductory technical white paper, European Telecommunications Standards Institute (2015).
- [11] Dimou, K., Wang, M., Yang, Y., Kazmi, M., Larmo, A., Pettersson, J., Muller, W. and Timmer, Y.: Handover within 3GPP LTE: Design Principles and Performance, *In proc. of IEEE 38th Vehicular Technology Conference Fall*, pp. 1–5 (2009).
- [12] NTTドコモ: 5G 無線アクセス技術, NTTドコモテクニカルジャーナル, Vol. 23, No. 4, pp. 18–29 (2016).
- [13] NTTドコモ: ドコモ5Gホワイトペーパー (2014).
- [14] Kohler, E., Morris, R., Chen, B., Jannotti, J. and Kaashoek, M.: The click modular router, *ACM Transactions on Computer Systems (TOCS)*, Vol. 18, No. 3, pp. 263–297 (2000).