

2018(平成 30)年度 修士学位論文

自動駐車システムの開発と RoboCar による実験的検討

Development of automatic parking system and experimental study by RoboCar

2019 年 2 月 6 日

高知工科大学大学院 工学研究科基盤工学専攻
知能機械システム工学コース

機械・航空システム制御研究室
学籍番号 1215026 日野 卓磨
指導教官 岡宏一 教授
原田 明德 講師

目次

第 1 章 諸元	1
1-1 研究背景	1
1-2 研究目的	3
第 2 章 駐車システムの概要	4
第 3 章 駐車経路の設計	5
第 4 章 RoboCar 実験のための経路生成実験	7
4-1 実験目的	7
4-2 実験方法	7
4-3 実験結果および考察	9
第 5 章 経路設計結果を用いた RoboCar の走行実験	19
5-1 実験目的	19
5-2 実験方法	19
5-3 実験結果および考察	19
第 6 章 誤差補正	22
6-1 誤差補正概要	22
6-2 点 02 における誤差補正方法	23
6-3 点 03 における誤差補正方法	24
第 7 章 誤差補正を含んだ経路設計実験	27
7-1 実験目的	27
7-2 実験方法	27
7-3 実験結果および考察	27
第 8 章 誤差補正検証のための RoboCar の走行実験	34
8-1 実験目的	34
8-2 実験方法	34
8-3 実験結果および考察	34
第 9 章 まとめ	42
9-1 結言	42
9-2 課題	42
付録	43
参考文献	58
謝辞	59

第1章 諸元

1-1 研究背景

近年、自動車における自動運転技術に注目が集まり、多くの自動車会社や日本政府が自動運転車を量産させる目標を掲げて、開発が盛んになっている。また、センサやカメラを用いた衝突回避における自動ブレーキサポートや一定の設定した速度で走るクルーズコントロールなど様々な自動運転支援システムが開発され、現在発売されている車に装備されている車も多数存在する。その中でも、車を駐車する際に接触してしまうことや駐車枠に上手く入らないなど駐車を苦手とする人を支援する形で自動駐車を行うパーキングアシストシステムが開発された。現在、ボタン一つで駐車可能な自動的駐車が行えるシステム搭載車や自動パーレー駐車の実証実験など実用化は目前と迫っているが、場所的課題や非システム搭載車との共存など解決すべき課題は残っているのが現状である。

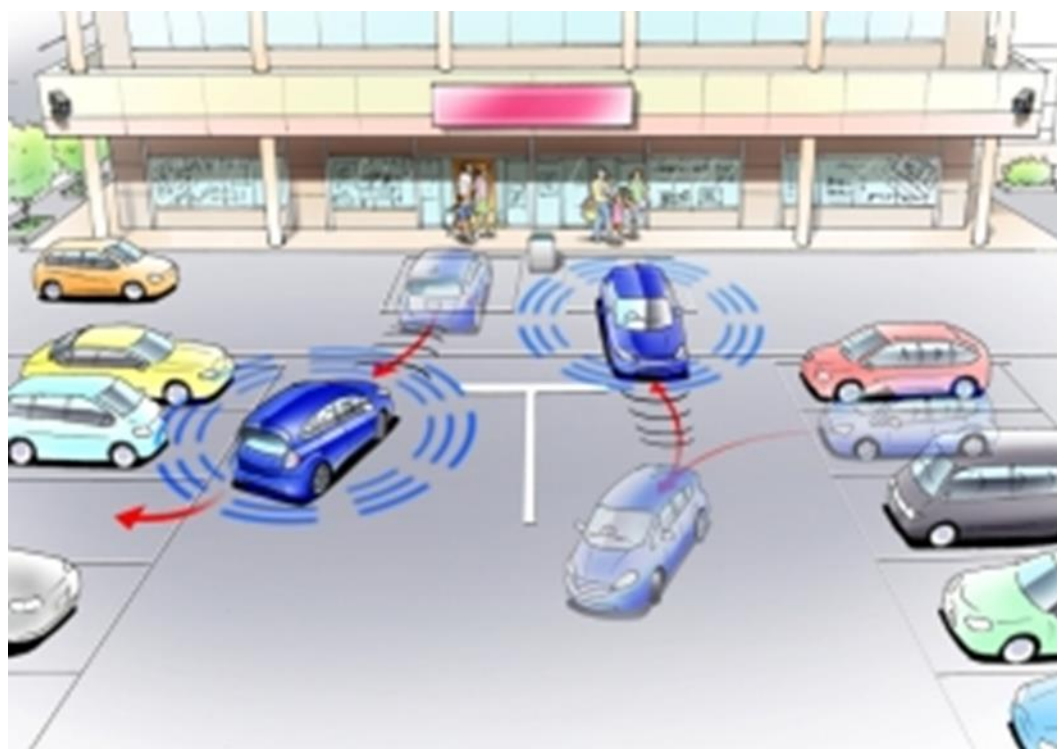


Fig. 1 This illustration shows the basic idea how automatic parking system^[1]

ここで、現状の自動駐車システムについて述べる^{[2][3]}。現在、日本のメーカーで市販化されているものとしては、自動車単体で駐車を行うシステムであり、駐車場内には人が立ち入らず、自動で駐車を行う「自動バレー駐車システム」は実用化目前まで進んでいる。

また、現在の駐車システムにおけるシステムの流れとしては、駐車枠の認知をステレオカメラで行い、ドライバーが空いている駐車枠前に到着後システム動作のボタンを押す。その後、目標の駐車位置までの駐車経路を自動で作成し、ハンドル操作は自動、アクセルとブレーキは手動で行う。さらにシステム作動中に障害物などの検知を行うためにレーザーレンジファインダーによって周囲の認知を行う。

自動バレー駐車システムについては、現在大規模商業施設や空港などの大型駐車場向けに開発・実証実験が盛んに行われており、システムの流れとしては主に二つ存在する。一つは車両と駐車場が協調し、自動車のコントロールは管制コントロールによって行うもの、もう一つは、自動車自身が判断を行い、自ら空きスペースを探し駐車を行うものがある。多くのバレー駐車では、前者の方式を採用しており、効率よく自動で駐車を行うことを重視している。主なメーカーにおける自動駐車機能などの詳細を Table 1 に示す。

Table 1 Detail of automatic parking system^{[4][5][6]}

トヨタ自動車	「インテリジェントパーキングアシスト」 イーザーセット機能、目標駐車位置を自動で補正 「インテリジェントパーキングアシスト2」 超音波センサーとカメラで駐車スペースを検知、 スイッチを押すだけで、適切な駐車開始位置への誘導、バック時のステアリング操作を行う レーザーレンジファインダーによる障害物を検知
日産	「インテリジェントパーキングアシスト」 アラウンドモニターのトップビュー（真上からの映像）を見ながら、クルマを駐車する位置を指定 ドライバーはアクセルとブレーキを操作し、速度を調節しながら駐車を行う
ホンダ	「スマートパーキングアシストシステム」 カメラを使って駐車枠を認識し、最適な駐車経路を割り出し、ハンドル操作を自動で行う 作動中には、タイヤの回転数から自車位置を推定し、アクセルとブレーキ操作、シフトチェンジを行う タイミングをアナウンスしてくれます マルチビューカメラの画像処理技術を活用して、駐車枠を認識し、左右のドアミラー内に搭載したカメラによって、駐車枠かどうか自動で判断。駐車枠を決定した段階で、たくさんある駐車経路から最適な1 経路を選び出し、停車して「スタート」スイッチを押した瞬間からスピーディにハンドル操作のアシストを始められます。

1-2 研究目的

本研究では、自動駐車技術の発展を促進するため、自動駐車専用の駐車場において自動で駐車を行う車の走行システムを提案する。車は、周辺の状況判断から目標駐車枠までの経路を一つに絞りだし、目標経路に従って完全に自動走行を行う。また、雨や雪など路面が非常にすべりやすい天候でも、目標の駐車枠まで駐車を行える自動駐車を可能にする。



Fig. 2 Automatic parking area

第2章 駐車システムの概要

ここで開発を行うシステムは駐車場の出入り口にて、乗車していた人が降り、その後自動で駐車が行われるシステムが対象とする。

Fig. 3 にシステムの流れを示す。システムが作動すると空いている駐車枠を認識し、目標の駐車枠を決定する。目標の駐車枠の決定は、システム上で自動的に行う。まず車に無駄な舵角操作を与えず、誤差なくスムーズに駐車を行えることを考慮した目標駐車経路を作成し、経路に沿って駐車を開始する。走行時、経路誤差を駐車場に設置したカメラによる車両の位置情報から検知する。誤差が設定値以上になれば、位置情報から再び目標駐車位置に向けて設計した再設計経路を作成し、経路修正を行うことにより適宜補正しながら、駐車を完了させる。

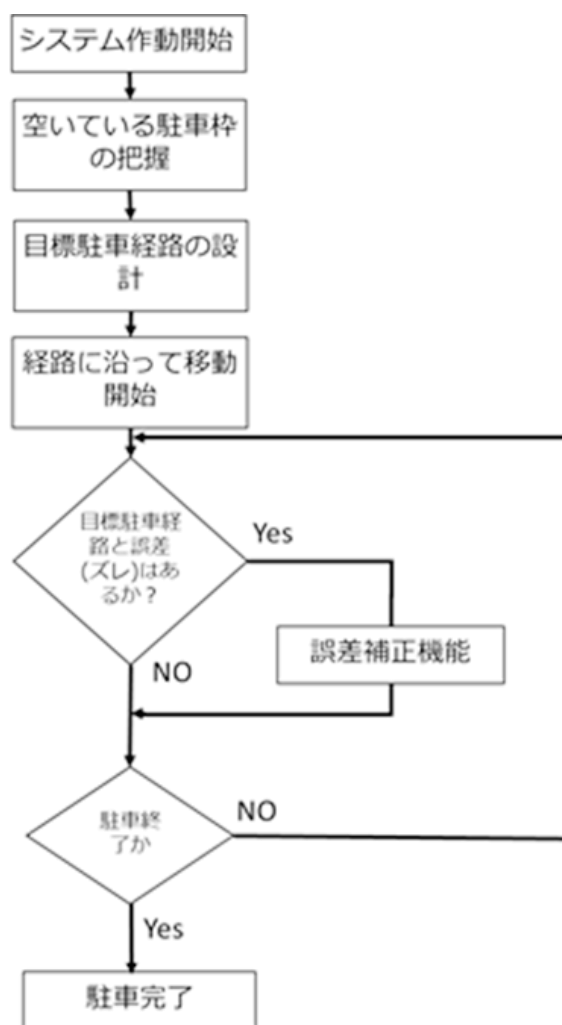


Fig. 3 System overview

第 3 章 駐車経路の設計

まずは、車が走行する前にどのような経路に沿って駐車を行うのかを示す目標駐車経路の設計を行う。設計時に用いる情報は、車の X,Y 座標と車の向きである。

車の駐車において駐車をする車の動きは複数通りあるが、比較的駐車経路の修正が行いやすい方法で駐車を行う。経路は、車が真っすぐ進む直線経路と舵角を一定の角度に切ったときの円軌道の 2 つを用いる。概略を Fig. 5 に示す。Fig. 4 における手順、以下の通りである。

- (1) 車両は左方向から進入し、点 O1 まで直進する。(スタート位置 → O1)
- (2) O1 で一旦停止し、舵角を切り、定常円旋回（舵角一定）で切返し点 O2 まで前進する。
(O1 → O2)
- (3) O2 での停止時に後退の舵角に切替え、点 O3 まで定常円旋回（舵角一定）で後退
(O2 → O3)
- (4) O3 で駐車枠と平行になったことを確認して、舵角を 0 にして直進後退 (O3 → O4)

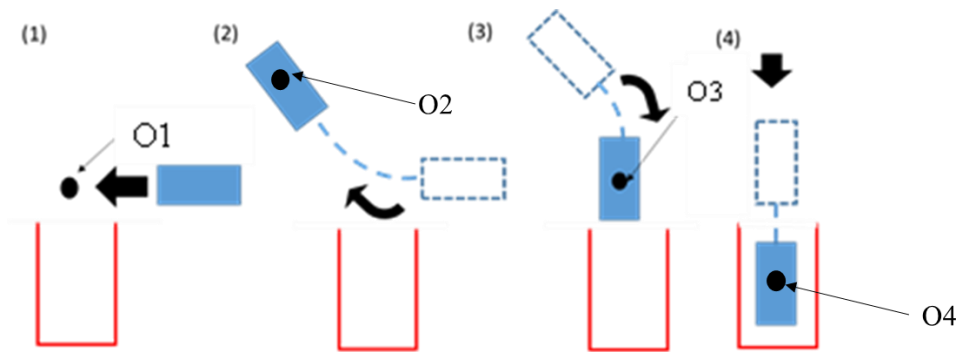


Fig. 4 How to car parking

なお、経路における各点は全て車両の後輪の中央の部分の点で考える。

この経路設計のための具体的な方法^[7]は、以下の通りである。まず目標駐車枠に関連して点 O3 における座標と点 O2 における車の方向角を決め、車両の各舵角における旋回半径を入力する。

点 O3 の座標値は、枠と周囲の壁などを考慮し、スムーズに点 O3 に車を移動させることができるように決める。その後、点 O3 の座標は、 R_r と θ_2 から、

$$X_2 = X_3 + R_r(1 - \cos(90^\circ - \delta_2))$$

$$Y_2 = Y_3 + R_r \sin(90^\circ - \delta_2)$$

として算出できる。

また、点 O1 の座標は、 R_f と θ_2 から

$$X_1 = X_2 - R_f \sin \delta_2$$

$$Y_1 = Y_2 + R_f(1 - \cos \delta_2)$$

として算出できる。

スタート位置はY軸上にあるものとして、点O1まで直進することから $X_0 = 0, Y_0 = Y_1$ となる。
 旋回半径については、

$$R = 0.33\theta^{-0.84}$$

で算出する。(付録2にて後述)

以上よりスタート位置からO4までの座標を基に経路生成を行う。

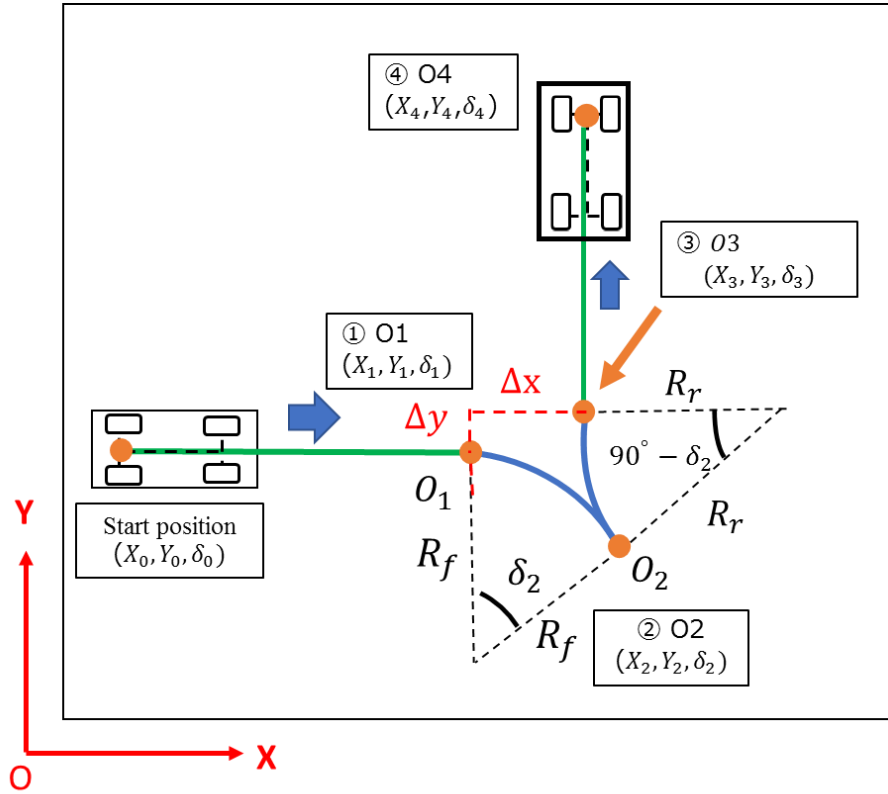


Fig. 5 Parking route

第 4 章 RoboCar 実験のための経路生成実験

RoboCar 駐車経路を設計する上で、実験を実施するための条件を以下に示す。

1. 駐車場内には人や物など障害物はない。
2. 最大舵角は 30 度とし、旋回中の舵角は基本等しいとする。
3. 図 3 に示す $x=0$, $y=0$ のスタート位置以外、および $x=3.1$, $y=2.0$ には壁があると考え衝突しないようにする。また、駐車枠線は踏まない。
4. 実車の 1/10 スケールサイズのモデルであるため、誤差の許容範囲内は、0.1m とする。
5. 駐車場の入口から X 方向に平行に入場し、RoboCar の後輪中心点が $Y=0$ となった時から経路を描くとする。

以上の条件(以下、駐車完了条件と記す)の下、実験のための経路設計を行った。

4-1 実験目的

駐車完了条件を全て満たした経路を最適経路とし、一つの駐車枠における唯一の経路と定めるため、点 O3 の Y 座標 Y_3 と車の方向角、車の舵角を変化させて経路生成実験を行った。

4-2 実験方法

車の方向角と点 O3 の Y 座標 Y_3 、車の舵角の値について、各 3 つ設定した。設定した条件の詳細を Table 2 に示す。車の舵角については、旋回操作が容易になると判断したため前進旋回時と後退旋回時で共に等しくする。数値解析ソフトウェア matlab による経路設計を行った。全ての経路設計を終えた後、 Y_3 のみを変化させた場合、車の方向角のみを変化させた場合、車の舵角のみを変化させた場合について、順に比較し考察を行う。

Table 2 Condition details

conditons number	car direction angle	Y3	Steering Angle	conditon s number	car direction angle	Y3	Steering Angle
1	25	0.4	20	15	45	1.2	25
2	25	1	20	16	45	0.4	30
3	25	1.2	20	17	45	1	30
4	25	0.4	25	18	45	1.2	30
5	25	1	25	19	75	0.4	20
6	25	1.2	25	20	75	1	20
7	25	0.4	30	21	75	1.2	20
8	25	1	30	22	75	0.4	25
9	25	1.2	30	23	75	1	25
10	45	0.4	20	24	75	1.2	25
11	45	1	20	25	75	0.4	30
12	45	1.2	20	26	75	1	30
13	45	0.4	25	27	75	1.2	30
14	45	1	25				

なお、実験を行うための駐車場を Fig. 6 に示し、各駐車枠番号における点 O4 を Table 3 に示す。以下の図から、駐車番号 1 番から 6 番まで順に仮定し、経路設計実験を行う。

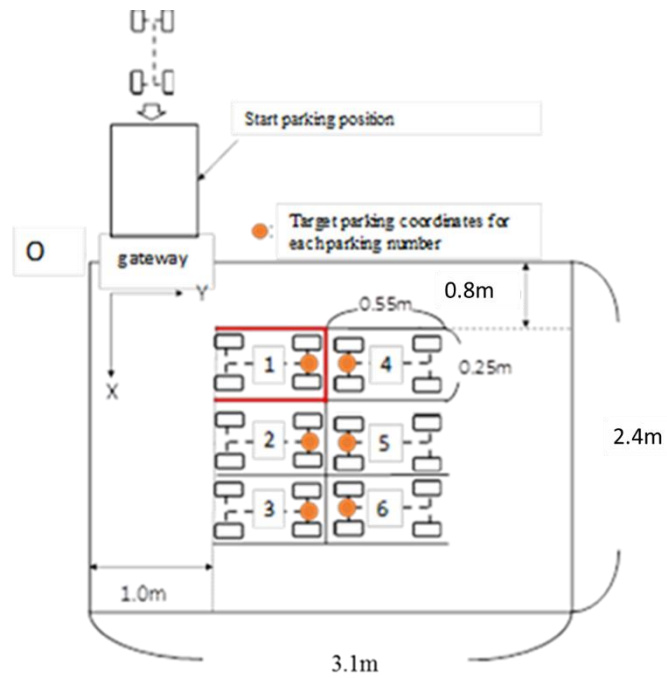


Fig. 6 Parking area map for experiment

Table 3 Point O4 in each parking frame

parking number	Coordinate	
1	0.73	1.4
2	0.98	1.4
3	1.23	1.4
4	0.73	1.7
5	0.98	1.7
6	1.23	1.7

4-3 実験結果および考察

まずは、4-2 に示した実験方法を用いて、駐車枠 1 における実験を行った結果を示す。まず、なお、今回設計した駐車経路が 27 通りあるため、結果を比較し、変化が顕著であった経路のみをここで記載する。(本項で記載できなかったものは、付録 3 に示す。)

まず、車の方向角を 25° 、舵角を 25° とし、Y3 のみを変化させた場合の経路の比較を条件 4、5、6 で行う。表 1 の条件番号 4、5、6 の経路設計結果を順に Fig. 7、Fig. 8、Fig. 9 に示す。

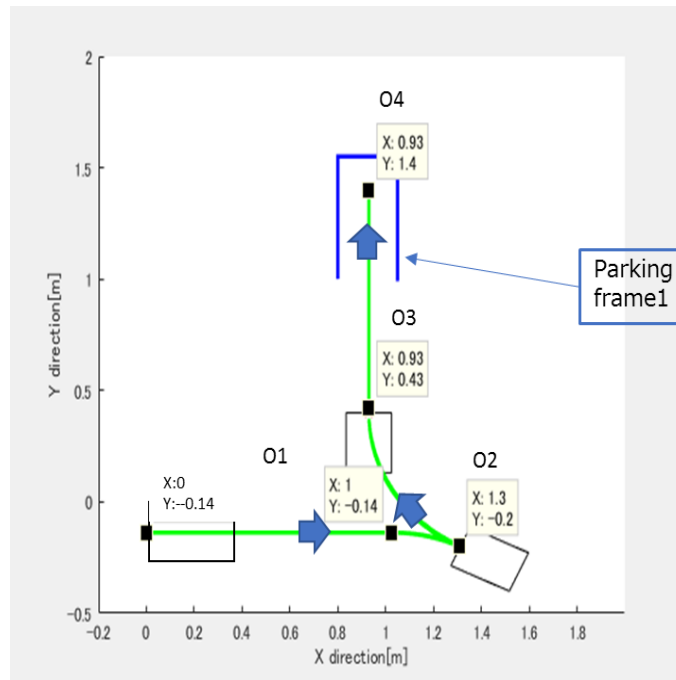


Fig. 7 Parking route result of condition 4

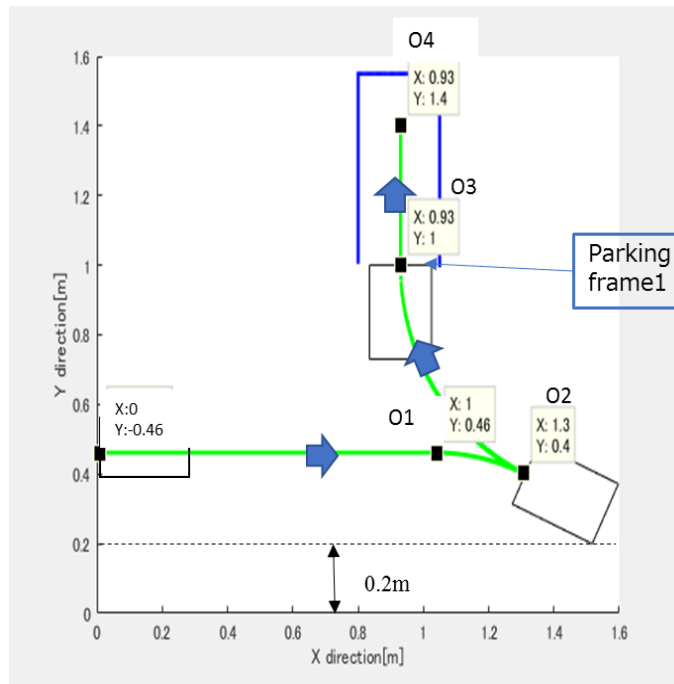


Fig. 8 Parking route result of condition 5

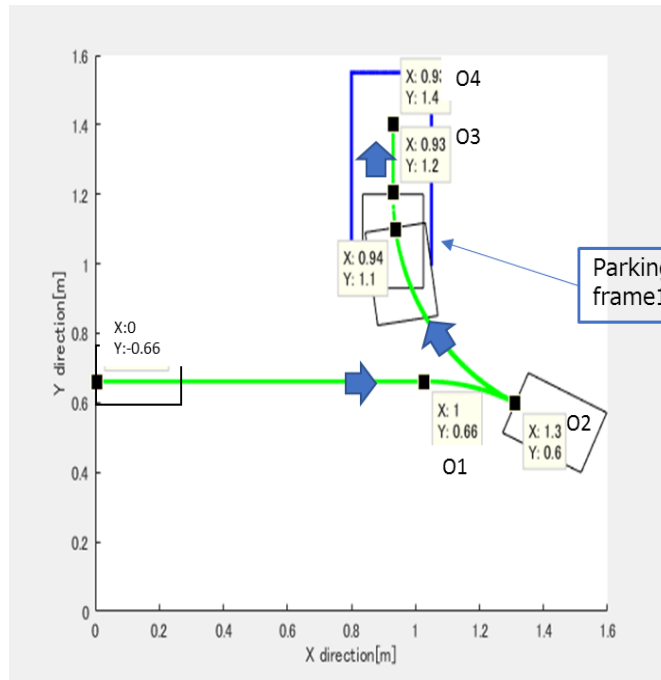


Fig. 9 Parking route result of condition 6

Fig. 7 から **Fig. 9** までの経路設計結果を見ると、Fig. 7 の経路は、スタート位置から点 O2 までの Y 座標の値が負の値になり、O2 に到達するまで駐車場の範囲外を走行することになってしまう結果となった。次に、**Fig. 9** の条件では、点 O2 の座標が(1.3,0.6)であることから、壁に接触する可能性は低いが、点 O3 付近に着目してみるとわずかではあるが、駐車枠の線を踏んでいることがわかる。最後に **Fig. 8** の条件では、まず点 O2 に着目すると座標が(1.3,0.4)、O2 時に壁に一番近い、車体の右前方の座標は(1.52,0.2)であった。車が点 O2 に到着の際、Y 方向における車と柵との距離は、0.2m であることから、経路通りに駐車を行えば柵に衝突する可能性はないと考えた。また、点 O3 付近では、駐車枠線を踏まずに駐車を完了できるこ

とから条件4から条件6のうち、駐車完了条件を満たす最適経路は、Fig. 8の条件5の場合の駐車経路であり Y3 の値については、1.0[m]が最適であると考えた。

次に Y3 の値を 1.0[m]、舵角を 25°とし、点 O2 における車の方向角のみを変化させた場合について条件5、14、23 で比較を行う。条件14、23 の設計結果を順に Fig. 10、Fig. 11 に示す。

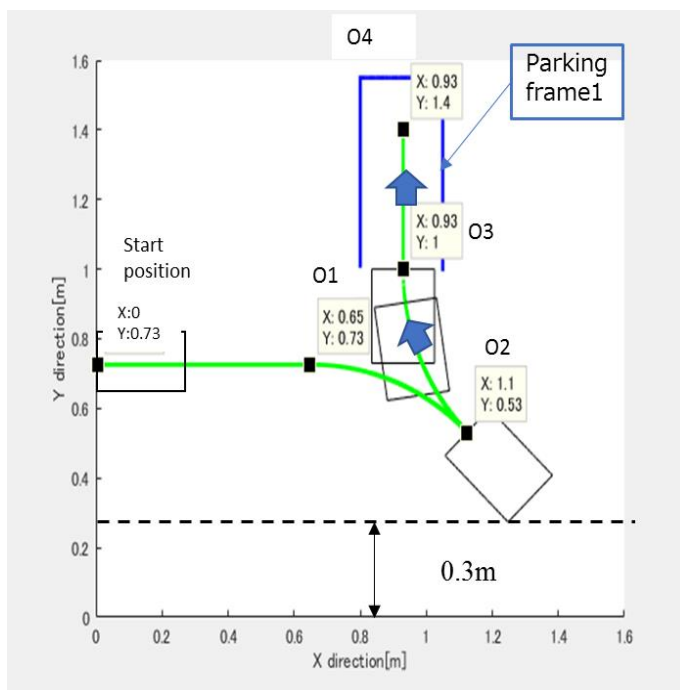


Fig. 10 Parking route result of condition 14

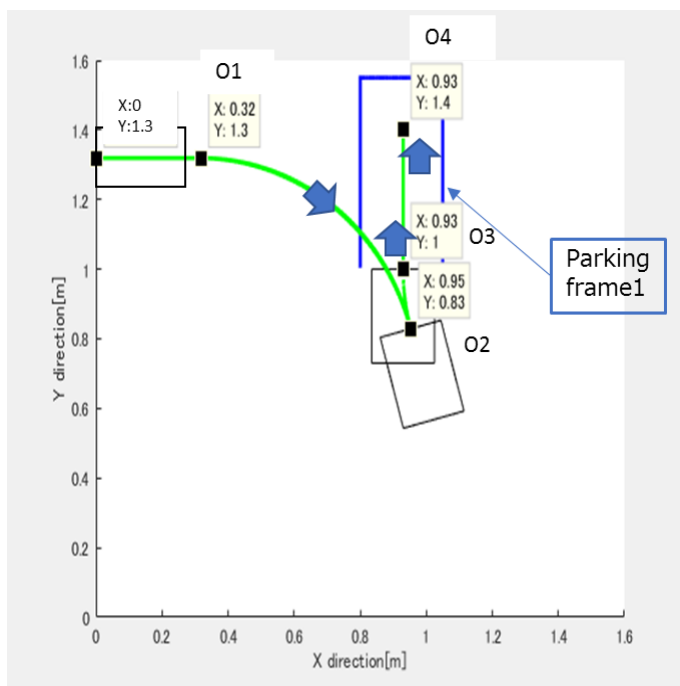


Fig. 11 Parking route result of condition 23

Fig. 8、Fig. 10、Fig. 11 の経路結果を見てみると、Fig. 11 の条件 23 の駐車経路は、柵に衝突する可能性は極めて低いものの、駐車枠をまたいで走行しているため、駐車中の車両に衝突する危険性がある。よって、最適な駐車経路ではないと判断できる。残る Fig. 8 の条件 5 の駐車経路と Fig. 10 の条件 14 の駐車経路だが、どちらも経路通りに駐車を完了すると柵に衝突せず、駐車枠を踏むことはないため、駐車完了条件を満たす駐車経路である。そこでどちらがスムーズに駐車を行えるかを比較し、スムーズな方を今回の比較における最適経路とする。

スムーズな経路であるかどうかを判断するため、以下の定義を定めた。

- 速度一定で、なるべく速く駐車を完了させるため、全体の経路の長さが短くあること

駐車終了条件と上記の定義から Fig. 8 と Fig. 10 を比較する。

まず経路の全体の長さであるが、図 Fig. 8 は、2.47m であったのに対し、Fig. 10 は、2.09m であった。Fig. 8 の経路が長くなった原因として、点 O2 から点 O3 において車の方向角を 75° 変化させることから後退旋回の経路を長くなり、Fig. 10 より点 O2 がスタート位置からより離れた所になければならないからと考えられる。

次に点 O2 にそれぞれ着目すると、Y 方向における柵との距離が Fig. 8 では、0.2m であるのに対し、Fig. 10 では、0.3m であった。接触する可能性を下げるには、壁との距離が離れている方がよいと考えたため、経路の長さの検討も含め、Fig. 8、Fig. 10、Fig. 11 における最適駐車経路は、Fig. 10 の条件 14 の経路であると判断した。

最後に Y3 を 1.0m、点 O2 における車の方向角を 45° に固定し、旋回中の車の舵角を変化させた時の駐車経路の比較を行う。条件 11、17 の設計経路結果を Fig. 12、Fig. 13 に示す。

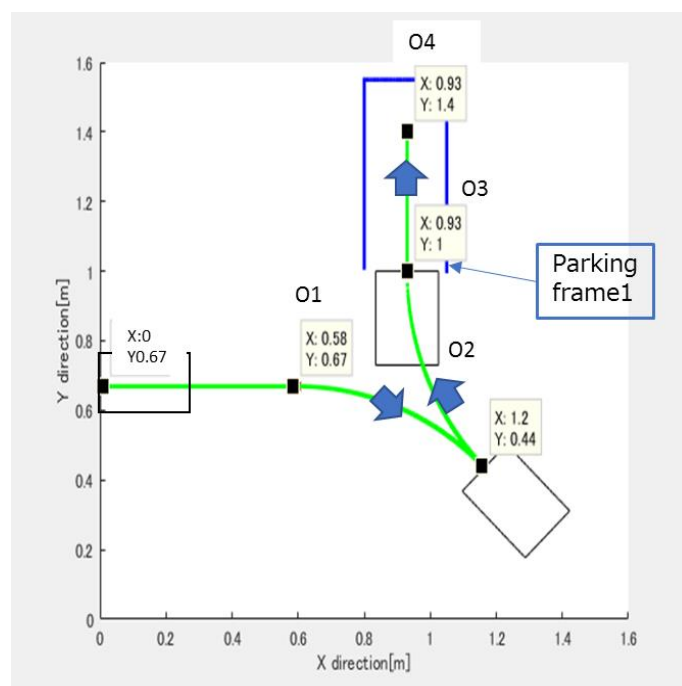


Fig. 12 Parking route result of condition 11

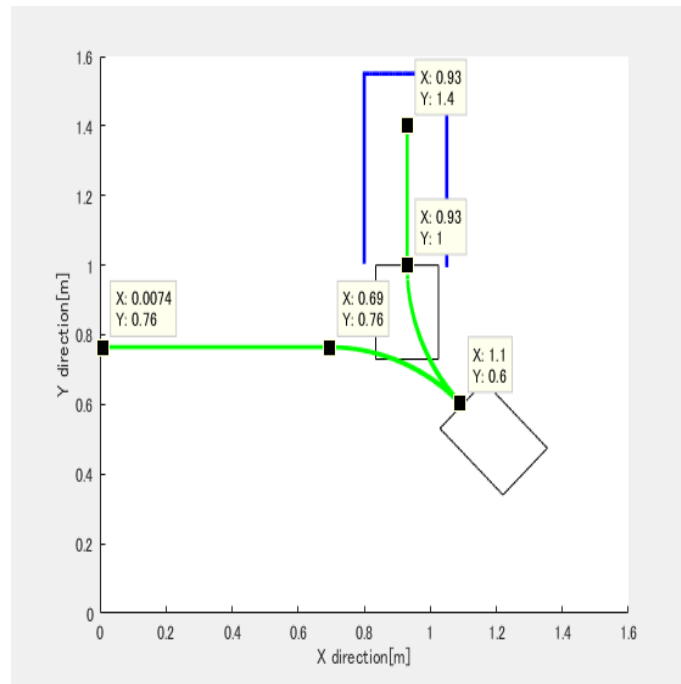


Fig. 13 Parking route result of condition 17

Fig. 10、Fig. 12、Fig. 13 の結果を比較すると、どの経路も駐車完了条件を満たしている結果となった。条件 11、14、17 の経路から最適経路は、スムーズさを考えると最も旋回半径が小さい舵角 30° で旋回している経路である Fig. 13 を選ぶべきであるのだが、本研究で実験的検討を行う車の最大舵角は 30° と設定しているため、万が一経路との誤差が生じ、補正しなければならない場合、対応ができないもしくは遅れるといったことが考えられる。そこで、旋回中の舵角は 25° とし、経路を補正する際に舵角に余裕を持たせるべきであると判断した。以上より駐車枠 1 の最適駐車経路は、Fig. 10 の条件 14 の駐車経路と決定した。

同様の流れで決まった駐車枠 2、3 の最適駐車経路を順に Fig. 14、Fig. 15 に示す。

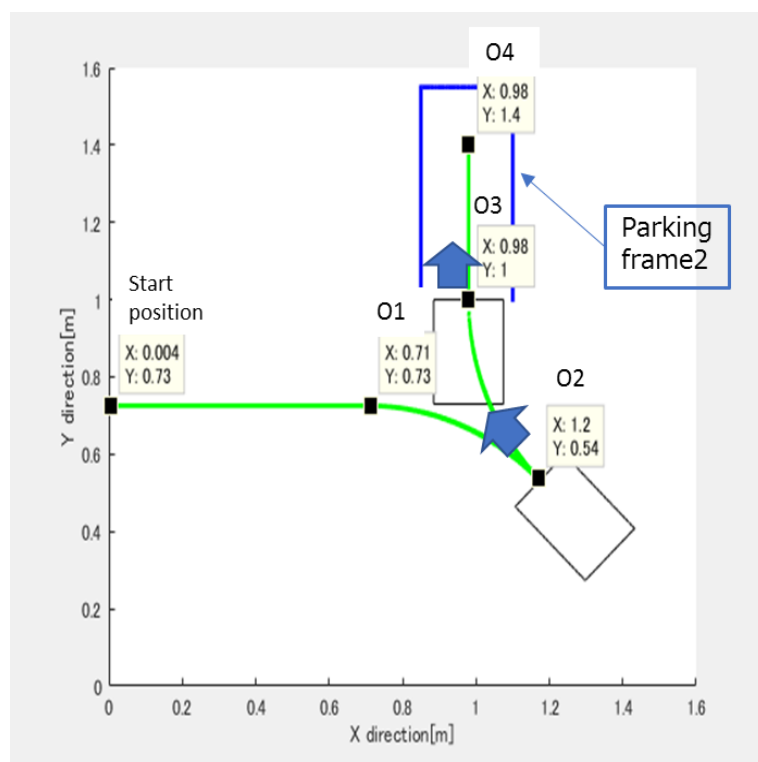


Fig. 14 Optimum parking route at parking space number 2

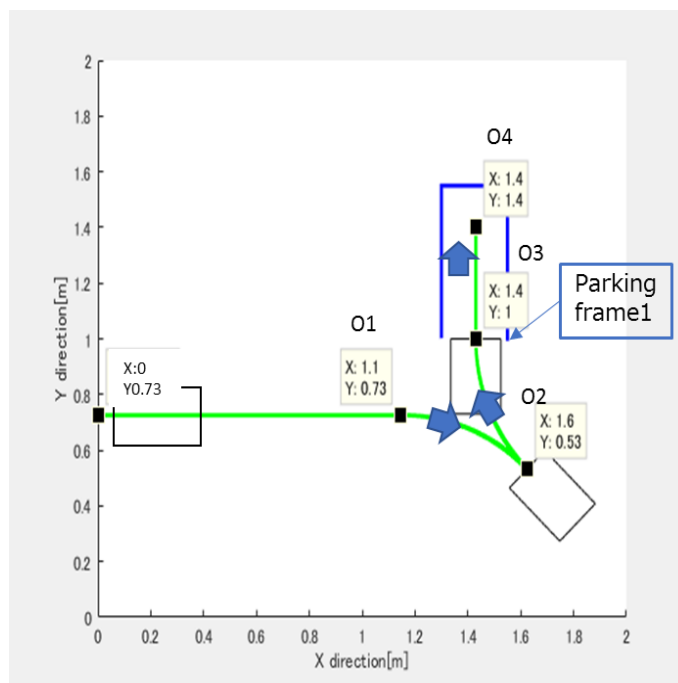


Fig. 15 Optimum parking route at parking space number 3

駐車枠番号 1 から 3 までと経路の形が少し異なる、駐車枠番号 4 から 6 までの最適駐車経路について検討する。駐車動作(点 O1 から点 O4 まで)中の最適経路に加え、目標の駐車枠に接近するまでの方法について考える。方法は、Fig. 16 のように二通りある。

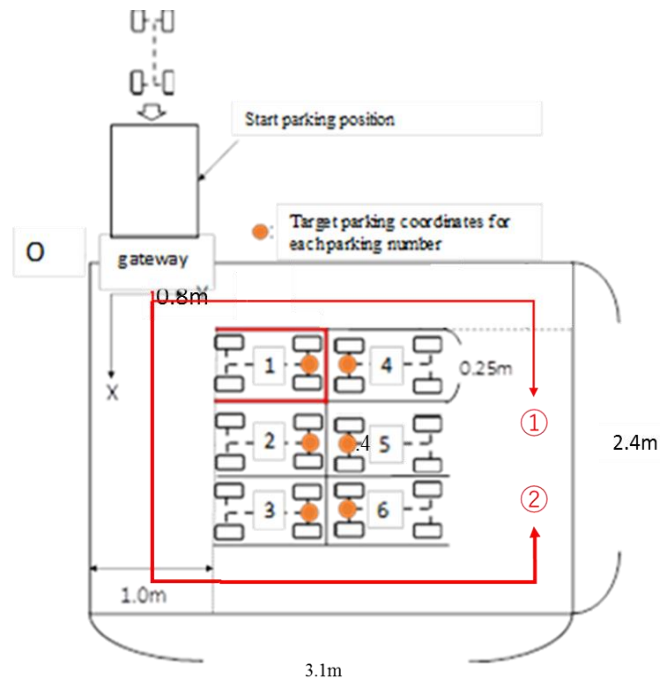


Fig. 16 Approach method for parking space numbers 4 to 6

スムーズに駐車動作を終えることを考えると①の経路を通る方が良く、②の経路の場合経路が長くなるため、駐車完了までに時間がかかり、また万が一誤作動や経路から車が外れた場合、他の衝突する危険性が高くなる。よって、①の経路を通り4から6までのそれぞれの駐車枠に駐車を行う経路を考えた。まず駐車枠4の経路①と経路②の最適駐車経路 Fig. 17 と Fig. 18 に示す。

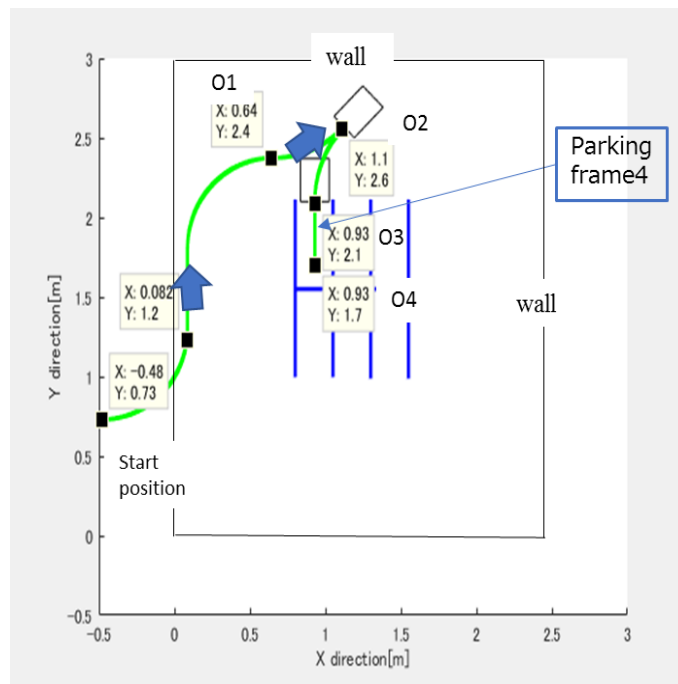


Fig. 17 Parking route with parking space number 4 using route ①

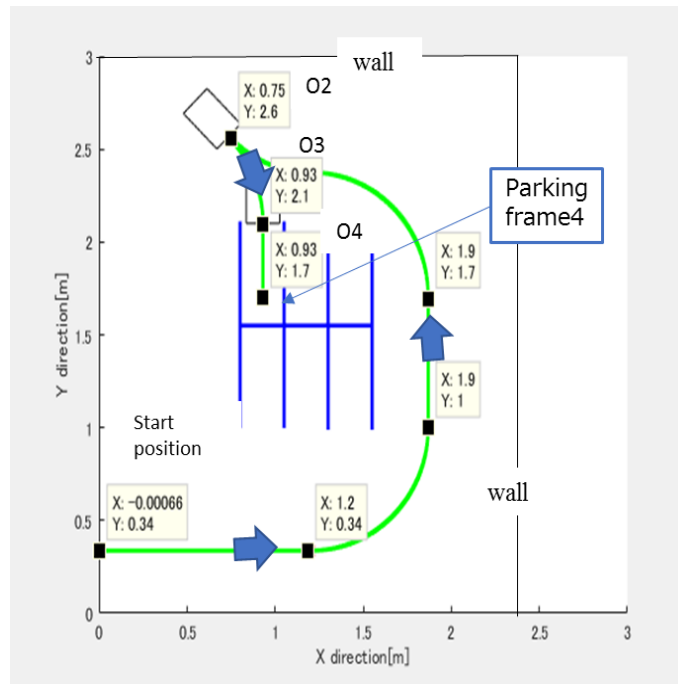


Fig. 18 Parking route with parking space number 4 using route ②

Fig. 17 と Fig. 18 を見てみると、Fig. 16 で示した経路①の方を通る Fig. 17 は、スタート位置が駐車場外にあるため、この経路に従って駐車を行うことは不可能であると判断した。また、経路②を通る Fig. 18 は、経路長は Fig. 17 よりも長くなっているが、経路通りに駐車を行えば駐車枠を踏まず、壁との接触もないため、駐車枠 4 の最適駐車経路は、Fig. 18 であると判断した。もし、駐車場の Y 方向と平行にある通路の部分はいくらか広くスペースがあれば、Fig. 17 の経路の方が最適経路になると考えられる。

次に駐車枠番号 5 における経路①と経路②それぞれの方法で駐車を行ったときの経路設計結果を Fig. 19、Fig. 20 に示す。

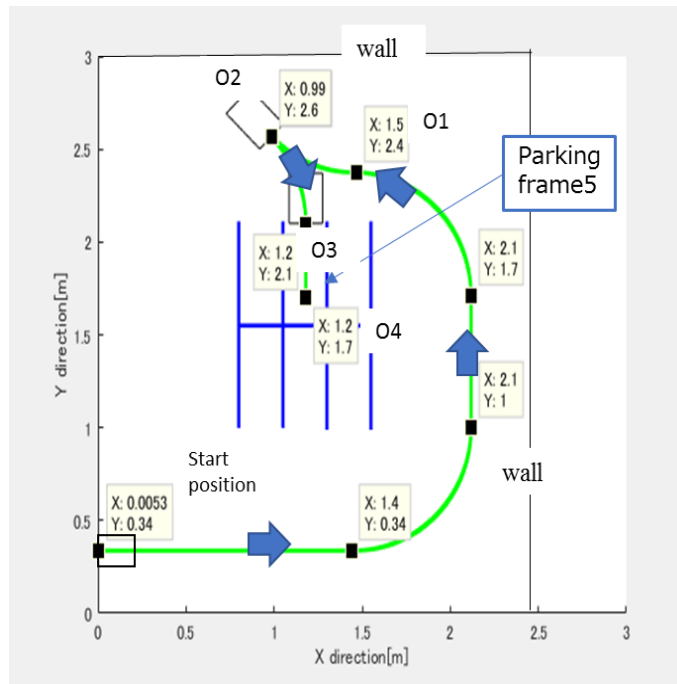


Fig. 19 Parking route with parking space number 5 using route ①

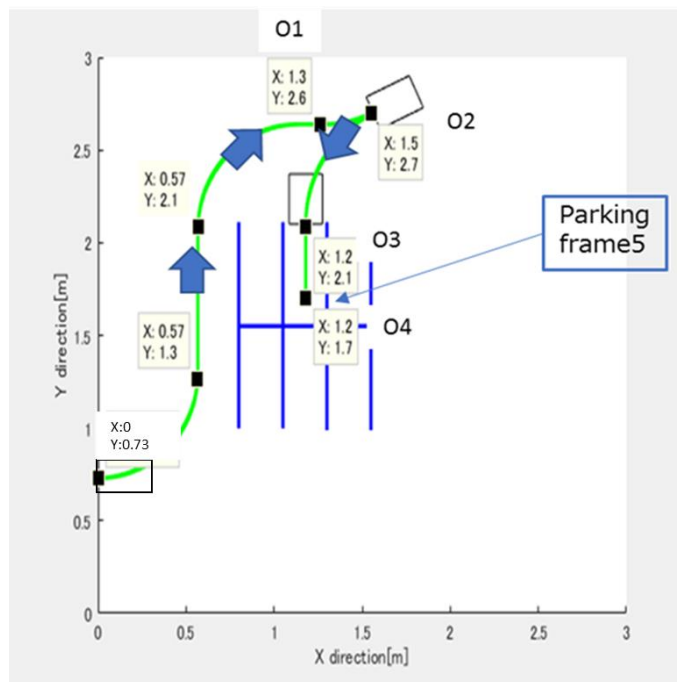


Fig. 20 Parking route with parking space number 5 using route ②

Fig. 19 と Fig. 20 の経路結果を比較すると、どちらも経路に沿って駐車を行えることが可能であることが確認できた。Fig. 20 の旋回中の舵角は 25° 、点 O2 の方向角は 45° であったが、Fig. 19 の点 O1 までの旋回中の舵角は 30° 、点 O2 の車の方向角は 25° であった。また、Fig. 16 における経路②に従う駐車方法であれば、複数通りの駐車完了条件を満たす経路が設計できたが、経路①に従う駐車方法は、Fig. 20 の一つのみであった。駐車枠 5 の場合、舵角や車の方向角の設定値が限定的にはなるが、経路長が短く、駐車場周辺の柵や駐車枠線を踏まない

と考えると Fig. 20 の経路結果が駐車枠 5 の最適経路であると考えた。駐車枠番号 6 に至っても駐車枠 5 と同様の結果となり、経路①の方法で設計した駐車経路結果を Fig.21 に示す。

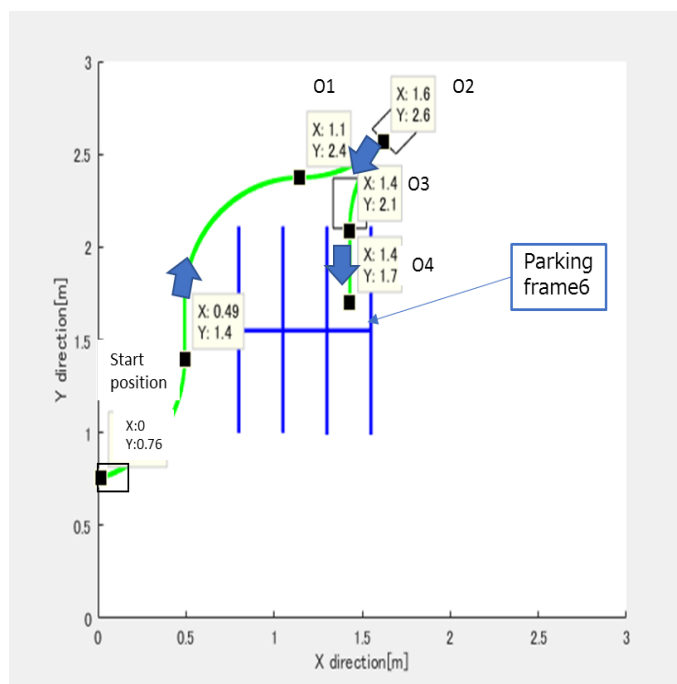


Fig. 21 Optimum parking route at parking space number 6

以上の結果から駐車枠番号 1 から 6 までの最適駐車経路が決定し、駐車場の入口の位置は、駐車枠番号 4 は、Fig. 18 から(0,0.34)の位置、それ以外の番号は、(0,0.73)と分かった。

第 5 章 経路設計結果を用いた RoboCar の走行実験

5-1 実験目的

4-3 で示した最適駐車経路に沿って本当に駐車が可能であるかを判断するために実際に RoboCar の走行実験を行った。そして、最後に最適駐車経路との誤差が生じた際、どれだけ誤差が生じているかを測定するため、RoboCar の実際の走行経路と最適設計経路との比較も行った。

5-2 実験方法

4-2 の Fig. 6 で提案した駐車場を実際に作成した後、経路設計結果から算出されたスタート位置から目標駐車枠まで走行実験を行う。実走行経路のデータは、RoboCar の後輪軸の中心にマーカーを置き、Fig. 22 の Optitrack 製モーションキャプチャー V120 Trio_Duo を用いてマーカーの座標を得ることから行う。速度は低速の 0.3m/s、舵角と走行経路の長さの情報は、Fig. 10 の設計駐車経路結果を基にする。走行は 3 回行い、目標駐車枠番号は 1 のみとする。設計経路と走行経路の比較を行う。



Fig. 22 Optitrack V120 Trio_Duo

5-3 実験結果および考察

Fig. 10 の設計経路情報を基に走行実験 3 回行った結果を以下の Fig. 23、Fig. 24、Fig. 25 に示す。

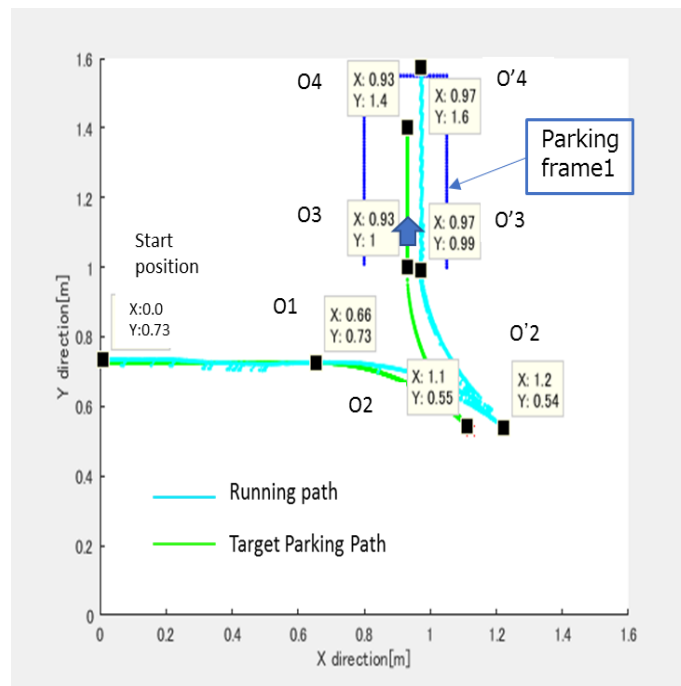


Fig. 23 Travel experiment 1

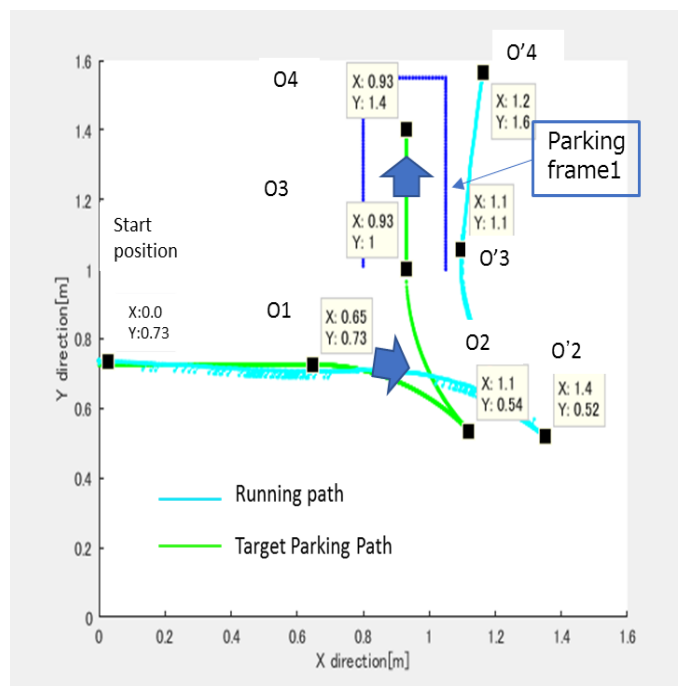


Fig. 24 Travel experiment 2

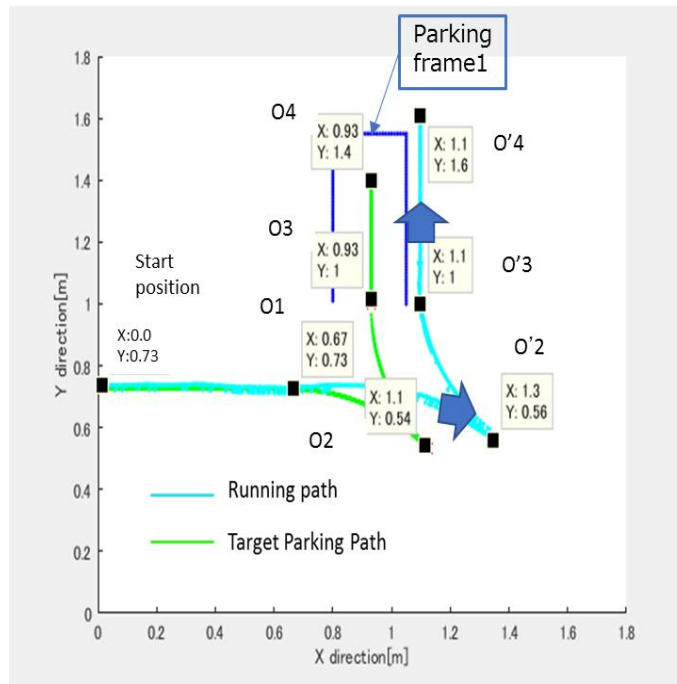


Fig. 25 Travel experiment 3

Fig. 23 から Fig. 25 を見てみると、どの結果も点 O1 の前進旋回開始の時から目標駐車経路と実走行経路とに誤差が生じ初めており、点 O4 の位置においては、X 方向の座標の誤差は最大で 0.3m であり、誤差の許容範囲内であるが、Y 方向に至っては、最大で 0.2m の誤差が生じていることがわかった。さらに駐車枠と平行に停車できなかった場合もあり、結果的に大きく経路から外れ、誤差が許容以上になってしまった。Fig. 23 の点 O4 の停止位置では駐車枠の線を踏んで停車していることから周りに駐車している車がある場合、接触する危険性がある。また、誤差が生じた原因としては、路面摩擦やタイヤのすり減り具合、重量による走行抵抗が考えられる。

以上の結果を踏まえて、駐車動作中に目標駐車経路を修正しつつ、より目標駐車位置に近い位置で駐車完了できるようにしなければならないと考えた。

第 6 章 誤差補正

6-1 誤差補正概要

設計した経路に従って走行を行うが、天候等の状況で経路から外れ、目標の駐車枠に駐車できない可能性も考えられる。例として、4-3 で最適とした条件 14 の駐車経路結果と点 O2 まです実際に RoboCar を走行させたときの経路を照らし合わせたものを Fig. 26 に示す。

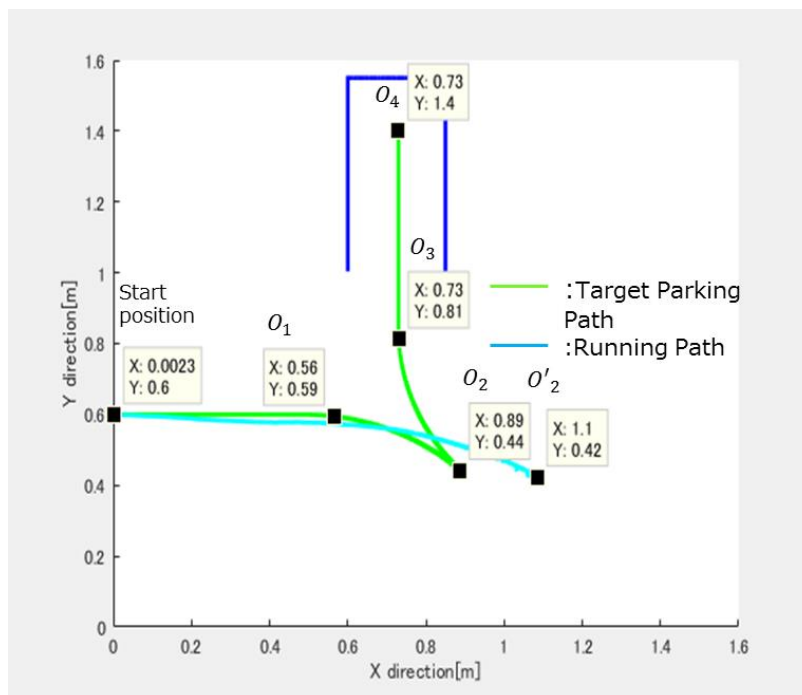


Fig. 26 Route design example

Fig. 26 を見ると点 O2 の座標は、経路設計実験時は(0.89,0.44)であったが、走行実験時は、(1.1,0.42)となり、X 方向においては約 0.2m もの誤差が生じた。このように全て設計経路通りに車が動くとは限らないので設計経路との誤差を検知した際に補正をかける。補正をかけるかどうかの判断は、駐車経路上の点 O2 と点 O3 において一旦停止後、駐車場の上部に設置されているカメラの情報を基に判断する。このような誤差補正方法を選んだ理由としては、誤差を補正する回数をなるべく少なくし、車における無駄な操作を減らすことができると考えたためである。また、点 O1 から点 O3 の内、点 O2 と点 O3 において誤差補正をかけることを選んだ理由については、点 O3 において車の方向角が駐車枠とほぼ平行でなければ駐車完了条件を満たすことが難しくなるため、点 O4 に直結する点 O3、そして車の方向角が大きく変えることができる点 O2O3 間を走る前の点 O2 で修正するべきであると考えたためである。次項から点 O2 と点 O3 の場合に分けて誤差を補正する方法を順に説明する。

6-2 点 O2 における誤差補正方法

点 O2 における誤差補正方法は、点 O2 に車が来た時に車の現在位置から点 O4 までの駐車経路の再設計を行う。概略を Fig. 27 に示す。

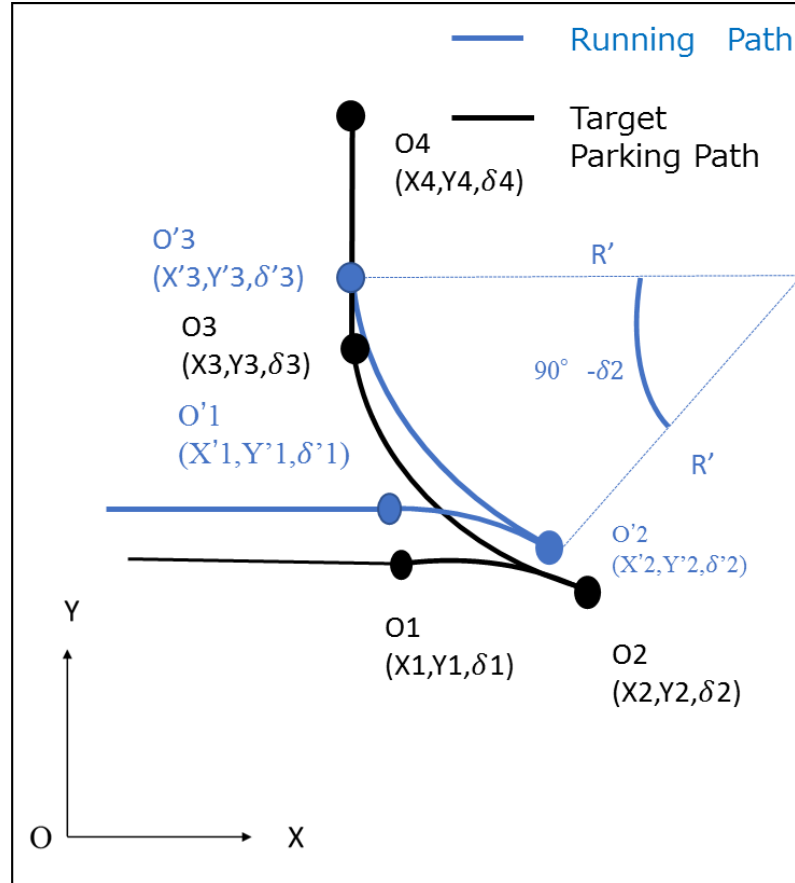


Fig. 27 Error correction method for O 2

点 O1 から O4 までを通る目標駐車経路に対し、車が O'1 から O'2 まで進んだとする。経路の再設計の方法であるが、車の現在位置情報(X'_2, Y'_2, δ'_{22})と点 O4 の座標値から点 O3 を算出し、その経路における車の旋回舵角 θ'_r を決める。点 O3 から点 O4 まで直進後退することを考え、

$$\theta'_r = \tan^{-1} \frac{l(1 - \cos \delta'_{22})}{(X'_2 - X_4)}$$

と表すことができる。

ただ、旋回半径が範囲外になる、仮に範囲内であっても点 O'3 が駐車枠の上部に位置し、駐車枠からはみ出してしまう場合も考えられる。その場合、舵角 30° の円旋回経路と舵角 0° の直線経路を組み合わせた駐車経路を考える。例として、車が後退旋回を開始する際に駐車枠番号 3 の前付近に停車した際の車の動きを以下の Fig. 28 に示す。

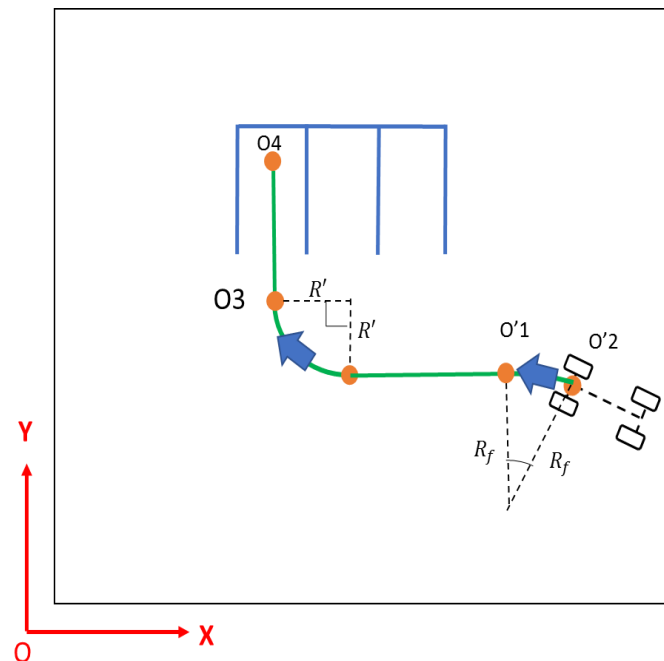


Fig. 28 Path improvement overview in 7-2.2

Fig. 28 のように、前進旋回時と同じ舵角で通った経路をなぞる様に後退旋回を行い、旋回を始める前の点 O'1 に戻った後直進後退を行い、目標の舵角で車の方向角が 90° になるように後退旋回をもう一度行い、目標駐車位置まで直進後退するという経路である。その他にも最初に設計した経路に合わせるよう、駐車場の周りを一周しスタート位置付近まで戻する方法や円旋回経路を組み合わせるなども考えたが、経路が長い、舵角操作が複雑になるなど、スムーズに駐車を完了するためには、この経路の形が良いと考えた。

6-3 点 O3 における誤差補正方法

点 O2 で誤差が 0.1m 以下であるもしくは、O2O3 間のみで誤差補正が完了出来ないと判断した場合、点 O3 で誤差補正を行う。点 O3 については、目標駐車経路上で点 O4 に向かい直進後退を始める点であり、駐車枠を踏んではいけないことも考えると補正を行うのが難しい状況である。今回、提案した補正方法は、左定常円旋回と右定常円旋回を組み合わせた経路で、二つの円共に円周角と半径が等しいものである。概略を Fig. 29 に示す。

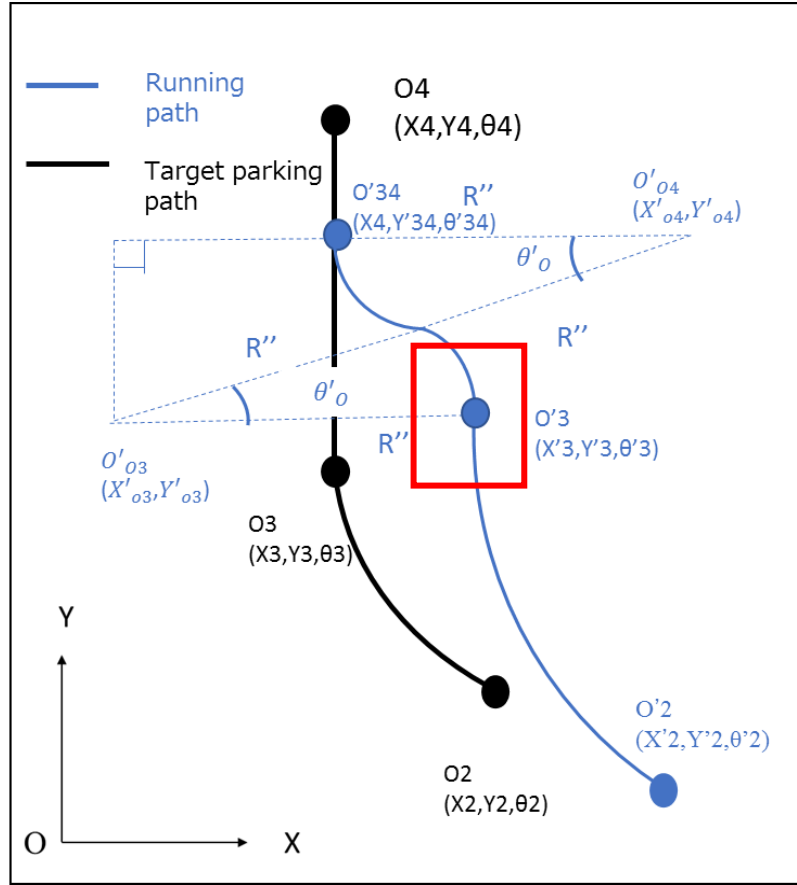


Fig. 29 Error correction method for O 3

点 O'2 から後退で走行してきた車が点 O'3 に来た際、直進後退を行っても目標の駐車枠に辿り着けない場合、Fig. 31 のように Y 方向における平行移動のような形で経路修正を行う。

例として、 $\theta'_3 = 90^\circ$ 半径を R'' と仮定すると、点 O'3 の座標は、

$$X'_{o3} = X'_3 - R''$$

$$Y'_{o3} = Y'_3$$

となる。

また、二つの円の中心の X 座標 X'_{o4}, X'_{o3} と円周角 θ'_{34} から

$$X'_{o4} - X'_{o3} = 2R'' \cos \theta'_o$$

が成り立つ。すると、点 O'34 の Y 座標 Y'_{34} は、

$$Y'_{34} = Y'_{o3} + 2R'' \sin \theta'_{34}$$

となる。

上部の場合、車の方向角が駐車枠と平行の 90° であるが、次に 90° 以外の場合を考える。この場合は、車の方向角を一度駐車枠と平行にしてから補正をかける。 $X_4 < X'_3$ の時を例に概図を Fig. 30 に示す。

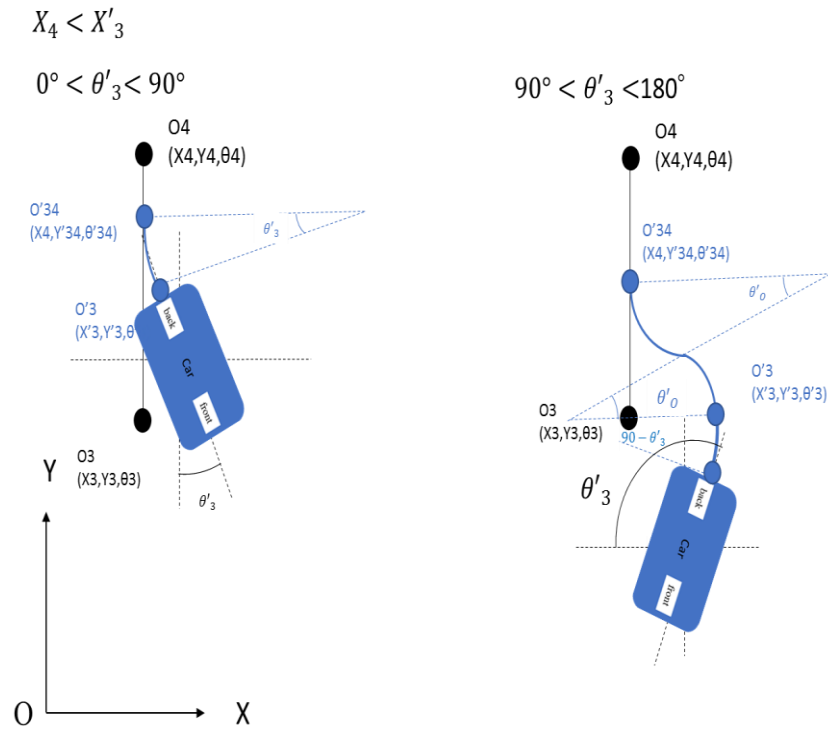


Fig. 30 Details of error correction method at O 3

なお、 $X_4 > X_3$ の場合、Fig. 30 の経路の概図が $90^\circ < \theta_3 < 180^\circ$, $0 < \theta_3 < 90^\circ$ で反対となる。

第7章 誤差補正を含んだ経路設計実験

7-1 実験目的

6-2 と 6-3 で提案した誤差補正方法で実際にどのような場合でも経路を修正できるかどうかを検証するため、補正前と補正後の経路をシミュレーションし、比較を行う。

7-2 実験方法

今回の実験では、5つのシチュエーションを考え、生成された経路で補正前と補正後の経路共に、駐車完了条件を満たしながら同じ目標駐車位置に到達できるかどうかを4章同様、数値解析ソフトウェア matlab で評価する。提案した五つの状況の詳細を以下に示す。

7-2.1. 前進旋回時の舵角が 25° と設定していたが、舵角が 20° でしかできなかった場合

7-2.2. 駐車枠番号1番に駐車する予定の中、後退旋回を始める位置が以下の Fig. 31 のように駐車枠番号3側の端に来た場合

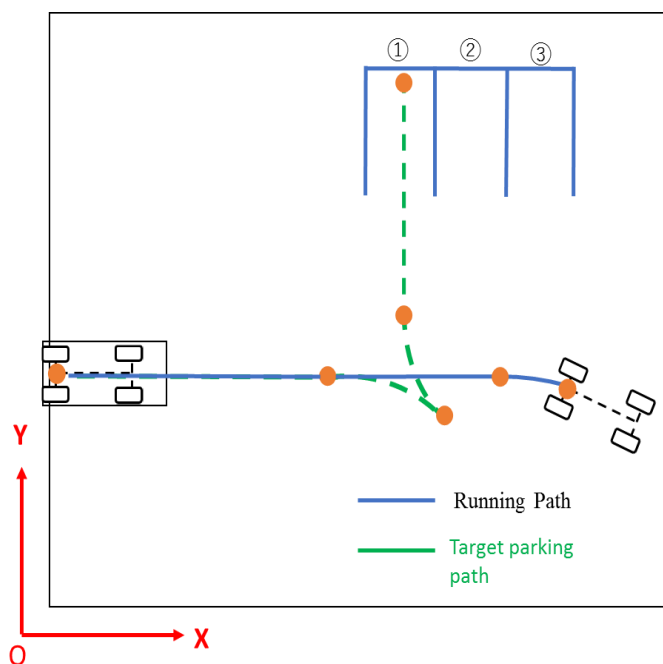


Fig. 31 Example of situation in 7-2.2

7-2.3. 駐車番号1に駐車動作中、点O3において車の方向角は駐車枠と平行の 90° であるが、直進後退をしても駐車完了出来ない場合。

7-2.4. 点O3で誤差が生じ、車の方向角が 90° 以外であった場合

7-3 実験結果および考察

7-2.1 の状況について、経路設計実験を行った。設計結果を Fig. 32 に示す。

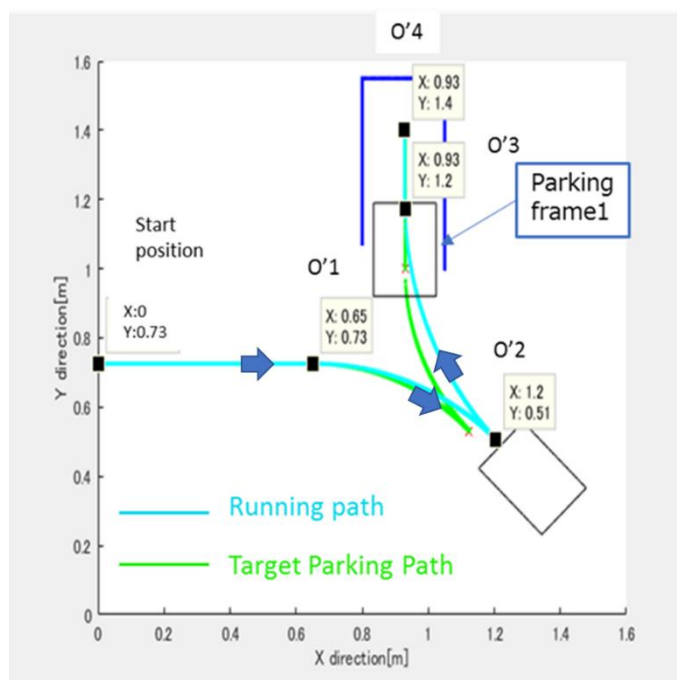


Fig. 32 Path design result of 7-2.1

それぞれの経路の座標点と旋回半径を比較しやすいように以下の Table 4 に示す。

Table 4 Details of 7-2.1

	O1	Radius during turning[m]	O2	Radius during turning[m]	O3	O4
Target parking path	(0.66,0.73)	0.66	(1.12,0.54)	0.66	(0.93,1)	(0.93,1.4)
Redesign parking path	(0.66,0.73)	0.8	(1.22,0.49)	0.92	(0.93,1.16)	(0.93,1.4)

Fig. 32 と Table 4 を見てみると、再設計を行った駐車経路でも駐車枠番号 1 の目標駐車位置に辿り付けることがわかる。最初に設計を行った目標駐車経路では、前進旋回中、後退旋回中共に舵角は 25° であったが、再設計を行った後の駐車経路では、舵角 25° で前進旋回した後、6-2 の誤差補正方法を用いて、舵角が 16° となった。また、再設計後の経路に従って走行を行っても壁との接触や駐車枠の線を踏むことはなく、よって 7-2.1 の状況の場合、6-2 で述べた誤差補正方法に従って駐車を完了させることは可能であると判断した。

次に 7-2.2 の状況について、経路設計実験を行う。図の通り駐車枠番号 3 付近に位置する (2.0,0.54) に前進旋回終了後停止したと仮定し、後退旋回を始めようとする場面である。

まずは、(2.0,0.54) の位置から、誤差補正方法 2 を用いて経路を再設計した経路設計結果を Fig. 33 に示す。

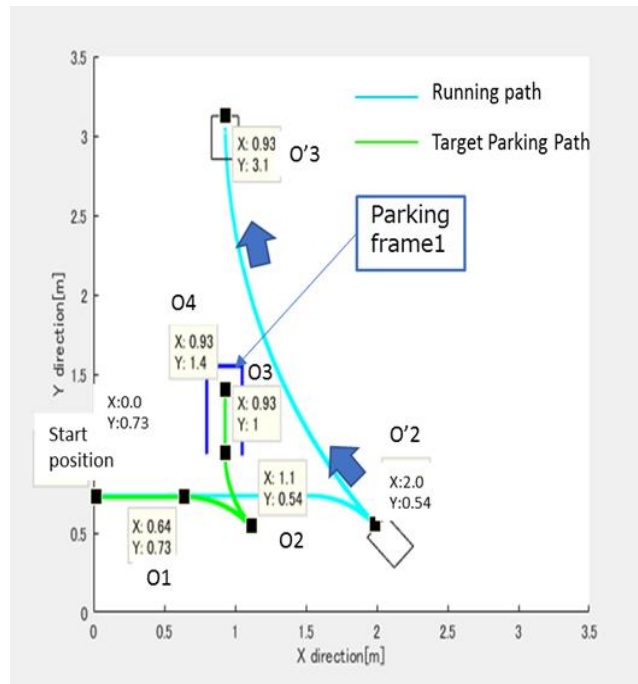


Fig. 33 Path design result of 7-2.2

Fig. 33 の結果の通り、再設計した経路が目標駐車枠から大きく外れてしまう結果となった。無論、この経路に従って駐車を行うことは出来ないため、今回の補正方法では、このような状況の場合駐車不可能であると判断した。そこで6-2の後半で提案した舵角 30° の円経路と舵角 0° の直線経路を組み合わせた駐車経路を考える。

実際に Fig. 28 を用いて経路設計を行った結果を Fig. 34 に示す。

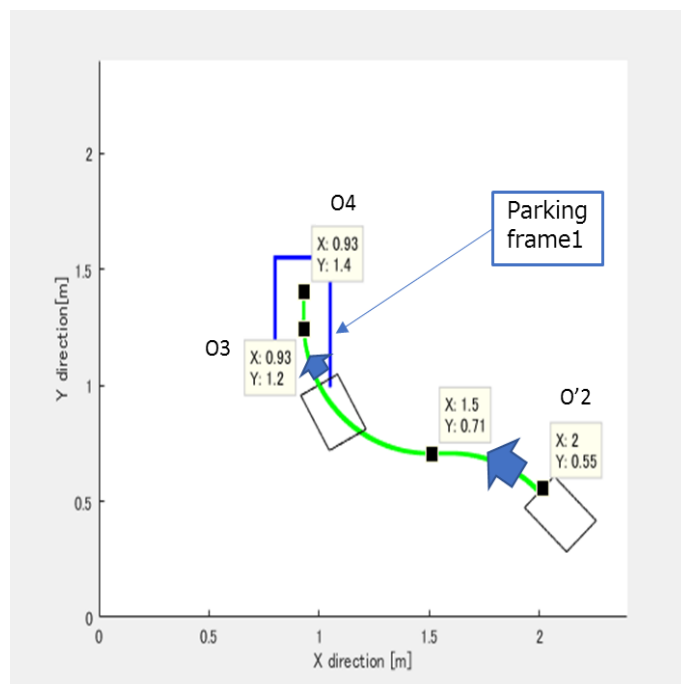


Fig. 34 Path design after improvement 7-2.2

Fig. 34 を見ると、円旋回、直進後退、円旋回を経て、目標駐車位置の点 O4 に駐車できることが確認できる。しかし、駐車枠線を踏んで駐車を行う可能性があり、駐車完了条件を満たすことは出来ない。その他の駐車経路も検討してみた所、効率よくスムーズに駐車を行うことができる経路はこの他にないと判断したため、このような場合車にステレオカメラやセンサから駐車枠線や周りの状況を認識しつつ、踏みそうになると切り返すといった駐車方法で駐車を完了させるしかないと考えた。

次 7-2.3 の状況を考える。この場合、6-3 で示した誤差補正方法を用いて、目標駐車位置の点 O4 までの経路を設計する。目標の駐車枠は、1 番、点 O4 の座標は(0.93,1.4)、点 O3 の座標は、(0.93,1.0)とし、現在車が O'3(1.08,0.7)の位置に停車していると考ええる。概図を以下の Fig. 35 に示す。また経路設計結果を Fig. 36 に示す。

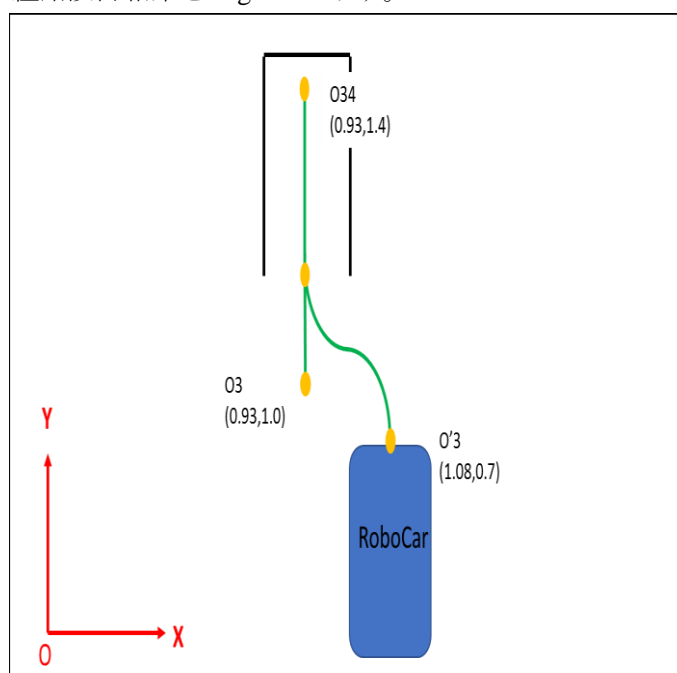


Fig. 35 Status of 7-2.3

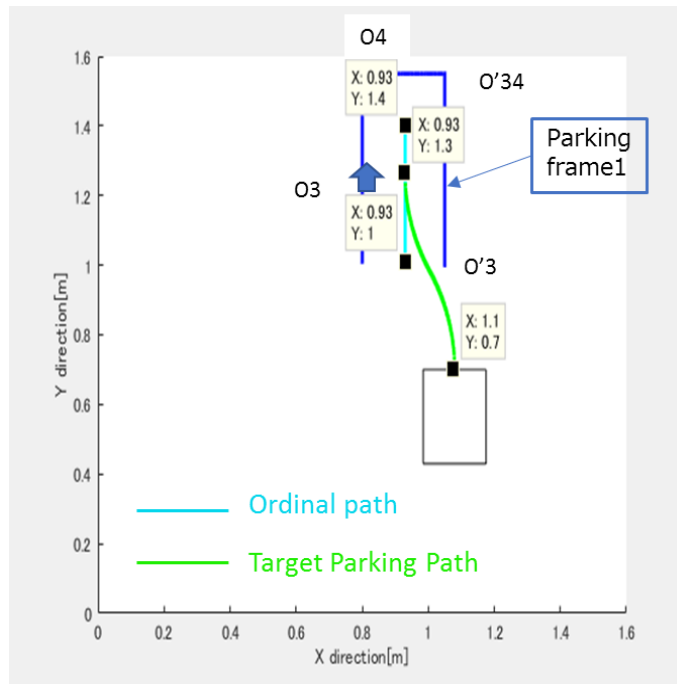


Fig. 36 Path design result of 7-2.3

Fig. 36 の結果を見ると、舵角 30° を用いて、6-3 で提案した補正方法で駐車ができることがわかる。この時、旋回中の円周角は、共に 30° であったことから車の方向角は、 $90^\circ \rightarrow 60^\circ \rightarrow 90^\circ$ と変化していることがわかった。また、少し際どいが駐車枠線を踏まずに駐車完了できることがわかり、今回のような状況では、この経路で駐車を完了させることが最適であると判断した。

次に O'3 の車の方向角が 90° ではない場合を考える。今回の条件は、点 O4 の座標は (0.93,1.4)、点 O3 の座標は、(0.93,1.0)とし、現在車が O'3(1.1,0.7)の位置に停車するとし、点 O'3 の車の方向角は、 45° と点 O4 の座標を(0.73,1.4)、現在の車の位置点 O'3 を(1.1,0.2)とし、車の方向角を 135° とする。

経路設計実験結果をそれぞれ Fig. 37、Fig. 38 に示す。

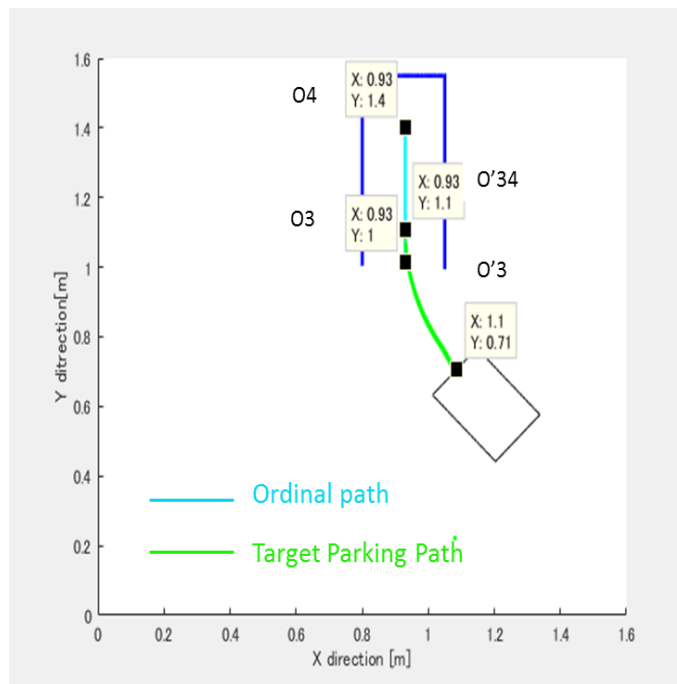


Fig. 37 Path design result at 45°

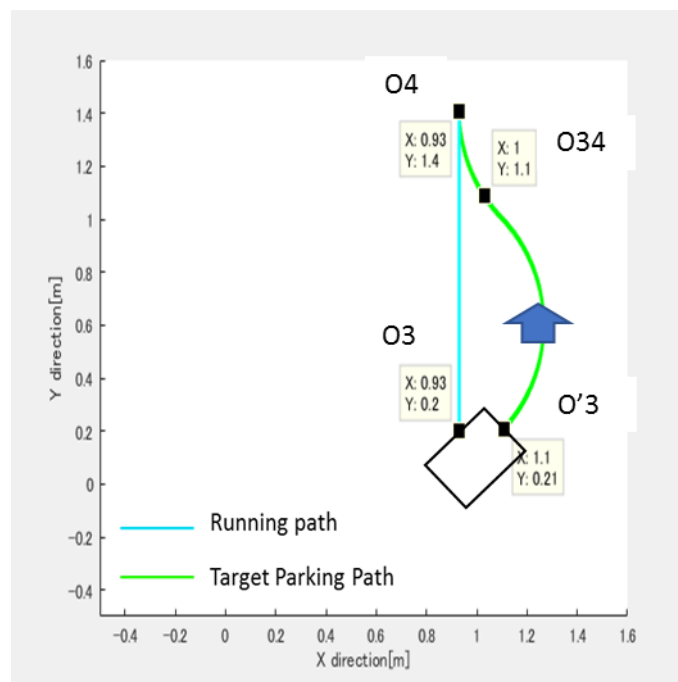


Fig. 38 Path design result at 135 °

Fig. 37 から、車の方向角が 90°以下であれば、円旋回のための経路となり、経路も短くスムーズに駐車できることがわかった。

一方、Fig. 38 の結果を見ると、非常に湾曲した経路となり、経路長も 1.32m と比較的長い経路となった。この 7-2.4 で提案した条件は、駐車場の周りの柵に接触してしまうのであり得ないが、仮に点 O3 が駐車枠から離れており、車の方向角が 90° 以上である場合、Fig.38 のような経路を取らなくてはならず、方向角の変化も大きいことから実際に駐車する際、駐車

する際の車との接触や駐車枠を踏む可能性が高い。そのため、Fig.38 のような経路を取る際、センサやステレオカメラを用いて周囲の状況を判断しつつ、駐車を完了させる必要があると考えた。

第 8 章 誤差補正検証のための RoboCar の走行実験

8-1 実験目的

第 6 章で提案した誤差補正方法を実際に RoboCar に組み込ませて、その有用性を検証する。

8-2 実験方法

RoboCar を第 5 章同様、2 回ずつ走らせ、モーションキャプチャーによって走行経路を測定し、補正後の設計経路と走行経路と比較を行う。走行を行う経路は、以下に示す。

8-2.1. Fig. 10 の経路

8-2.2. Fig. 32 の経路

8-2.3. Fig. 33 の経路

8-2.4. Fig. 36 の経路

8-2.5. Fig.37 の経路

8-2.6. Fig.38 の経路

以上、6 種類の経路設計データを基に走行実験を行う。

8-3 実験結果および考察

実際に走行させ、実測した経路を順に示す。まず、8-2.1 の経路に従って、得た走行結果を示す。この経路は、前進時後退時のそれぞれの旋回舵角は 25° である。経路設計結果と実測結果の比較を Fig. 39 に示す。

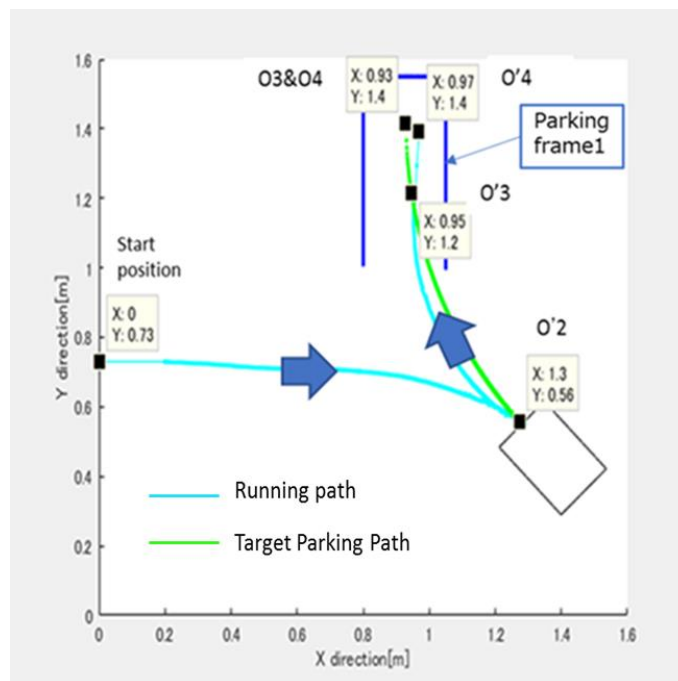


Fig. 39 Running experiment 1 for condition 8-2.1

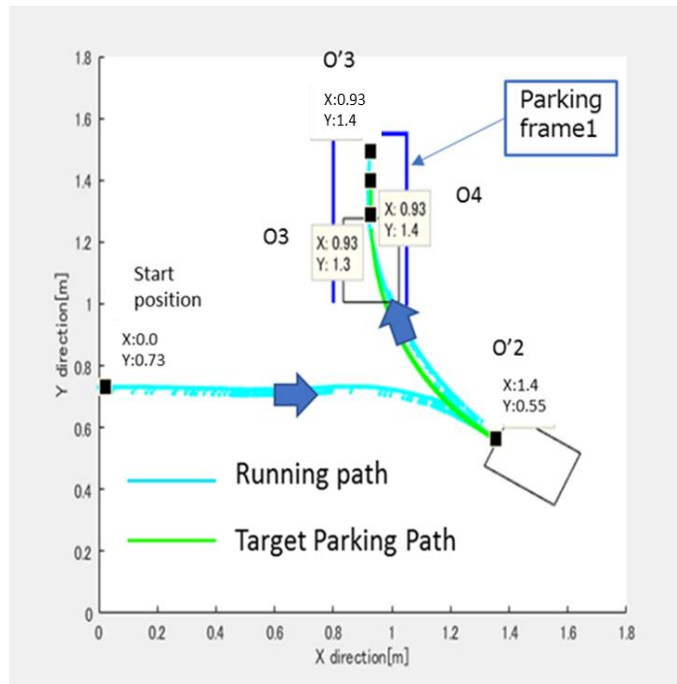


Fig. 40 Running experiment 2 for condition 8-2.1

Fig. 39、Fig. 40 の結果を見てみると、二回行った実験で Fig. 40 のように目標の駐車枠にぴったりと駐車完了出来たものもあれば、Fig. 39 のように駐車枠内に収まったものの、車の方向角と駐車位置共に少しずれが発生したものもあった。ただ誤差が許容範囲内であるため今回の場合、補正を用いて駐車を完了させることができたと考えた。

次に 8-2.2 の経路について、比較を行う。実測経路と設計経路の図を Fig. 41、Fig. 42 に示す。

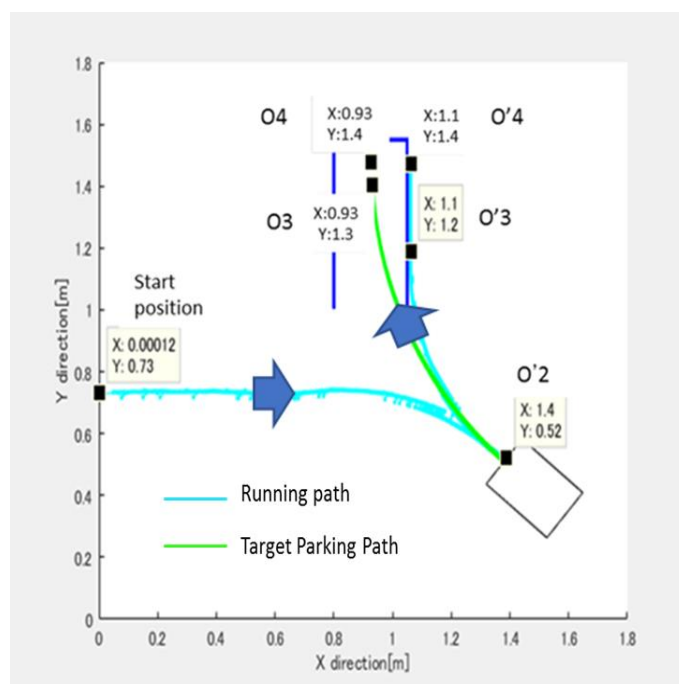


Fig. 41 Running experiment 1 for condition 8-2.2

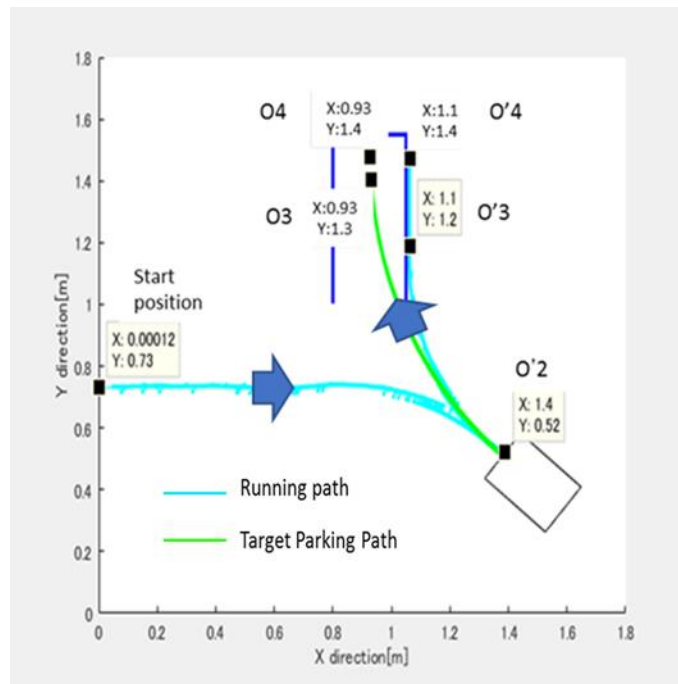


Fig. 42 Running experiment 2 for condition 8-2.2

Fig. 41、Fig. 42 の結果を見てみると、2 回の実験共にほぼ同じような結果となり、駐車枠線上で RoboCar が停止していることがわかる。舵角は、7-2.1 の Table 4 に示した 0.8m と 0.92m の舵角が切れていたが、車にかかる抗力から主に旋回中に誤差が生じ、誤差も X 方向に 0.17m 生じた。この場合も同様にセンサやステレオカメラによる誤差補正で最終的に駐車を完了させる方が最適であると考えた。

次に 8-2.3 の経路について、比較を行う。実測経路と設計経路の図を Fig. 43、Fig. 44 に示す。

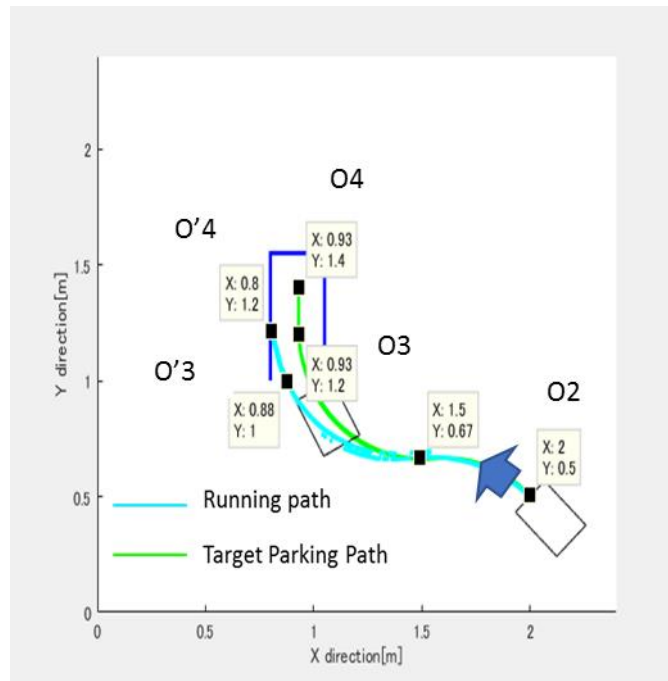


Fig. 43 Running experiment 1 for condition 8-2.3

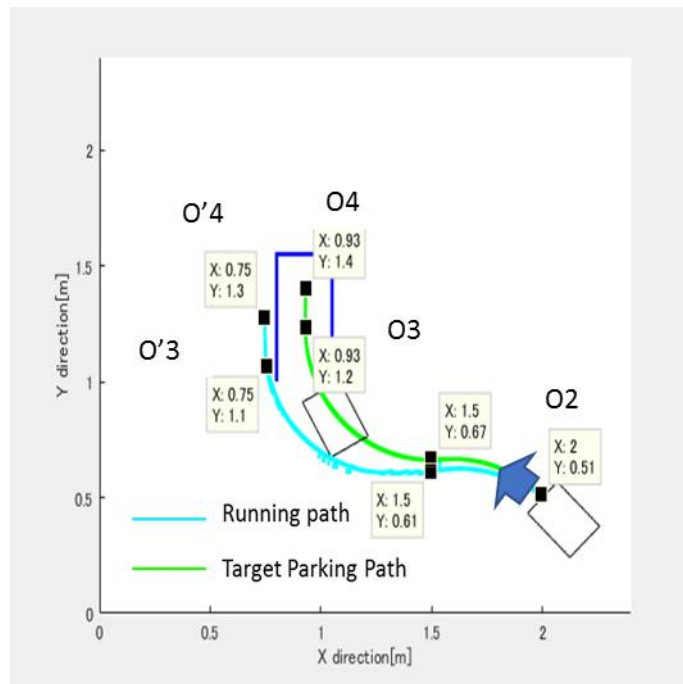


Fig. 44 Running experiment 2 for condition 8-2.3

Fig. 43、Fig. 44 の結果を見ると、この経路も 8-2.2 同様駐車枠の外で駐車完了していることがわかる。対応策としても、途中経路の誤差を無視し、最終的な駐車枠に駐車完了を考え、8-2.2 同様カメラやセンサを用いて駐車を完了させることが考えられる。また、8-2.1 から 8-2.6 まで実際に走行実験を行ってみた所、経路が長くなるにつれ誤差も大きくなることから経路長が長い場合、2 点で補正を行う局所的な補正よりもリアルタイムで補正を行う方が、よりスムーズに駐車を完了できることがわかる。

次に 8-2.4 の場合の実験結果を Fig. 45、Fig. 46 に示す。

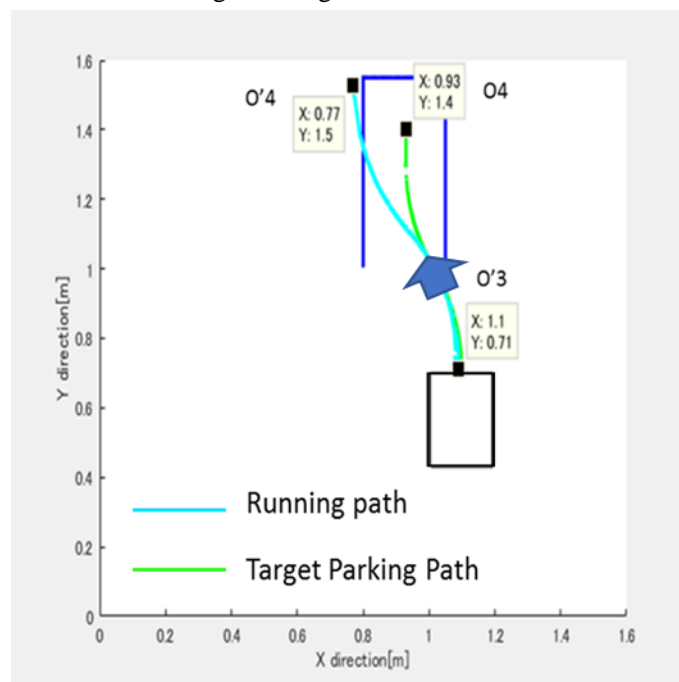


Fig. 45 Running experiment 1 for condition 8-2.4

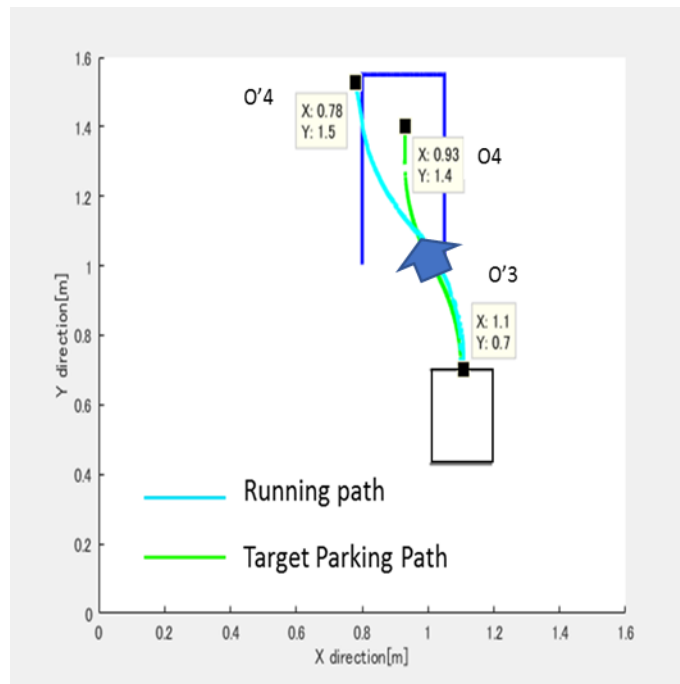


Fig. 46 Running experiment 2 for condition 8-2.4

Fig.46 と 47 の結果から、後退旋回における右に舵角を切って後退する左旋回時の誤差はほとんどなかったものの、左に舵角を切った時における右旋回時において誤差が大きく生じたことがわかる。原因として考えられるのは、左旋回から右旋回へと間なく舵角を切ったことから少し余分に走行してしまったことが考えられる。このような場合舵角を変化させる際に一度舵角を右に $30^\circ \rightarrow 0^\circ \rightarrow$ 左に 30° のように舵角を変化させる際に少し余裕を持たせた方がよいと考えられる。

また、停止時に駐車枠をはみ出してしまったことから、今回の場合も最終的にはセンサやステレオカメラを持ちつつ、駐車を完了させる方がよいと考えた。

最後に 8-2.5、8-2.6 のときの実験結果を Fig. 47 から Fig. 50 までに示す。Fig. 47、Fig. 48 は、点 O'3 が(1.1,0.7)、車の方向角が 45° の時の走行結果、Fig. 49、Fig. 50 は、点 O'3 が(1.1,0.7)、車の方向角が 135° の場合の走行結果を示す。

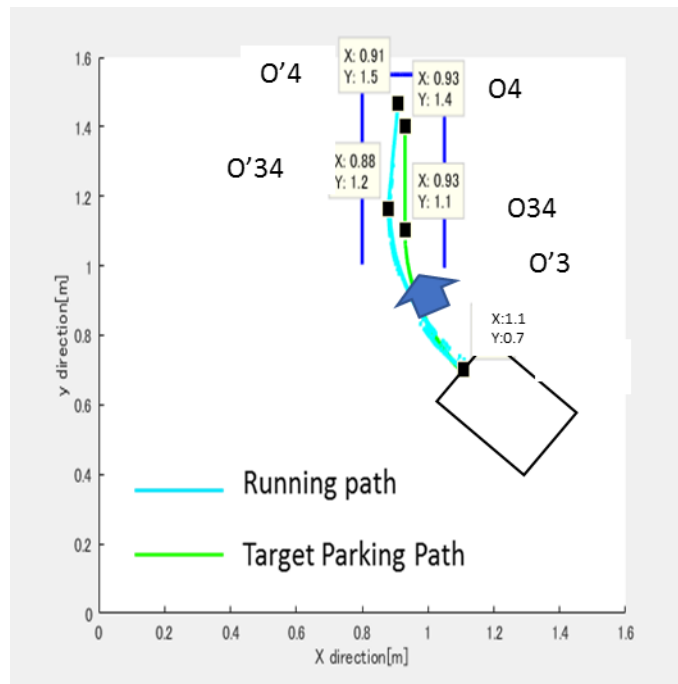


Fig. 47 Running experiment 1 for condition 8-2.5

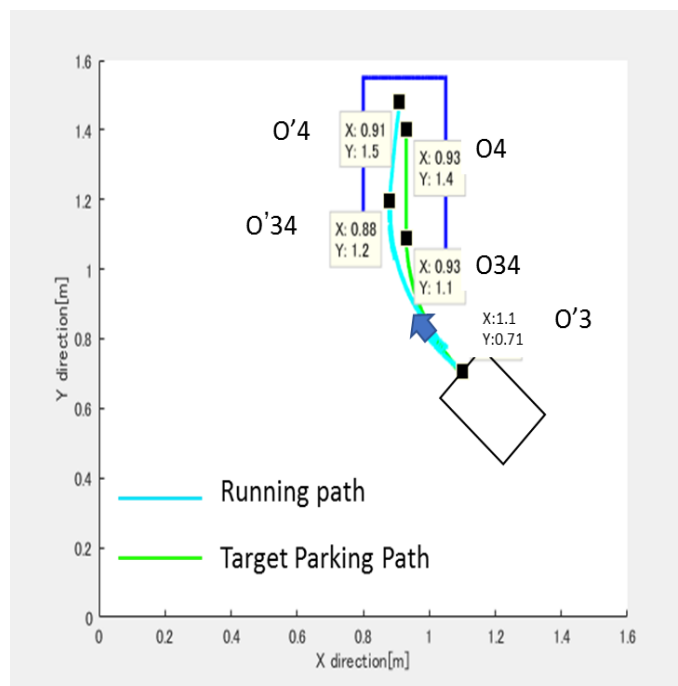


Fig. 48 Running experiment 2 for condition 8-2.5

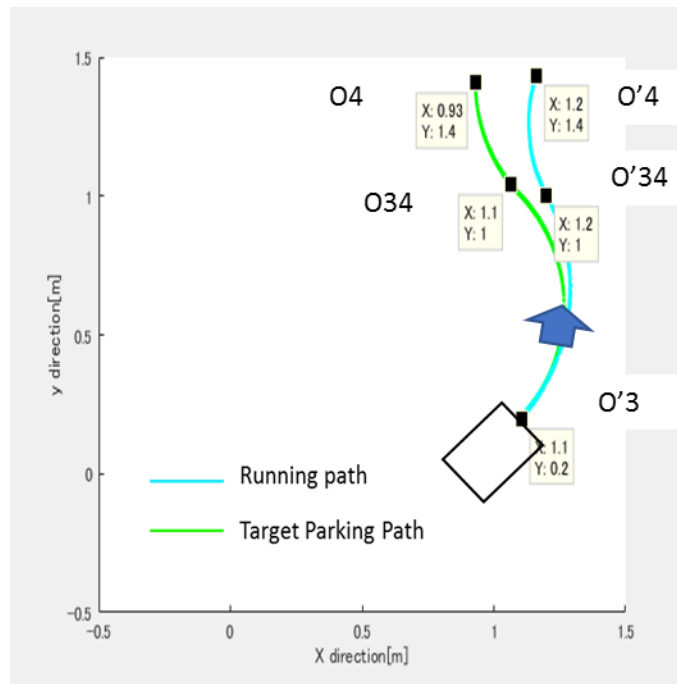


Fig. 49 Running experiment 1 for condition 8-2.6

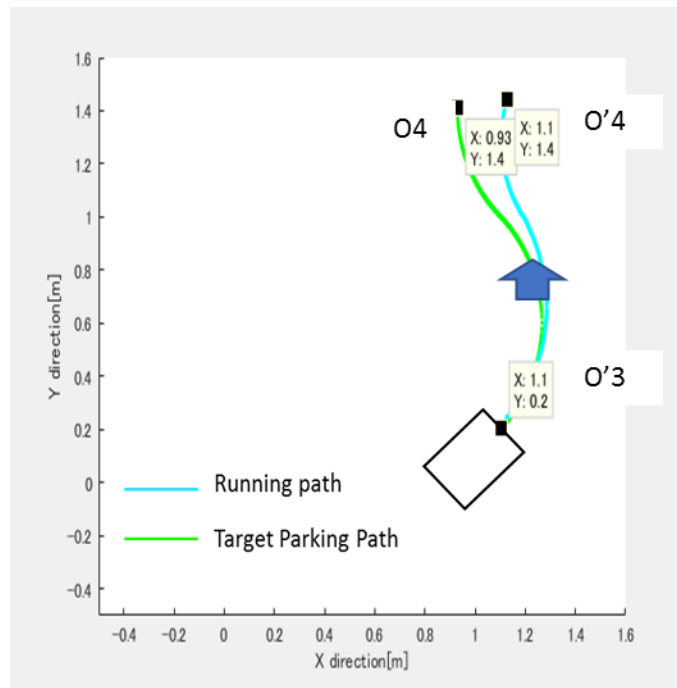


Fig. 50 Running experiment 2 for condition 8-2.6

Fig. 47 から 51 までの結果を見ると方向角 45° の際は、目標経路に沿うように駐車を完了させることができたと考えている。ただ、経路長が長い Fig. 49、Fig. 50 のような経路長が長くなる誤差補正経路であれば、誤差が積み重なり最終的な目標位置において誤差が大きくなっていく結果となった。

以上 8-2.1 から 8-2.6 の状況において、誤差補正方法を実装実験の面から検証を行ったが、誤差を補正できる状況もあったが、補正できず目標駐車枠から大きく外れて停車した状況もある結果となった。

第9章 まとめ

9-1 結言

今回、自動駐車専用の自動駐車システムを提案するにあたり、駐車方法とそれぞれの駐車目標における最適駐車経路の選定方法、さらに目標経路との誤差が生じた際に、車の進行方向の修正を行う誤差補正方法についての提案と RoboCar を用いた実装実験を行った。最適駐車経路については、舵角、車の方向角、点 O3 の座標の三つのパラメータを変化させた際に様々な経路が生成されるが、あらかじめ定めた駐車完了条件や提案した駐車場の大きさを考慮すると駐車経路が一つに絞ることができた。また、誤差補正方法については、特定の条件で提案した補正方法で補正可能であることがわかったが、一部の条件においては、駐車完了出来ないものもあった。

9-2 課題

今回提案した誤差補正方法について、状況に応じて誤差をさせることが出来たものの誤差が最大で 0.5m 程度あったものもあり、どのような状況においても完全に駐車完了条件を満たすことは出来なかった。そのため、最終的にはステレオカメラやセンサなど外界認識しつつ駐車を完了させる方法の確立が本研究における一番の課題であると考ええる。この課題における提案するシステムは、外界認識しつつ、物体との距離を測り接触を避けつつリアルタイムに舵角を切るシステムが良いと考える。

また、駐車場のカメラや車、管制サーバの役目を果たす PC との連携、いわゆるコネクティッド面の開発も課題と考えている。

付録

1 記号

本論文で用いる記号を以下の通りに示す。

l : ホイルベース

δ_0 : スタート時の車の向き

δ_1 : 点 O1 の車の方向角

δ_2 : 点 O2 の車の方向角

δ_3 : 点 O3 の車の方向角

δ_4 : 点 O4 の車の方向角

θ : 車両旋回半径

θ_r : 後退時旋回舵角

θ_f : 前進時旋回舵角

R : 車両旋回半径

R_f : 前進時旋回半径

R_r : 後退時旋回半径

ΔX : O1 と O3 における X 座標の差

ΔY : O1 と O3 における Y 座標の差

Fig.51 の通り、座標軸は、図の右上側を原点とし横方向を X 方向、縦方向を Y 方向とする。また、車の方向については、X 軸と平行な方向を 0° 、Y 軸に平行な方向を 90° とし、右回りを+、左回りを-とする。

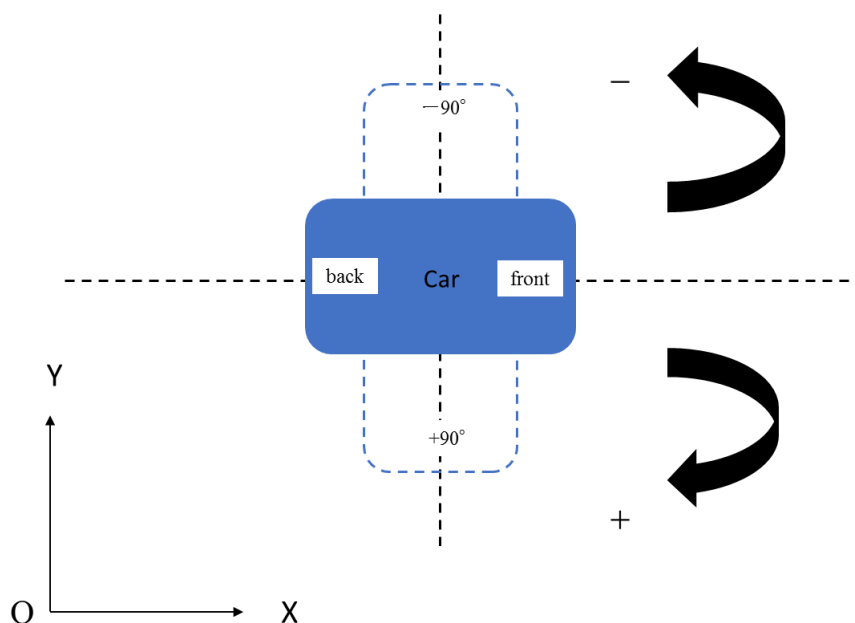


Fig. 51 Image of coordinate axis and direction angle

2 旋回半径導出のための RoboCar の実験的検討

経路を設計する上で、入力値として設定した舵角における旋回半径値を計測するため、RoboCar を実際に走行させ旋回半径を実測する実験的検討を行った。速度は、0.3m/s 一定とし舵角を 5~30° まで順に計測を 2 回行った。計測結果を Table5 に示す。

Table 5 The turning radius measurement result at each steering angle

入力舵角[°]	[rad/s]	実半径1	実半径2	平均
30	0.523599	0.525	0.52	0.5225
29	0.506145	0.545	0.55	0.5475
28	0.488692	0.565	0.57	0.5675
27	0.471239	0.585	0.585	0.585
26	0.453786	0.635	0.633	0.634
25	0.436332	0.657	0.66	0.6585
24	0.418879	0.685	0.695	0.69
23	0.401426	0.712	0.715	0.7135
22	0.383972	0.75	0.75	0.75
21	0.366519	0.785	0.79	0.7875
20	0.349066	0.82	0.83	0.825
19	0.331613	0.86	0.88	0.87
18	0.314159	0.905	0.895	0.9
17	0.296706	0.975	0.975	0.975
16	0.279253	1.02	1.03	1.025
15	0.261799	1.07	1.06	1.065
14	0.244346	1.12	1.11	1.115
13	0.226893	1.16	1.18	1.17
12	0.20944	1.24	1.23	1.235
11	0.191986	1.32	1.3	1.31
10	0.174533	1.43	1.43	1.43
9	0.15708	1.54	1.54	1.54
8	0.139626	1.64	1.65	1.645
7	0.122173	1.88	1.9	1.89
6	0.10472	2.08	2.07	2.075
5	0.087266	2.34	2.35	2.345

計測データを基に舵角と旋回半径の関係性を Fig. 52 に示す。なお、比較のために車両の低速時に用いられる二輪モデルを用いた旋回半径導出式^[2]

$$R = l / \tan \theta$$

の結果も併せてグラフに示した。

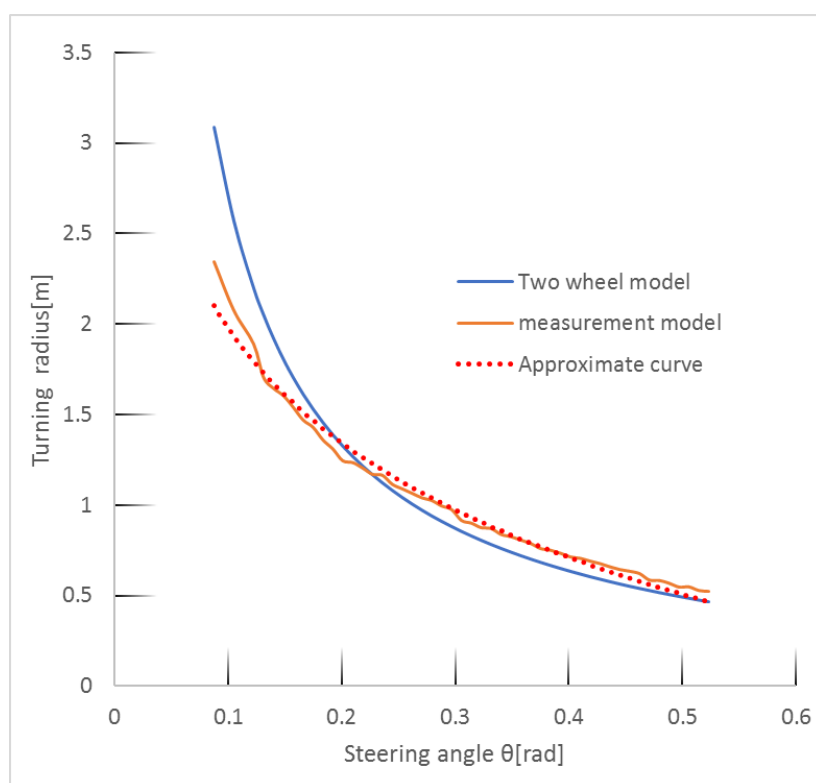


Fig. 52 Relationship between steering angle and turning radius

Fig. 52 を見てみると、二輪モデルでの旋回半径と実測値の旋回半径には明らかに誤差が大きく、舵角 5° の時は、誤差が 0.74m もあった。誤差が大きくなってしまった原因として、RoboCar の舵角が指示した通りに切れていないことが考えられる。そこで、本研究における車両の旋回半径の導出は、舵角－実測半径値の関係性を曲線で近似し、導出した数式を用いることとした。近似曲線の数式を導出すると

$$R = 0.33\theta^{-0.84}$$

と表された。また、近似曲線と舵角－実測半径値の曲線を比較すると誤差が最大で 0.9m となり、二輪モデルより精度が向上するため、上記の式を用いて旋回半径を計算することとした。

3 RoboCar 走行実験のための経路生成実験結果

4・3 で記載していない条件 1 から 3、7 から 10、12、13、15、16、18 から 22、24 から 27 の経路設計結果を順に以下に示す。

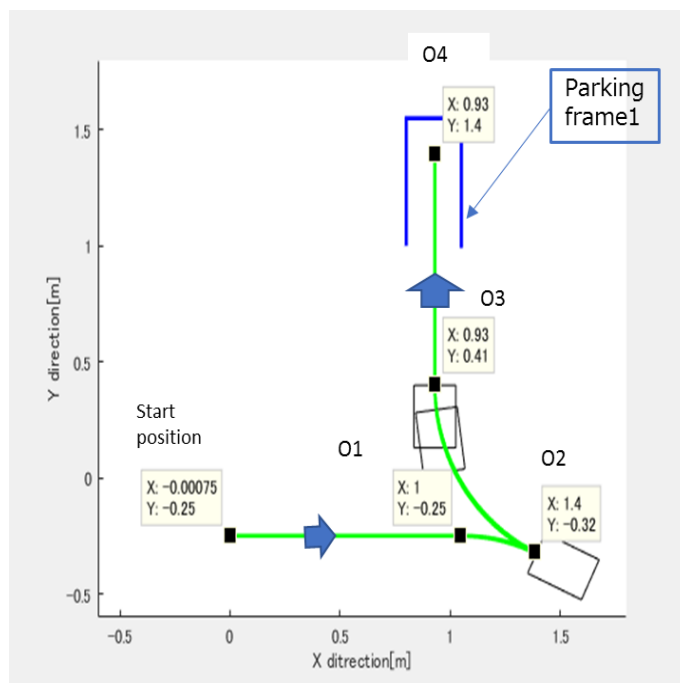


Fig. 53 Parking route result of condition 1

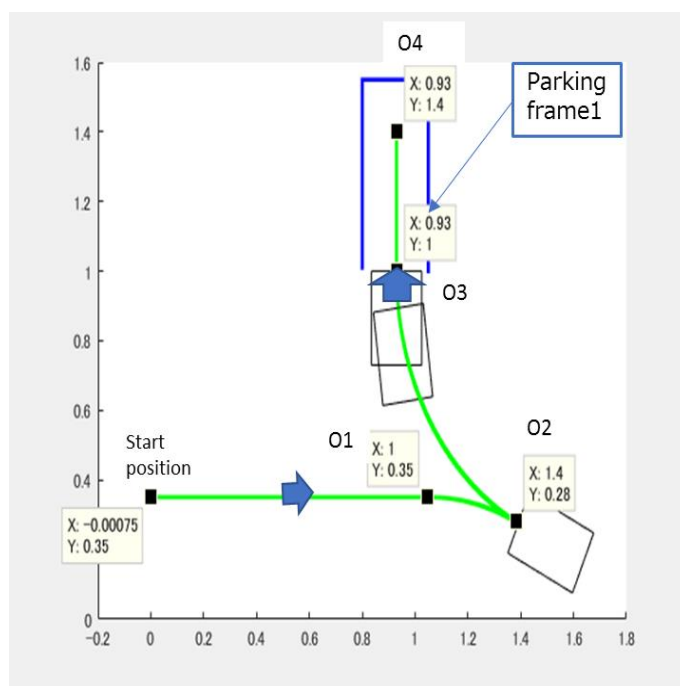


Fig. 54 Parking route result of condition 2

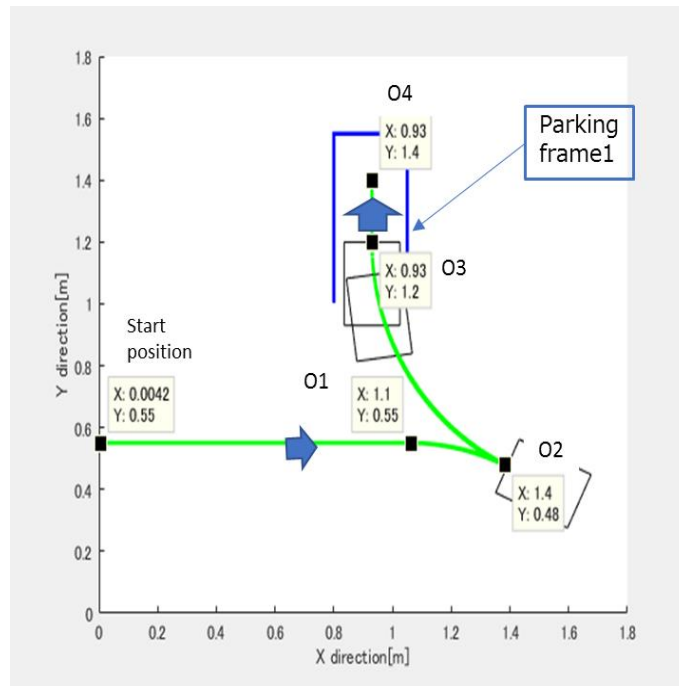


Fig. 55 Parking route result of condition 3

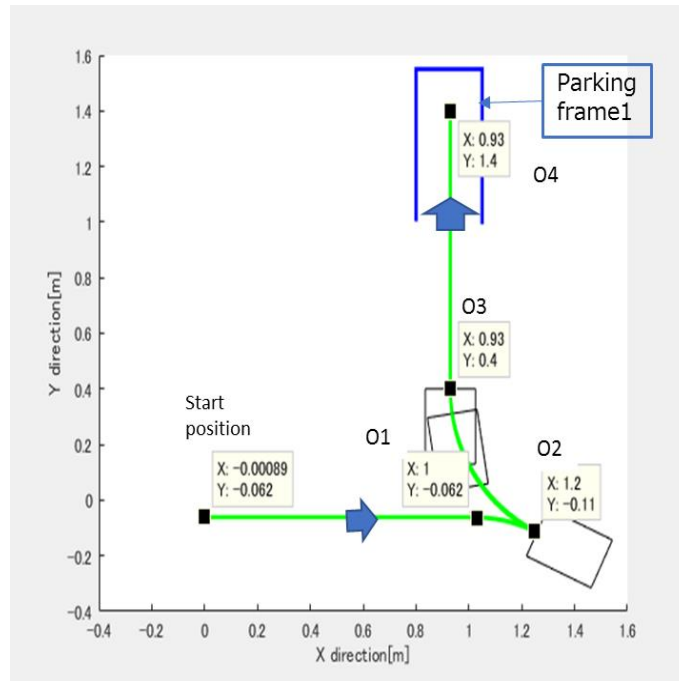


Fig. 56 Parking route result of condition 7

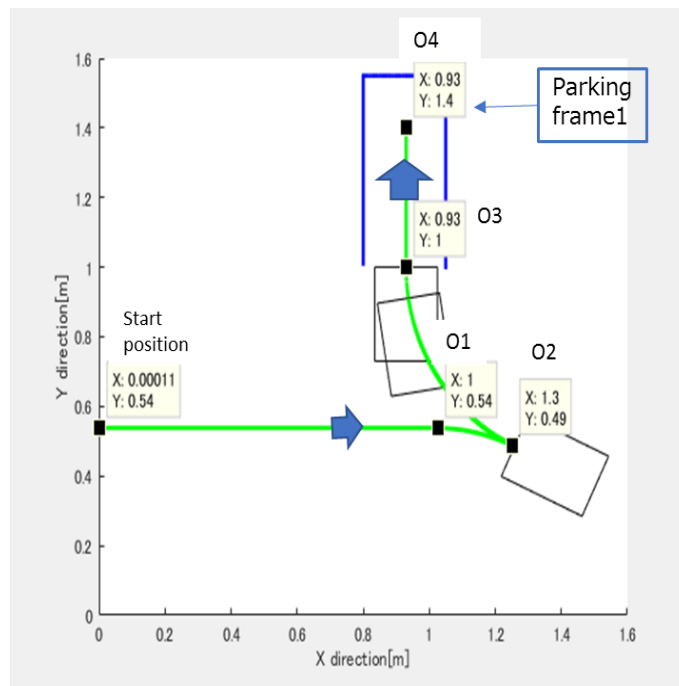


Fig. 57 Parking route result of condition 8

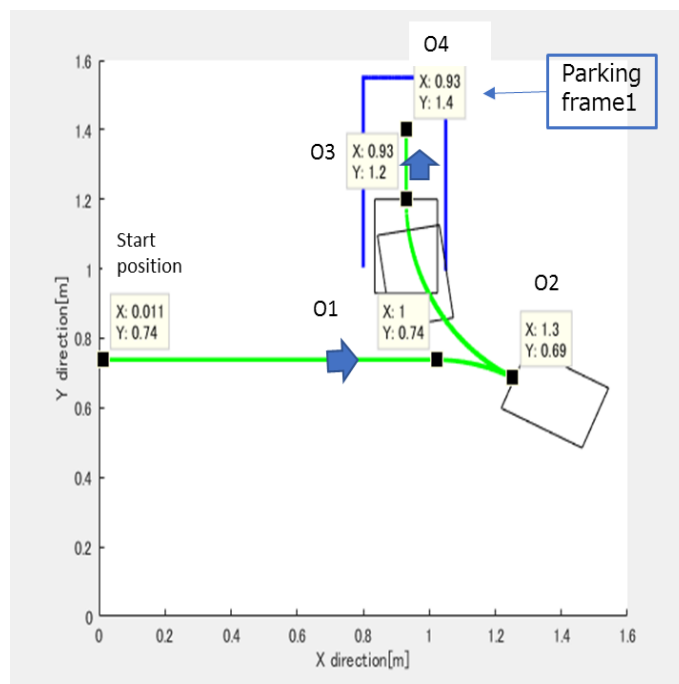


Fig. 58 Parking route result of condition 9

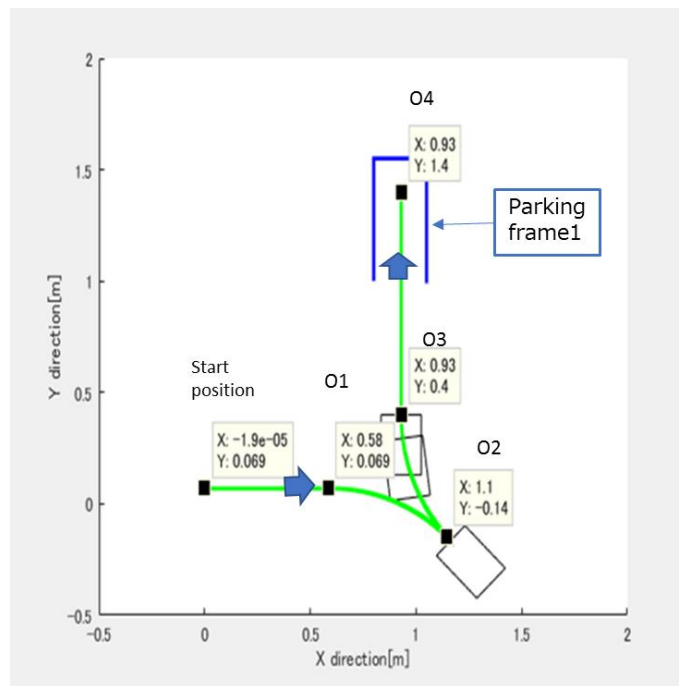


Fig. 59 Parking route result of condition 10

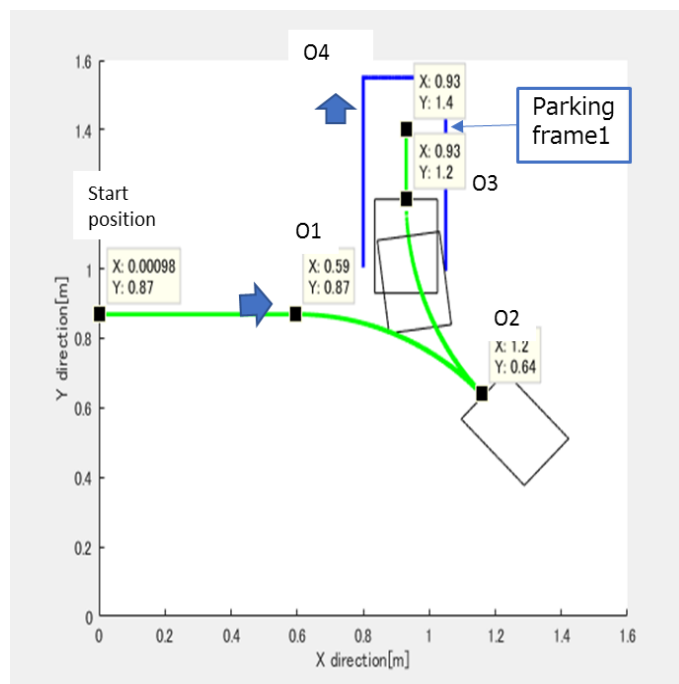


Fig. 60 Parking route result of condition 12

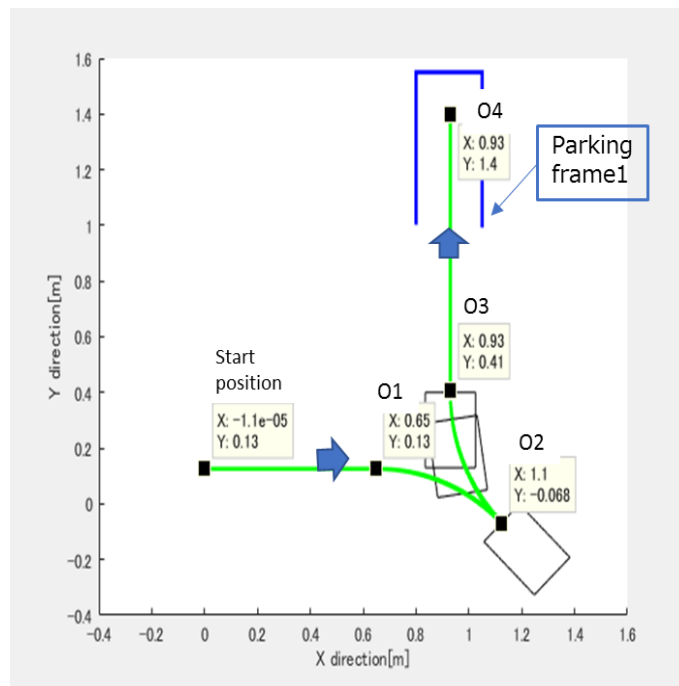


Fig. 61 Parking route result of condition 13

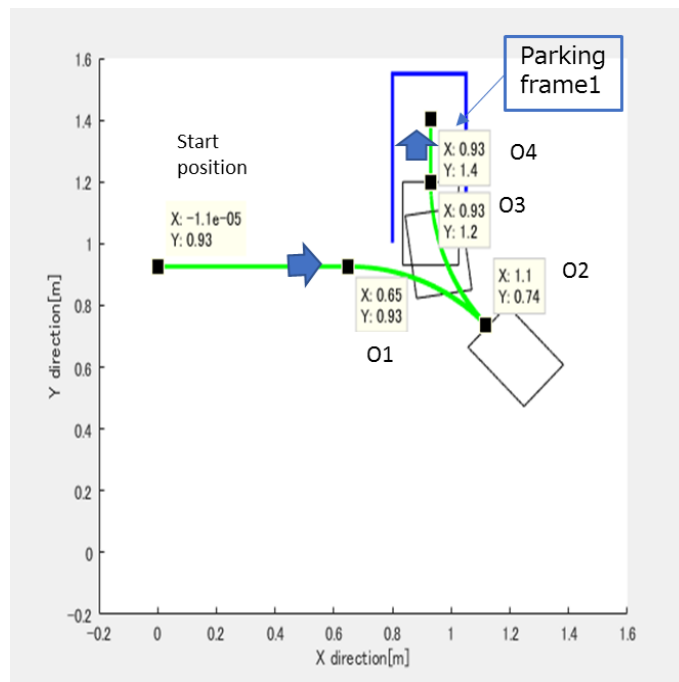


Fig. 62 Parking route result of condition 15

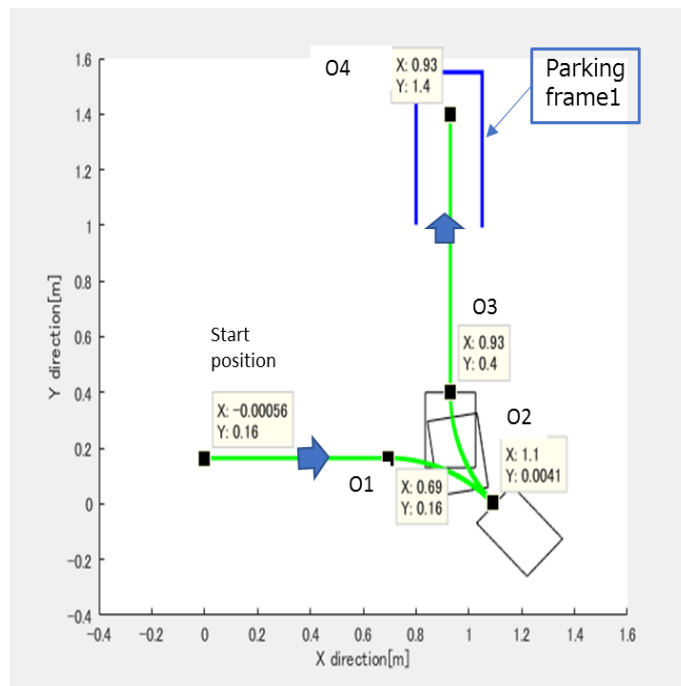


Fig. 63 Parking route result of condition 16

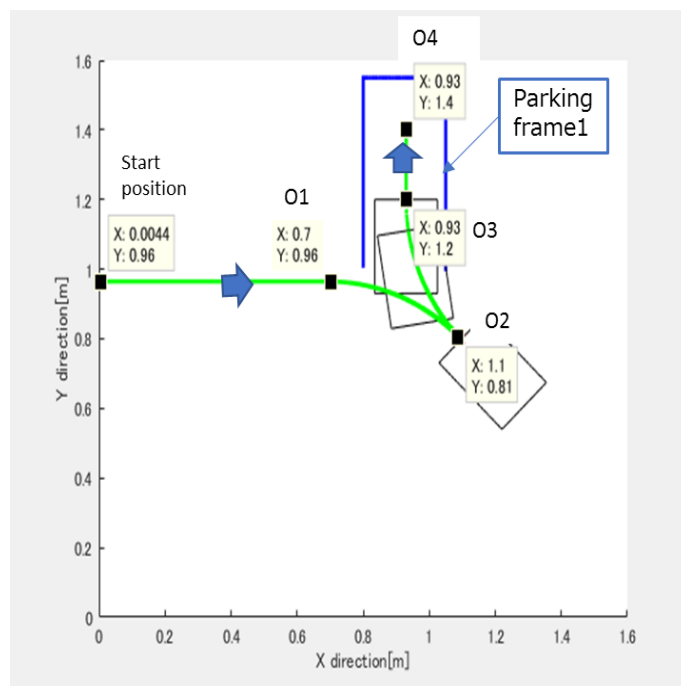


Fig. 64 Parking route result of condition 18

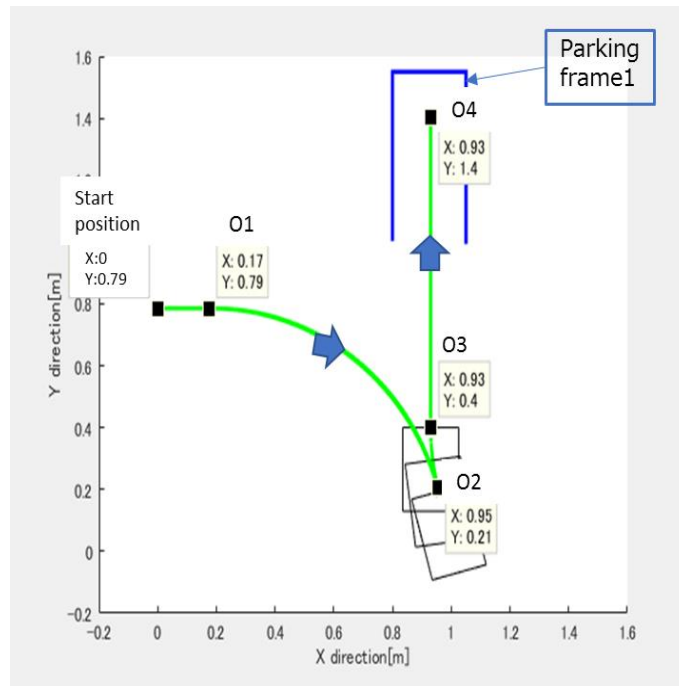


Fig. 65 Parking route result of condition 19

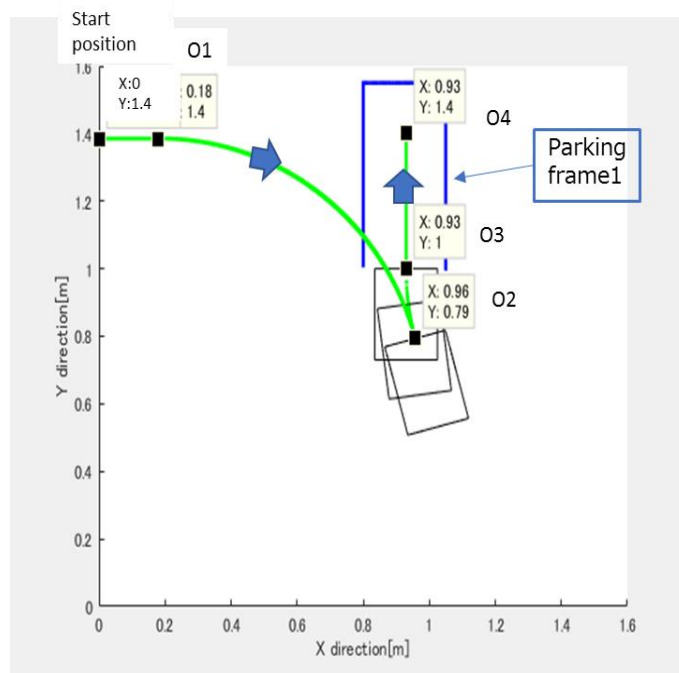


Fig. 66 Parking route result of condition 20

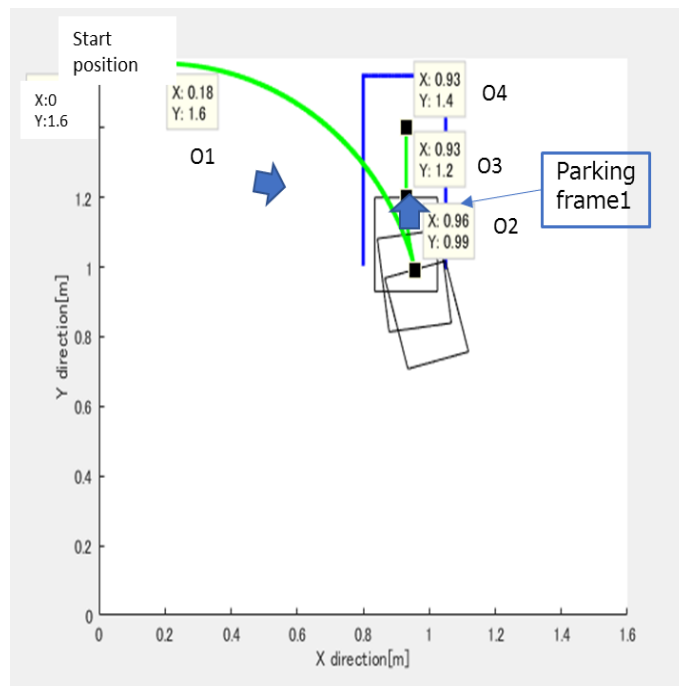


Fig. 67 Parking route result of condition 21

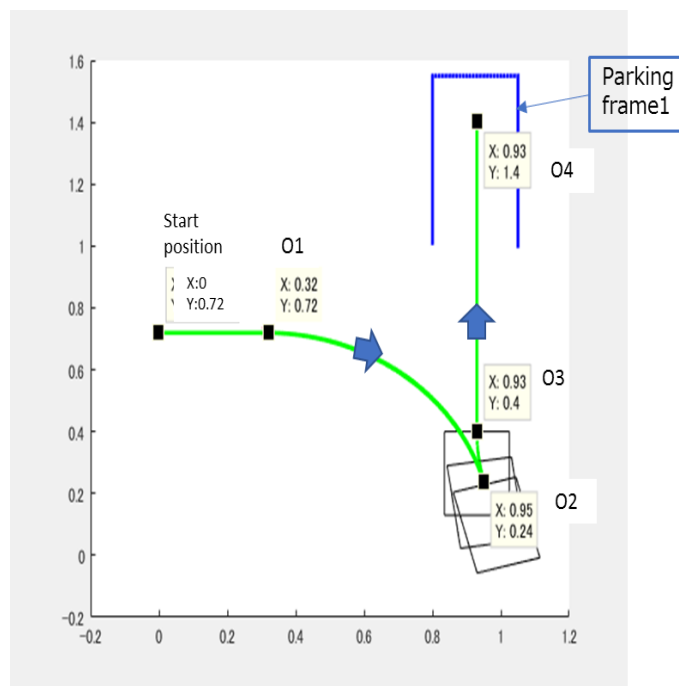


Fig. 68 Parking route result of condition 22

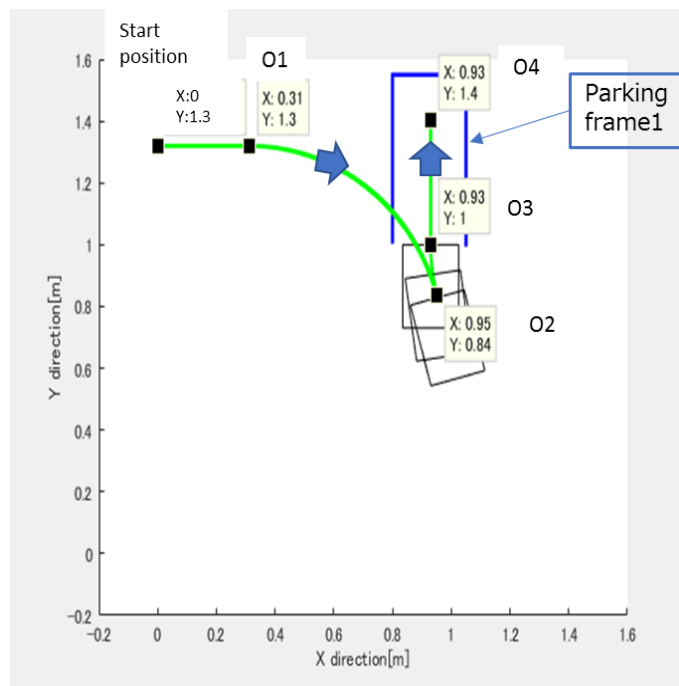


Fig. 69 Parking route result of condition 23

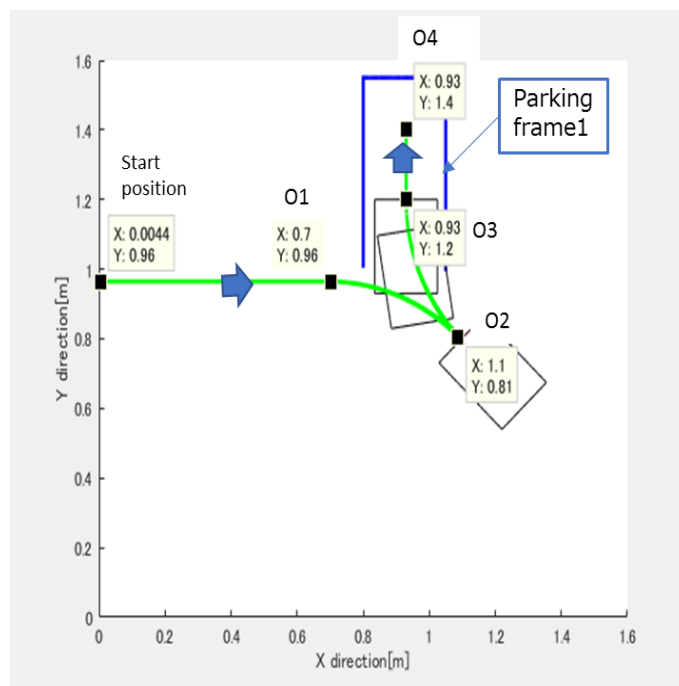


Fig. 70 Parking route result of condition 24

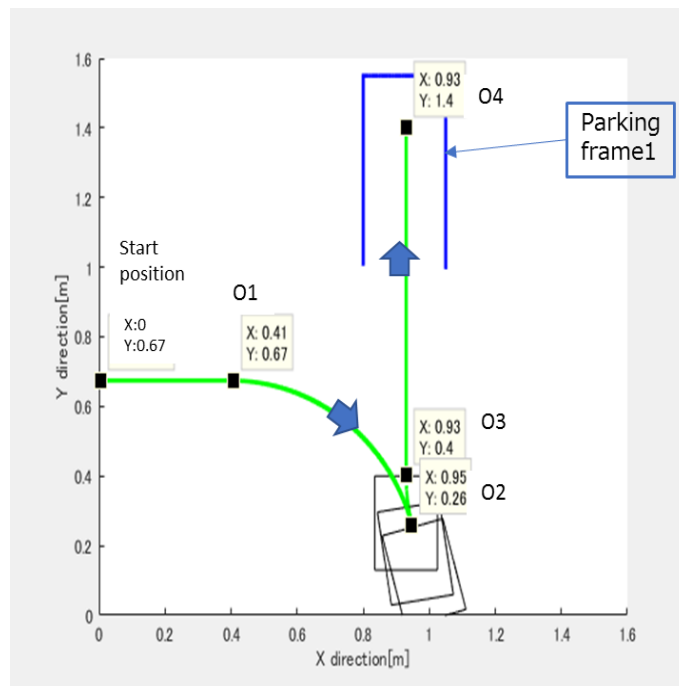


Fig. 71 Parking route result of condition 25

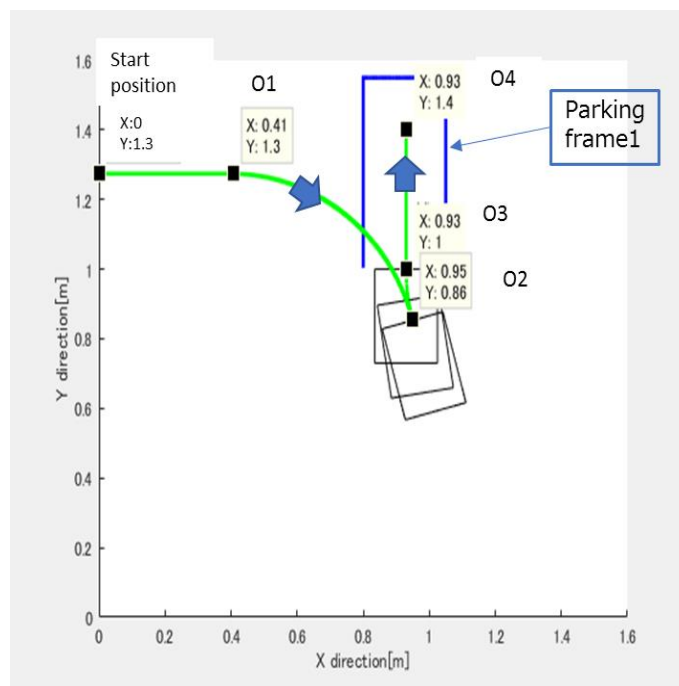


Fig. 72 Parking route result of condition 26

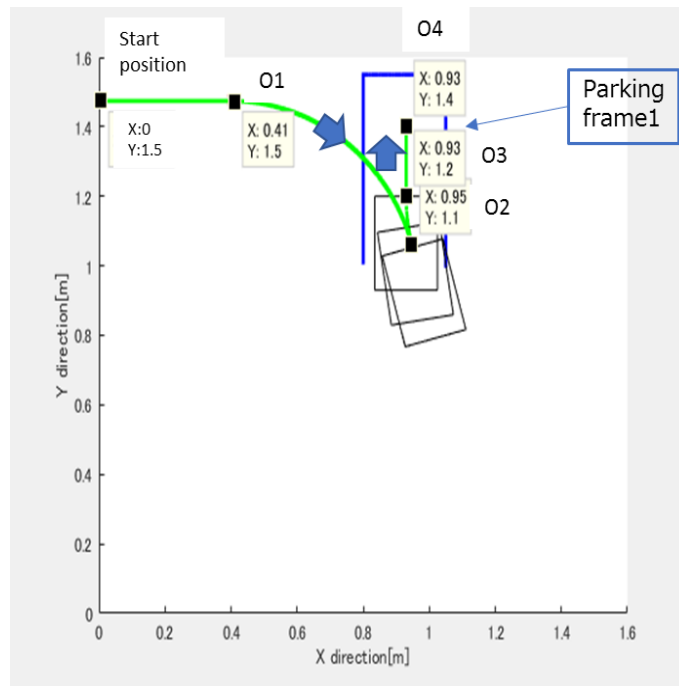


Fig. 73 Parking route result of condition 27

付録 3 RoboCar

実験装置は、Fig.74 に示す実車の 1/10 スケールサイズの RoboCar を用いる。概要を Table5 に示す。

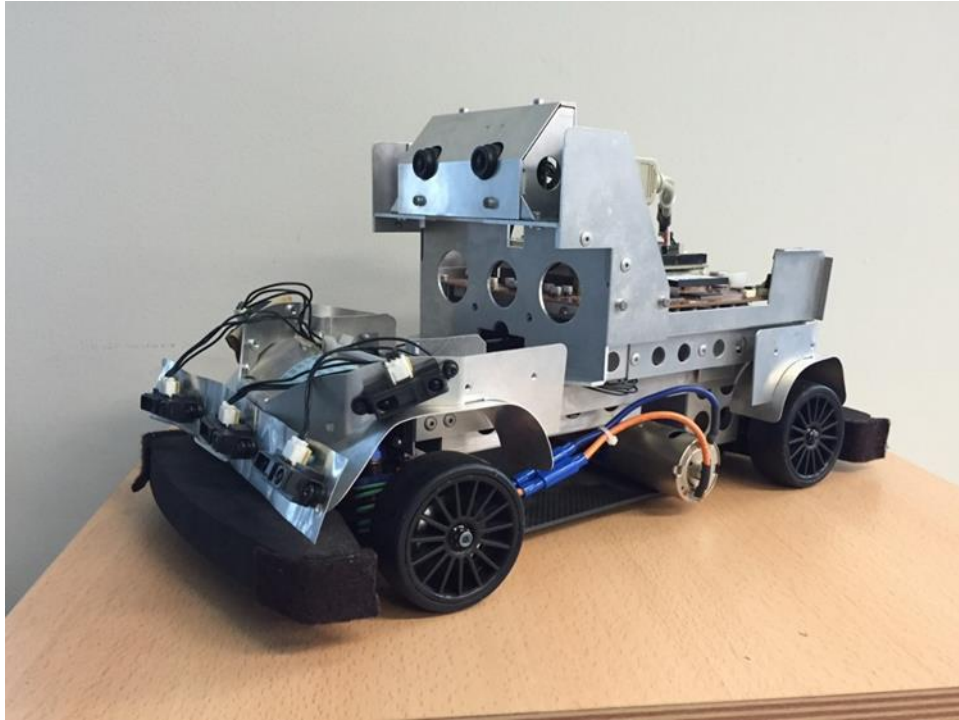


Fig. 74 RoboCar

Table 5 RoboCar summary

RoboCarの主な仕様		
全長×全幅×全高 / 全量		429.0mm×195.0mm×212.2mm / 約3kg
システム 基本構成	内界センサ	ジャイロセンサ1台、加速度センサ3台、ロータリエンコーダ（車軸4台、駆動モータ1台）
	外界センサ	赤外線測距(x8)
	モータドライバ	ZMP製モジュール
	サーボモータ	ロボット用サーボモータ
	駆動用モータ	小型EDCモータ
システム 基本構成	バッテリー	制御システム用バッテリー単三ニッケル水素電池
		駆動用バッテリー ニッケル水素バッテリー
スケール モデル カー本体	メインコントローラOS	Linux(ソフトリアルタイム)
	コントロールソフトウェア	制御ソフトウェア、ZMPライブラリ、ネットワークソフトウェア
	画像処理プロフェッサ	専用コード
PCソフト ウェア	OS	Windows/Linux
	開発環境	gcc

参考文献

- [1] 本田技研工業、デンソー 第20回 ITS 世界会議東京における出展概要・2016年 2/10
<http://monoist.atmarkit.co.jp/mn/articles/1310/08/news081.html>
- [2] 自動運転 LAB・自動バレーパーキングとは？ 自動運転技術を活用 開発企業は？・2018年 2/11
https://jidounten-lab.com/y_automated-valet-parking-autonomous
- [3] レスポンス・未来の駐車システム「自動バレーパーキング」実証実験、JARI が東京・台場で開催・2018年 2/10
<https://response.jp/article/2018/11/14/316137.html>
- [4] MOBY・インテリジェントパーキングアシストとは？トヨタや日産の搭載車種を紹介！・2018年 2/10
<https://car-moby.jp/96167>
- [5] レスポンス・【日産 リーフ 新型】ボタン操作ひとつで自動駐車できるプロパイロットパーキング初採用・2018年 2/10
<https://response.jp/article/2017/09/06/299412.html>
- [6] 株式会社本田技研・Honda スマートパーキングアシストシステム・2018年 2/10
<https://www.honda.co.jp/tech/auto/hondasmart-parking/>
- [7] 清水康夫・「先端自動車工学」・P297,298・東京電機大学出版局

[8] 安倍正人・「自動車の運動と制御 第2版」・P66,67・電機大出版局

謝辞

本研究を行うのにあたり、岡宏一教授、原田明德講師にご指導いただき感謝いたします。
また、RoboCar の操作等，生活面でも支えていただいた山野大輝君にも感謝いたします。本研究においてご指導いただいた研究室の皆さま、生活面で支援いただいたご両親に心から感謝いたします。