

東京大学大学院工学系研究科システム創成
学専攻

修士論文

広域での交通安全施策評価が可能な交通流
シミュレータの開発
指導教員吉村忍教授

久保恒太 *

2015 年 1 月 14 日

* 東京大学大学院工学系研究科システム創成学専攻

目次

1	序論	4
1.1	研究の背景	4
1.2	交通事故シミュレーションの現状	5
1.3	研究の目的	6
2	研究手法	7
2.1	研究手法の特徴	7
2.2	交通流シミュレータ MATES	9
2.3	MATES における認知・判断・操作	9
2.4	交通事故の分類	11
2.5	エラーの分類	13
3	交通事故のモデル化と実装	14
3.1	追突事故	14
3.2	出会い頭事故	17
3.3	右左折事故	23
3.4	進路変更事故	25
3.5	正面衝突事故	27
3.6	衝突判定	29
4	挙動検証と感度解析	37
5	実都市への応用	38
6	結論	38
7	はじめに	38
8	空白と強制改行	38

9	箇条書きと書体	39
10	数式	39
11	表と図	40
11.1	表	40
11.2	図	40
12	その他小ネタ	41

1 序論

1.1 研究の背景

国土交通省では平成 27 年までに 24 時間死者数を 3,000 人以下に、死傷者数を 70 万人以下とすることで、世界一安全な道路交通を実現するという目標を掲げている。また、死傷事故の約 70 %が道路全体の 20 %の区間に集中しているという事実から、幹線道路の中でも特に危険な箇所を特定し、その箇所に対して重点的に対策を仕掛けている。道路改良による対策としては、右折レーンの延伸、歩道の整備、最高速度制限などが挙げられる。また、交通安全施設の設置としては、防護柵、導流表示、注意喚起表示（LED 表示）などの設置が挙げられる。

これらの道路や施設設置による従来までの対策に対して、近年の傾向としては交通事故対策として、高度交通システム (ITS)[1] と呼ばれる、IT 技術を駆使し、人と道路と自動車の中で情報の受発信を行い、道路交通が抱える事故や渋滞、環境対策など、様々な課題を解決するためのシステムを利用したものも登場している。交通安全支援の ITS である ITS スポットサービス [2] では、道路に設置にされた ITS スポットから車載されているカーナビに対して情報を送る。これにより、カーブなどの見通しが悪い場所でも近くに車がいることを ITS スポットが知らせることで、事前に事故を防ぐことができる。他にも、車両前方のレーダーが先行車を察知し、追突の危険が高まった場合には自動的にブレーキをかけることで追突事故を未然に防ぐ、自動車被害軽減ブレーキ [3] などがある。

このような国土交通省をはじめとした、行政の重点的な交通事故対策により、日本の事故死亡者は 1992 年から、また、事故発生件数は 2004 年から連続して減少傾向にある。図 1 に近年の交通事故による死者数および負傷者数を示す。しかし、依然として平成 25 年中の年間交通事故死者数は 4,373 人、負傷者数は約 78 万人 [4] である。この数字が示すように交通事故は依然として大きな社会問題であり、交通事故をなくすことは国家の利益につながる。

このような状況の中で交通事故に対して効果のある安全支援対策が待ち望ま

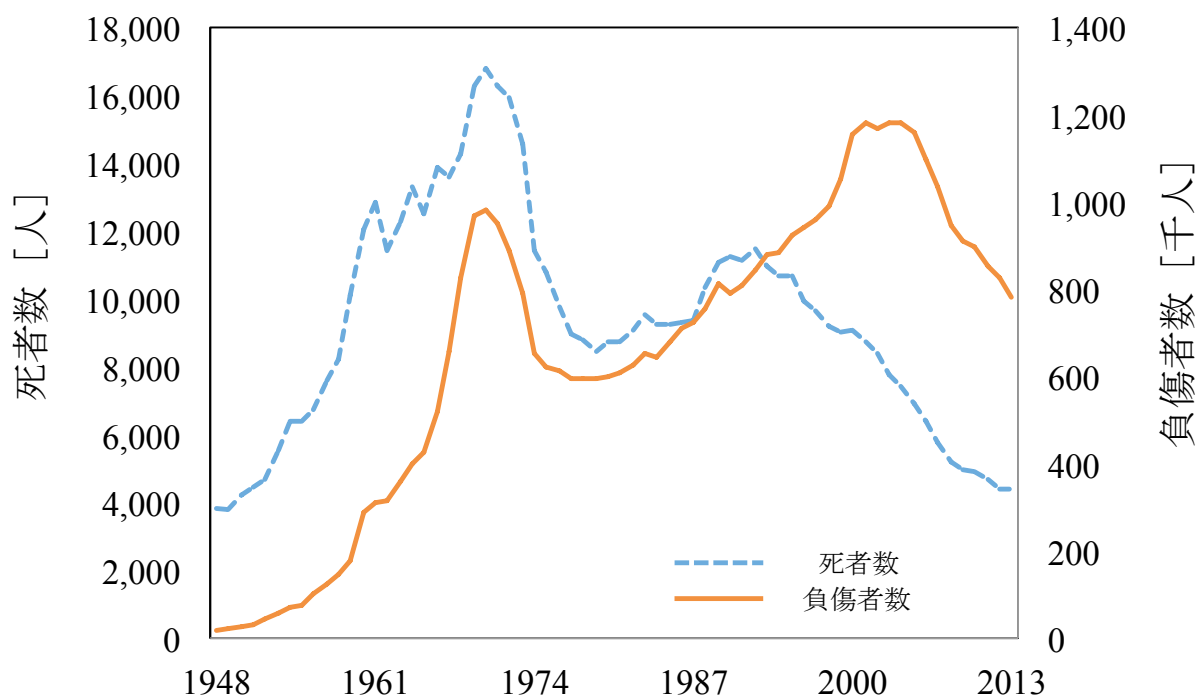


図1 交通事故死者数、負傷者数の推移

れるが、各対策を評価し、効率的に実施していくことが重要である。交通安全支援施策を評価しようとする時に、問題となるのが実験の困難さである。交通現象の実験では道路やその周辺施設を実験のたびに工事しては大きな費用がかかる。そのため、交通流シミュレーションによって、実際交通事故を再現できると、道路や交通規制の設計、ITS 推進や交通教育への利用などによる事故状況の変化を仮想的に再現することでそれらの施策を評価することができる。こうした点で交通事故を再現する交通流シミュレータによる安全支援施策評価は事故の減少に寄与すると考えられる。

1.2 交通事故シミュレーションの現状

マルチエージェントを用いた交通流シミュレーションに関する研究は現在まで、多くされてきた。それらの研究は交通渋滞に関するものなどが多く、交通事故に関するものは少ない。その理由としては、そもそも交通事故をモデル化することが難しいことなどが挙げられる。こういった中で行われてき

た既存研究では交様々なモデルを提案し交通事故の再現してきた。田中らの研究では認知エラー・判断エラー自動車の運転行動は認知・判断・操作の3つの過程に分類される。この3つの過程のどこかにエラー状態が起きた時に交通事故は発生する(2)。既存の交通流シミュレータにおける交通事故の実装においても、この考え方に沿ったものが多い。認知段階におけるエラーでは、認知フィルタという概念により、ドライバ間での認知レベルの差を表した田中ら(3)のモデルや、藤井ら(4)の中心視野と周辺視野を考慮したモデルなどがある。また、古川ら(5)の研究はエージェントの判断によるエラーに踏み込んだものである。これらの研究では交通事故を実現象に近い形で再現してはいるものの、例えば、特定の交差点など局所での安全施策評価での利用に限られており、広域なシミュレーションには利用できず、実際に街路を設計する際の交通安全施策評価の実用性の問題が残る。このような現状から広域で利用できる交通流シミュレータが必要とされている。

1.3 研究の目的

本研究の目的は都市単位での様々な広域での交通安全施策評価が可能な交通流シミュレータの開発である。交通情報である道路形状、交通量、信号の配置、点灯、建物の情報を入力することで、事故情報である事故発生件数、事故形態、事故発生箇所などを出力するマルチエージェントシミュレーションである。具体的な利用方法としては、都市などで道路構造を変えたり、交通量が変化した時にどの様に事故が増えるか減るかなどを行政や地方自治体が試算するために利用するなどが考えられる。

2 研究手法

2.1 研究手法の特徴

以下に本研究手法における特徴を述べる。

2.1.1 複数のモデルに対する複数の事故をモデル化・実装

本研究においては、5つの事故である追突事故・出会い頭事故・右左折事故・進路変更・正面衝突事故を扱う。これらの事故が起こる過程においては、それぞれ別々のエラーが起こり、事故に繋がるというモデルになっている。交通事故をモデル化する際に汎用的な単一のモデルを用い様々な事故を再現するという手法もあるが、本研究では実際に起こりやすい事故について、事故が発生するまでのプロセスはそれぞれに違った現象であるという立場からこの手法を採用した。

2.1.2 認知・判断エラーをモデルに加味

認知・判断エラーについて説明する前に、自動車の運転行動について説明する。運転行動は一般的に認知・判断・操作の3つの過程に分類される[11]。認知過程では、安全に通行するために必要な物を見る（発見する）ことであるが、これは単に見る判断過程では、認知した対象がどのような行動をするのか、自分はどのように行動すればよいのか操作過程では、判断や決定に従ってハンドルやブレーキなどを操作する。例えば赤信号における停止という運転行動では、赤信号を見て（認知）、減速しようと考え（判断）、ブレーキを踏む（操作）といった流れとなる。

過去の研究においては認知エラーが採用されているものが多い、認知エラーによる事故は件数も多く、見えているかどうかということが物理的に比較的明確な現象なのでモデル化しやすいからである。これに対して判断エラーを考慮することは人間の思考の一部をモデル化することとなり、様々な考慮をしなければならない。古川らは交通事故シミュレーションを行う上でこの判断エラーを取り入れた[6]。この研究ではUDM(Universal Design

Model) というモデルを提案している。UDM における認知過程ではエージェントがそれぞれ現実の世界とは違う外部世界というものを保持している。エージェントは認知段階において外部世界に情報を蓄える、そして判断段階では現実の外部環境ではなく、外部世界から情報を参照する。また、ドライバは個性や心理状態に応じて運転欲求を持っており、の欲求に応じて判断を変えてリスク判断を行っていく。このリスク判断の部分については確率的な計算を行っている。外部世界モデルの概念はエージェントによって見えているものがそもそも違い、判断についてもその時々によって異なるということを表している点で現実の運転行動をよく表現していると考えられる。そのため、本研究ではモデルを構築する上で外部世界の概念を取り入れ、判断・認知エラーを加味することとする。操作段階におけるエラーについて事故件数が少ないとされているため、本研究では扱わない。

2.1.3 計算時間を考慮した簡易なモデル

本研究の目的とする広域での交通流シミュレーションを実現するためには計算時間の短縮が不可欠であるが、認知・判断過程でエージェントにエラーを起こすということは複雑な計算を行う場合が多く、計算量が多くなりがちである。藤井らの研究 [7] は認知エラーに絞って単一のモデルから構築されており、ドライバに見えている景色を二次元の画像として、各ドライバのエージェントが持つことで、他のエージェントを認知できなくなった時に衝突の可能性がある。認知できるかどうかの判断については、中心視野、周辺視野を考慮している。この研究は人間の視覚を精緻に再現し、現実と近いものになってはいるものの、二次元の図を各タイムステップで描く必要があり、計算量が膨大となるため、広域での交通流シミュレーションには向かない。本研究では目的に合わせ、現実の事故を再現しながらも、簡易なモデルを構築するという手法で広域での交通流シミュレーションを実現する。

2.2 交通流シミュレータ MATES

本研究では、著者らが開発中の知的マルチエージェント交通流シミュレータ MATES (Multi-Agent-base Traffic and Environment Simulator) に新たな認知・判断モデルを追加・実装する。MATES においてドライバはエージェントとしてモデル化されており、周囲の環境から情報を取得して自律的に行動する。複雑な交通流を再現可能であり、現実の現象に即した交通事故を再現する。また、近年の傾向として、大規模なシミュレーションでも対応可能であるように並列化も行われおり、計算不可の高い交通事故をシミュレーションに適している。これらの理由から MATES は本研究の目的に適した交通流シミュレータであるといえる。

2.3 MATES における認知・判断・操作

ここで MATES における認知・判断・操作について説明する。MATES においては、0.1 秒ごとに各エージェントが運転行動を行う。各タイムステップでの認知・判断・操作の全体のフロチャートを図 2 に示す。

認知

認知段階では認知対象物である先行車、交差点交錯車両、信号、進路変更次の並走車などを仮想先行車と呼び、その位置や状態を取得し、エージェントに格納していく。どの様にエージェントが認知を行っているかを先行車の認知を例にして説明する。エージェントが所属する車線オブジェクトがその車線上にあるエージェントの情報を持っており、かつ、車線の始まりから終わりにかけて、車両の配置されている順番とともに情報を持っている。そのため、エージェントが先行車を取得するには車線オブジェクトに問い合わせるということになっている。この様に MATES では様々なオブジェクトが互いに情報を持ち合い参照し合うことで、認知がなせるようになっている。この様にして、認知したオブジェクトを仮想先行車のベクトルに格納していく。

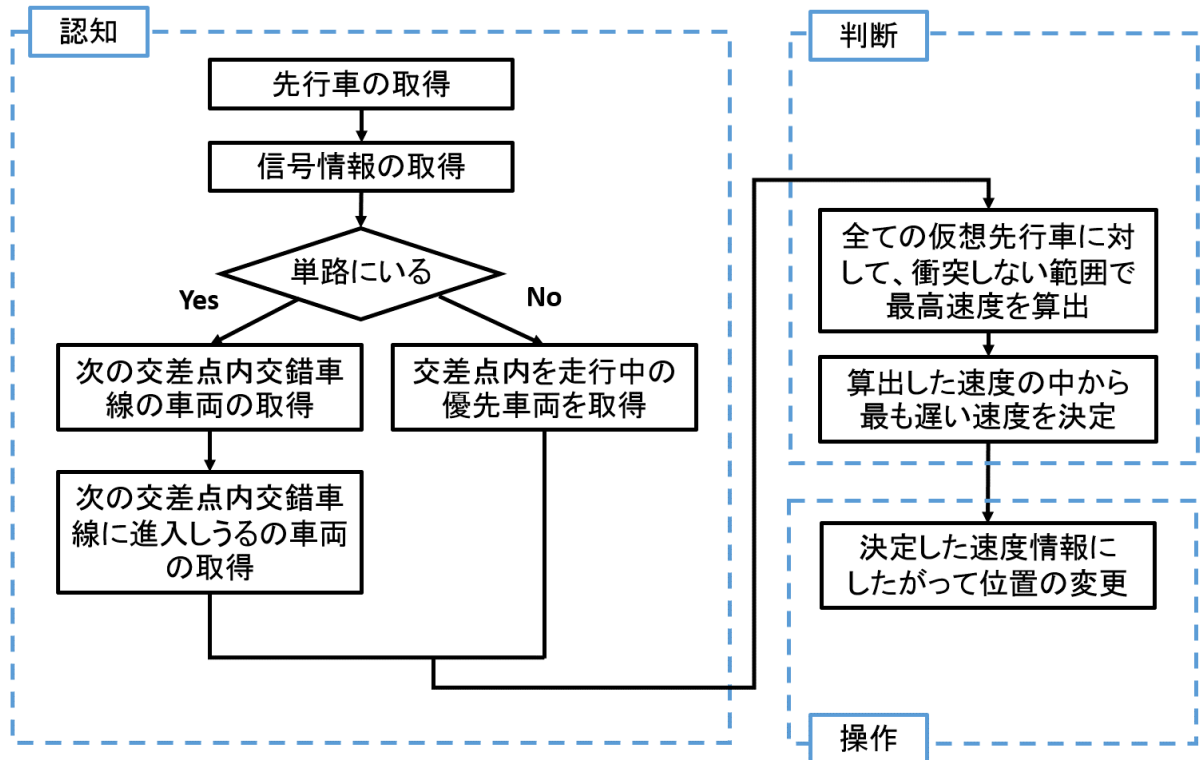


図2 認知・判断・操作のフロチャート

判断

判断段階では、次のタイムステップでの速度の決定をしている。認知段階で取得した各仮想先行車に対して、衝突しない中でそれぞれのエージェントが持つ希望速度に最も近い速度を算出する。速度の算出には追従モデルを用いている。ここで算出された速度のうち最小のものを速度とする。

操作

操作段階では判断段階で算出した速度に対して、実際のエージェントの位置情報を書き換えるこれを繰り返すことによって、MATES 内での運転行動は行われている。

2.4 交通事故の分類

交通事故は分類して考えられる。図 3 は平成 15 年から平成 25 年までに起きた交通事故を分類分けをした円グラフである。このグラフは警察庁のデータ [4] を元に作成した。

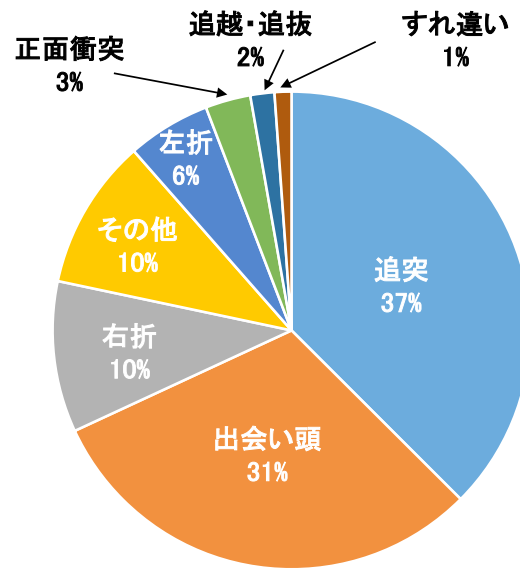


図 3 平成 15 25 年における交通事故の分類

これによるとこの年に最も多かった事故は追突、ついで出会い頭事故、右折と続いている。その他が 10 % と少なく、事故の大部分が大まかに分類されているのがわかる。この様に現実の事故が分類できることから、本研究においても、これらの事故から代表的なものを取り出し、それぞれ個別にモデル化することで、より実現象に近い形で交通事故が発生することを目指す。これらの事故のモデル化に際しては自動車技術会が発表している交通事故予測シミュレーションシステム検証マニュアル [8] を参考にした。このマニュアルは予防安全策の開発において、その効果を予測・故予測シミュレーションシステムが備えている機能レベルや交通状況の再現能力を評価するための標準的な検証・評価手続きを示したものである。以下に 5 つの事故の状況を述べる。

(1) 追突

交差点手前で停止した先行車に追突。先行車は早い段階で停止したが、自車は先行車以外の対象物を認知していたもしくは、脇見しており、先行車を認知できないため追突する。

(2) 出会い頭

交差点を直進で進入しようとした時、側方から他車が出てきたため、出会い頭の衝突。建物が密集していたため、他車の発見が遅れたと考えられる。

(3) 右左折

交差点内で自車が右折しようとしているときに、対向車は左折もしくは直進をしようとし、自車の方が先に通過できると見誤ったために衝突

(4) 進路変更

進路変更時、隣のレーン後方から車両が接近していることに気づかず進路変更を行ったため衝突

(5) 正面衝突

脇見運転をしており、対向車線に進入したため衝突

2.5 エラーの分類

交通事故の要因であるエラーの分析を行った。エラーの分類には自動車技術会の公開しているヒヤリハットデータや、山田らの交通事故シナリオの研究 [9] を参考にした。その結果、事故要因を以下の 3 つに絞った。見えないことによる認知エラー、見ていないことによる認知エラー、傲慢な判断エラーである。これら 4 つのエラーについて説明する。

2.5.1 見えないことによる認知エラー

衝突しうる、また、その原因となりうる対象物（先行車、信号、並走車など、以降認知対象物とする）が物理的に見えない状態である。例えば見通しの悪い交差点などで側方から進入した車両と衝突する場合などはこの原因が強いと考えられる。

2.5.2 見ないことによる認知エラー

物理的に認知対象物が見える状態であっても、ドライバーが見ていないという状態と定義する。例えば脇見運転が該当する。

2.5.3 傲慢な判断エラー

認知対象物を認知している場合において、判断の誤りによって起こるエラーである。右折対左折の事故において、対向車を認知しつつも、対向車が道を譲ってくれると判断した時などはこのエラーを原因とする。また、見通しの悪い交差点で、左右確認をせずに交差点に進入する場合もこのエラーが原因の一つとなる。これら 4 つのエラーを各事故に関して、実装した。

3 交通事故のモデル化と実装

3.1 追突事故

3.1.1 概要

追突事故は先行車の後方と、後方車の前方が追突することや、低速での発生が多いことから幸い死亡事故の割合は他の事故と比べて少ないものの、全体の事故の中でも 37% を締め絶対数としては最も多い事故である。2003 年に交通事故研究センターで行われた追突事故の調査 [10] によると追突事故の原因は図 4 の様に分けることができる。このグラフを見ると、追突事故の多くは見ないことに起因しているといえる。また、この先行車を見なかったという原因を図 5 に示す。この結果から、本研究における追突事故のモデルでは脇見をしたことにより、見ない認知エラーが生じ、事故が発生するものとする。

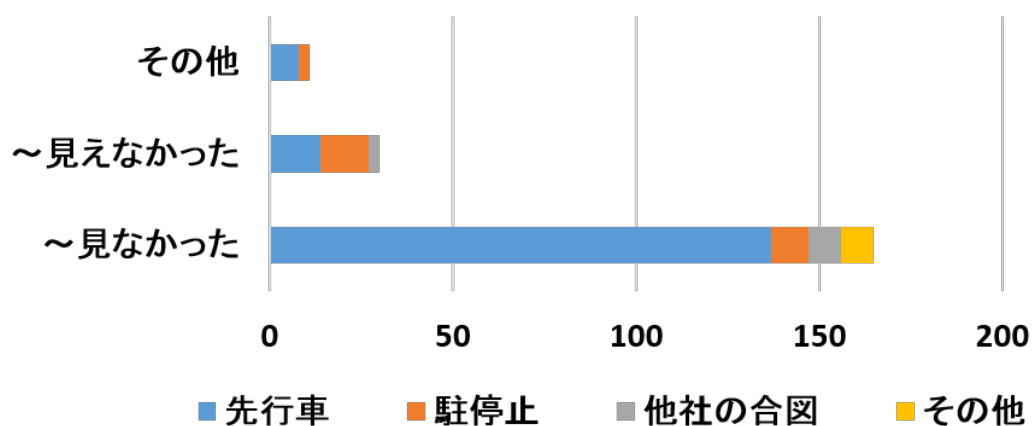


図 4 追突事故の要因

3.1.2 脇見運転モデルによる見ない認知エラーの発生

脇見をしたことにより、認知対象物を見ることができなくなる確率が高くなるということから、各タイムステップにおいて先行車以外の認知対象物が

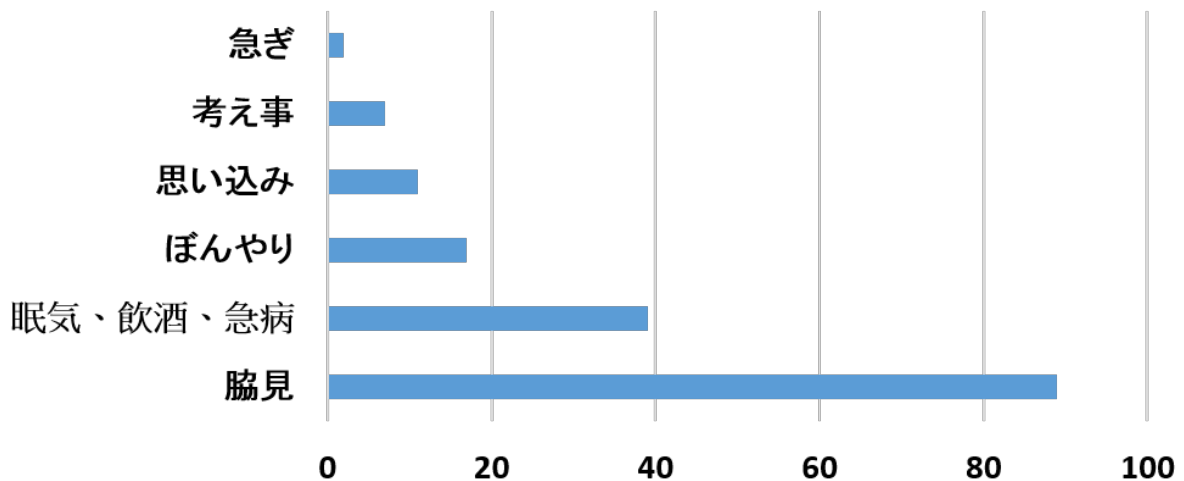


図5 先行車を見なかった要因

多いほど認知エラーが発生する確率が高くなるよう設定した。ここでの認知対象物とは先行車、信号、並走車（進路変更時）、合流車である。見ない認知エラーが発生する割合を追突事故におけるエラー率とする。

3.1.3 先行車位置の予測と事故の発生

見ない認知エラーが発生した時、自転車は先行車を認知するのではなく予想して、本来とは違う速度、距離を保持する。認知エラーが起きた後には、最後に先行車を認知した時に得られた先行車の速度情報をもとにその位置を外挿予測する。これは現実の運転行動においては、先行車を見ていないからといって完全に意識していない訳ではなく、記憶に残しつつ脇見をしていると考えるからである。この様にして計算された先行車と自転車の位置に乖離がある場合に先行車の正確な位置を把握できず、追突事故が発生する可能性がある。

追突事故が発生するまでのフロチャートを図6に示す。

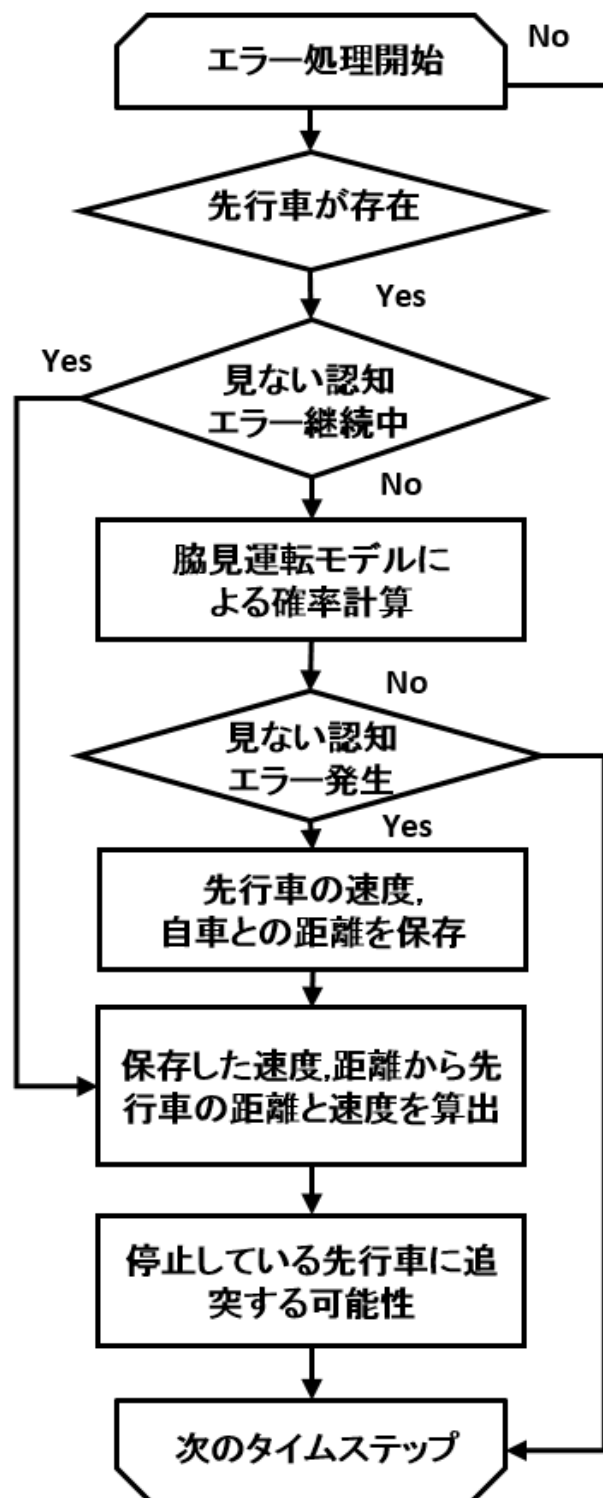


図 6 追突事故のフロチャート

3.2 出会い頭事故

3.2.1 概要

追突事故に次いで発生件数が多いのが出会い頭事故である。交通事故分析センターによる出会い頭事故の要因の分析結果 [11] によると、出会い頭事故の要因には認知エラーと判断エラーの 2 つがある。認知エラーのうち 80% 以上が交差車両を何らかの理由で認知できなかったという、信号のない見通しの悪い交差点などでおきるものである。そのため、本モデルでは信号のない交差点で建物等の遮蔽物があることで、遮蔽物に隠れた車両を認知できなくなるという、見えないことによる認知エラーを再現する。また、視野の制限が起きた時に、見通しの悪い交差点で一時停止するという機能を付け加えた。これに対して、判断エラーで多いのが、見通しが悪い状況でも、自車が優先していけると思った、特に見通しの悪さを気にしなかったなどの理由が挙げられる。これらの点から、傲慢な判断エラーが起きている時には、見通しの悪い交差点で（認知エラーが起きている時に）一時停止としないこととした。

3.2.2 遮蔽物の配置

シミュレーション開始時に全ての交差点について、遮蔽物を配置するかどうか決める。一つの交差点における遮蔽物の数はその交差点とつながっている単路の数から、見通しの計算に寄与しない遮蔽物の数を引いたものである。例えば四叉路の場合図??の様に遮蔽物が配置される。配置には交差点で隣合う単路、例えば単路 1,2 の間に遮蔽物を一つずつ配置する。図の様な場合遮蔽物は 4 つ配置されることとなる。また、この様にして遮蔽物が間にある単路に「見通しの悪い単路」と定義する。また、ここで言う見通し計算に寄与しない遮蔽物とは、交差点とそれにつながる 2 つの単路とのなす角度が 180 度以上になっているような場合である。例えば図 8 の様に単路 1 や 3 から交差点に進入する車両にとって遮蔽物 3 は見通しに寄与しない。この様にして、その 2 つの単路がなす角度が 180 度以上の場合、遮蔽物の配置は行わないものとした。

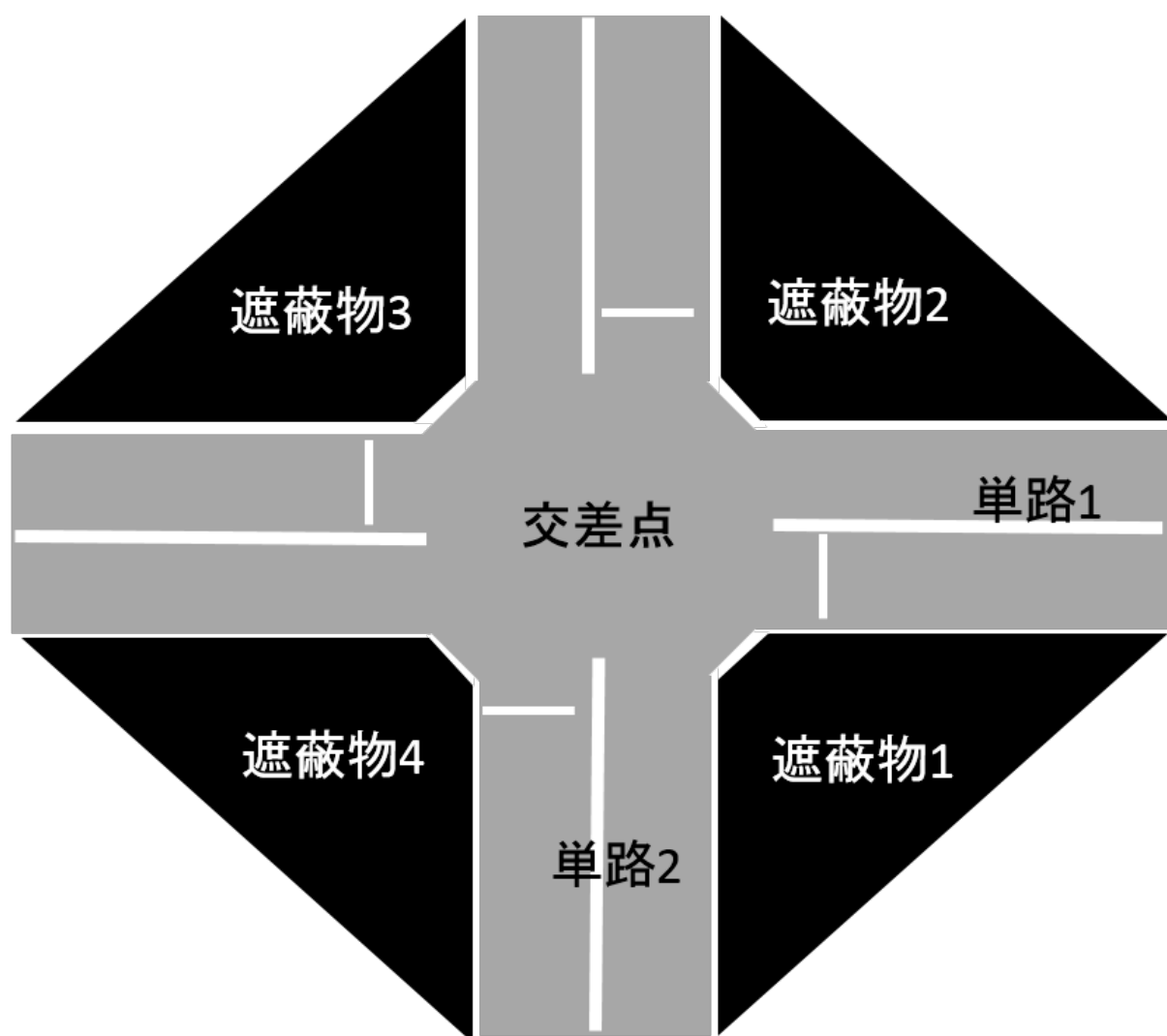


図 7 四叉路における遮蔽物

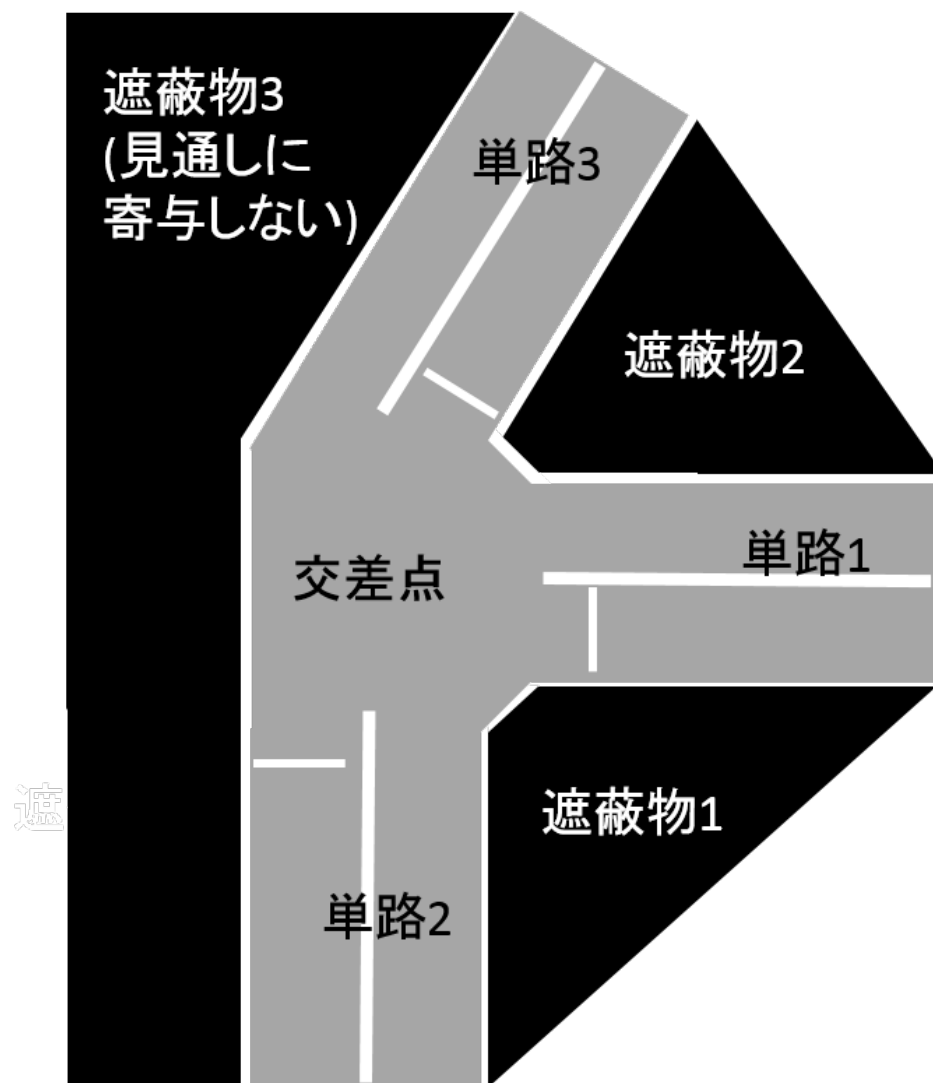


図 8 三叉路における遮蔽物

3.2.3 見通し計算とエラー

見通し計算について説明する。見通し計算には二つのエージェントを結ぶ線分をエージェントどうしの視線として、遮蔽物の対角線と交差判定することにより認知ができるかどうかを判定している。交差判定には外積による計算を用いている。図 9(左) は認知ができていない状態を表し、図 9(右) では認知ができていない状態である。この見通し計算は全ての遮蔽物が、各タイムステップで行っている。遮蔽物は生成された時に隣接する 2 つの単路と交差点の情報を保持している。そこから 2 つの単路上に存在し、かつ、遮蔽物が保持する交差点に向かっているエージェントの情報を参照し、それらの見通し計算を行う。ここで認知エラーが発生した場合、交錯するエージェントを認知することができなくなる。

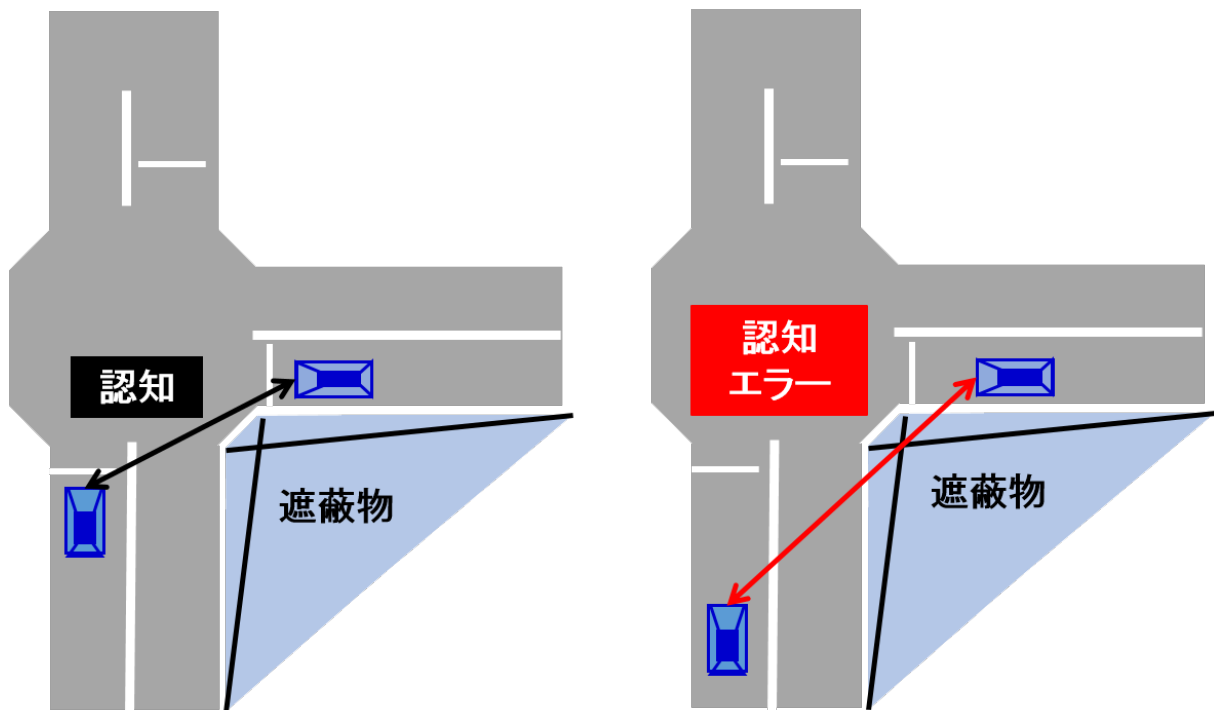


図9 （左）認知できているとき （右）認知エラーが生じているとき

3.2.4 傲慢な判断エラーの発生

傲慢な判断エラーが発生するかどうかは、確率的に決定している。判断エラーが起きるかどうかの決定は、常に同じエージェントがいつも、傲慢な判断エラーを起こすとは限らないという考えから、車両生成時ではなく、単路に進入するときに行っている。判断エラーがおきる確率を出会い頭事故のエラー率とする。見通しの悪い単路を通して交差点に進入するエージェントのうち、傲慢な判断エラーが起きていないエージェントについては、「遮蔽物の陰から車両が飛び出してくるかもしれない」と判断して交差点の前で一時停止をするようにした。逆に判断エラーが起きているエージェントは、交錯する車両の存在を確認しないため、出会い頭事故が発生する可能性がある。図10に出会い頭事故が発生するまでのフロチャートを示す。

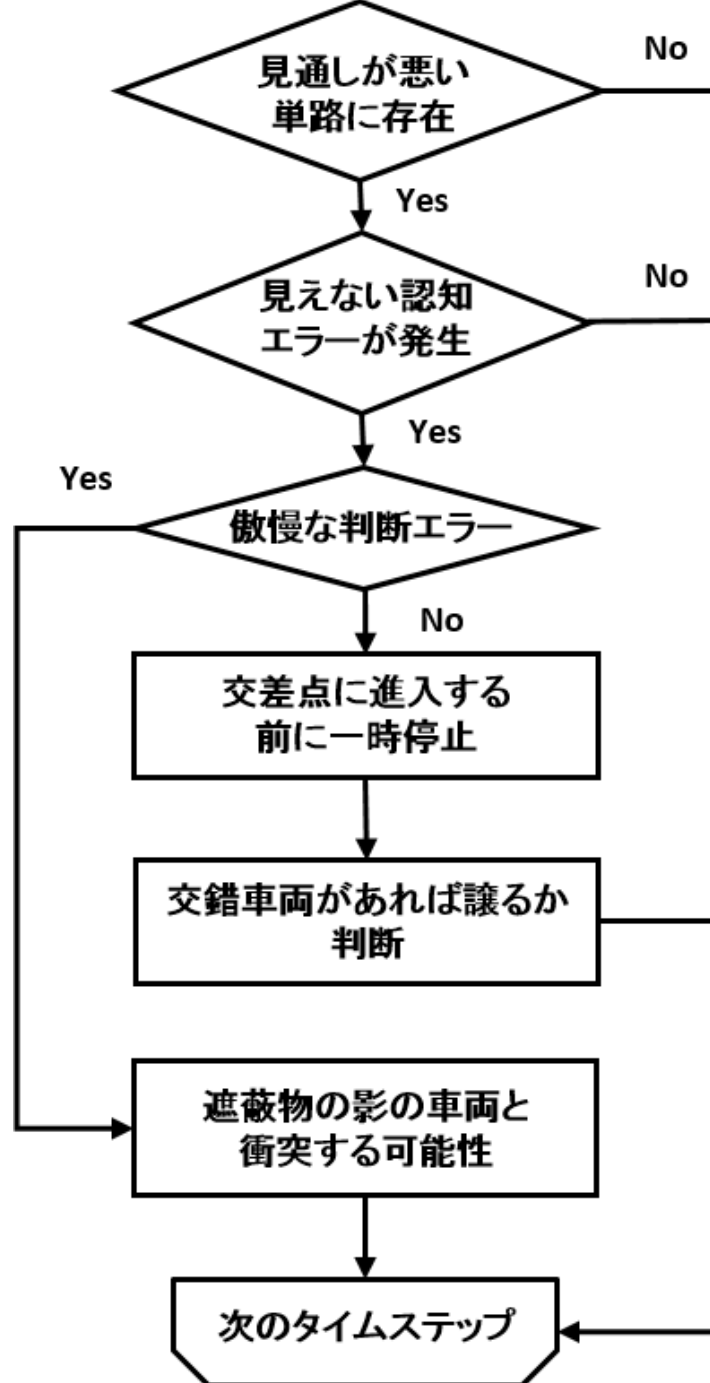


図 10 出会い頭事故のプロチャート

3.3 右左折事故

右左折事故の多くは信号で整理されている交差点で発生するものが多い。ここでモデル化する左折信号のある交差点で自車が左折しようとしてかつ対向車が右折をしようとしてきた時に発生するものである。交差点が前方に存在し、対向車が存在する場合に傲慢な判断エラーが起きる可能性がある。この傲慢な判断エラーが起きる割合を右左折事故におけるエラー率とする。エラーが生じている場合には自車と対向車が交差点を通過し切る時間を計算する。また、ここで自車が通過仕切る時間を $T1$, 対向車が通過仕切る時間を $T2$ とする。ここで判断エラーが発生しているエージェントは自車が通過仕切る時間を早く見積もるという考えから、その誤差を誤差時間 ΔT とすると、 $T1 - \Delta T$ が $T2$ より小さければ、自車の方が対向車より早く通過できると予測したとなり、実際には交差点に進入すると衝突する可能性がある場合でも、交差点へ進入する。実際には対向車が先に交差点に進入すべき時に自車も交差点に進入することがあるため、衝突事故が起こる可能性がある。右左折事故が起きるまでのフロチャートを図 11 に示す。

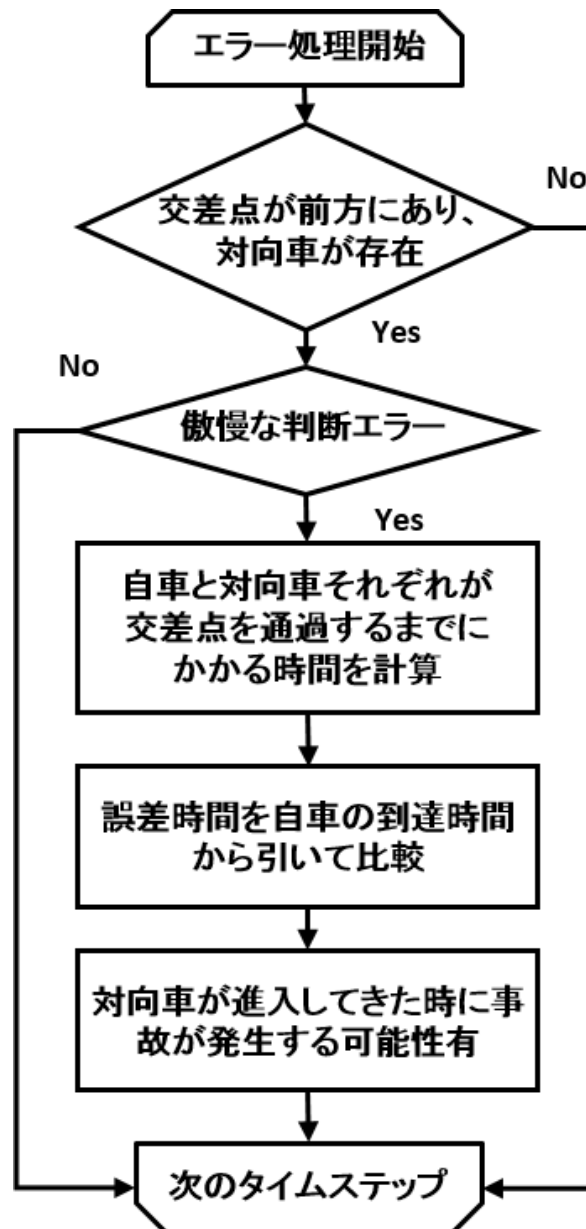


図 11 右左折事故のフロチャート

3.4 進路変更事故

進路変更は運転行動の中でも複雑な行動となり、危険な事故が発生する可能性がある。自動車技術会の HP で公開されている、典型的な進路変更による事故やヒヤリハットの動画 [12] によると、進路変更事故の原因の多くは脇見や不注意により、後方確認をせず、後方から来た車両と接触するというものが多い。そのため、本研究のモデルでは、見ない認知エラーが起ること、後方に車両があった場合であっても認知せずに進路変更を始めるというものとする。また、このエラーが発生するかどうかは車線変更を始める時に確率計算される。ここで MATES においてエージェントが車線変更を行うタイミングを説明する。エージェントが単路にいる時に、次の交差点で右左折、直進するために車線変更をしなければならない時であり、追越・追抜等で車線変更をするということはない。確率計算に関しては追突事故と同じように、認知対象物が多ければ多いほど、エラーが生じる確率が高くなる脇見運転モデルを用いる。また、このエラーが起きやすさを進路変更事故のエラー率とする。この見ない認知エラーが起こった時に進路変更による事故が発生する可能性がある。進路変更事故が起きるまでのフロチャートを図??に示す。

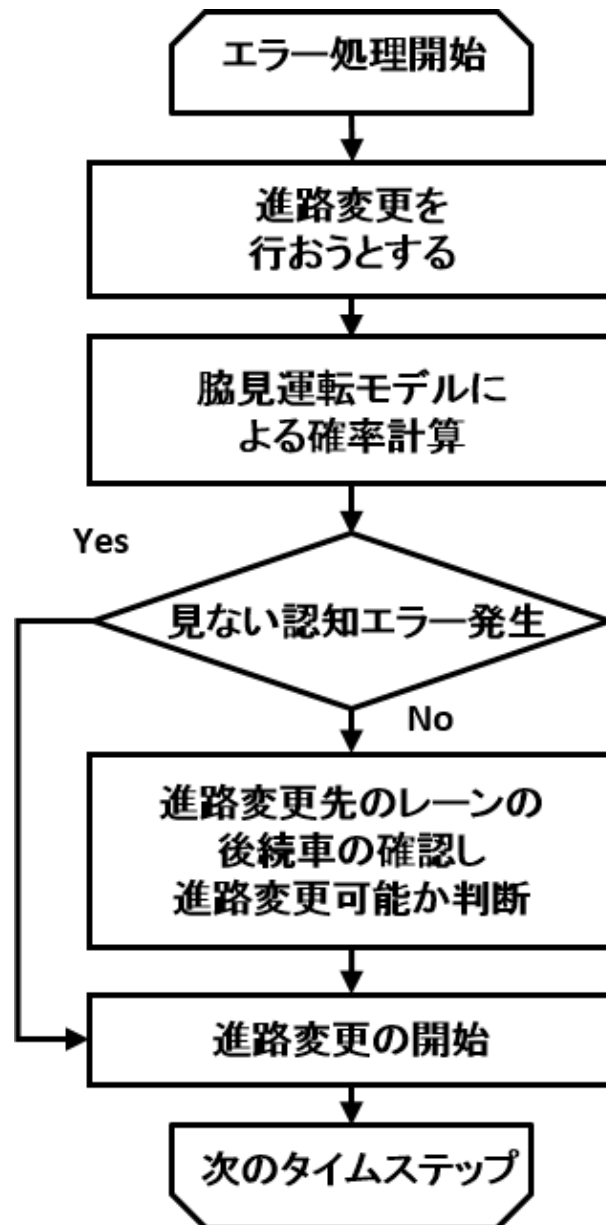


図 12 進路変更事故のフロチャート

3.5 正面衝突事故

正面衝突の事故は件数こそ少ないものの、衝突する車両どうしの速度が非常に速いことや、衝突する位置の関係上、重症事故・死亡事故になることが多く非常に重大な事故である。勝岡らの報告 [13] によれば、典型的な正面衝突事故は居眠りや、不注意によって車両が対向車線に進入することにより発生することが多い。そこで正面衝突では見ない認知エラーが発生することで、対向車線に接する最内車線を走行中に対向車線に進入するというモデルを構築する。

まず対向車線が存在するかどうかは、同じ単路内の車線を全て取得し、エージェントが単路内同一方向において、一番右側の車線（これを最内車線と呼ぶ）に存在するかどうかを確認することで確認する。エージェントが最内車線にいて、対向車線が存在する時に確率計算を行い、見ない認知エラーが発生するかを決定する。このエラーが起きる確率を正面衝突事故におけるエラー率と定義する。エラー発生時に、車両を対向車線側に移動するということとした。移動した後は元の位置から右側へ 3m 移動した後に元の位置へ戻るものとする。対向車両が存在するときに正面衝突の事故が起こる可能性がある。正面衝突事故のフロチャートを図 13 に示す。

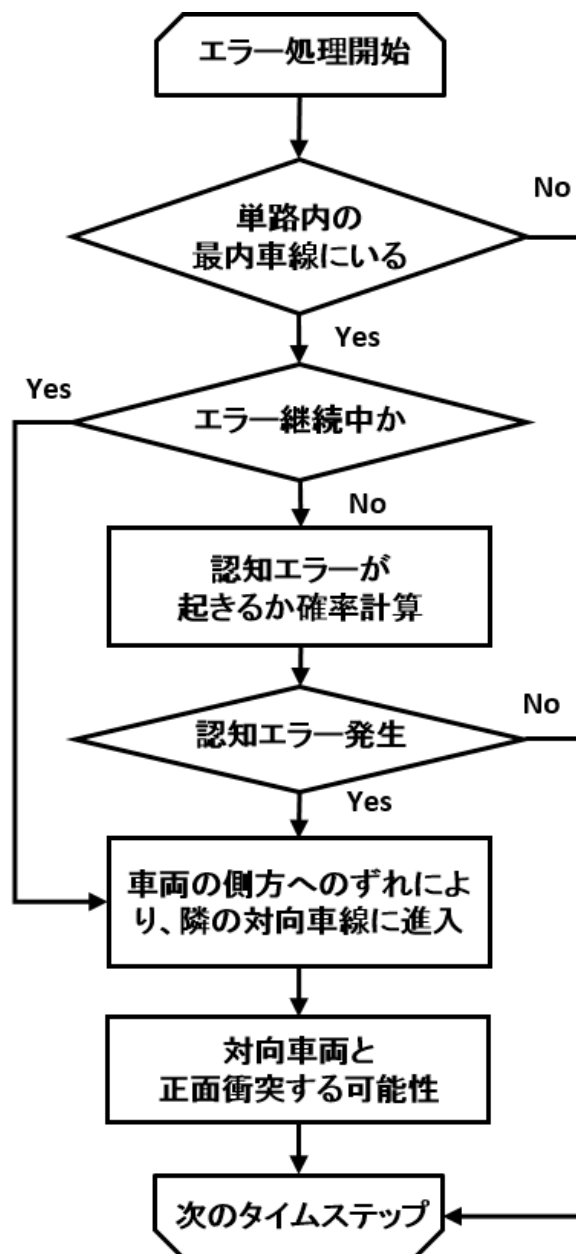


図 13 正面衝突事故のフロチャート

3.6 衝突判定

ここでは衝突判定のアルゴリズムについて説明する。計算負荷低減のため、衝突判定はエラーが起きているエージェントと衝突対象のエージェントに対してのみ行い、かつ事故の種類によって違う衝突判定を行っている。以下 3 つの種類の衝突判定を挙げる。

3.6.1 追突事故の衝突判定

追突事故の衝突判定は先行車との衝突であり、一次元的な考えで判定ができるため、比較的簡易なアルゴリズムを使用している。図??を用いて説明する。エージェント 1 とエージェント 2 のの中心の位置を取得し、その距離を d とする。また、1,2 それぞれの車体の進行方向の長さを l_1, l_2 とすると以下の式が成立するときに、衝突したと判定する。この判定は追突の可能性のあるエラーがあるときのみ、先行車判定の部分に実装している。

$$d < l_1 + l_2 \quad (1)$$

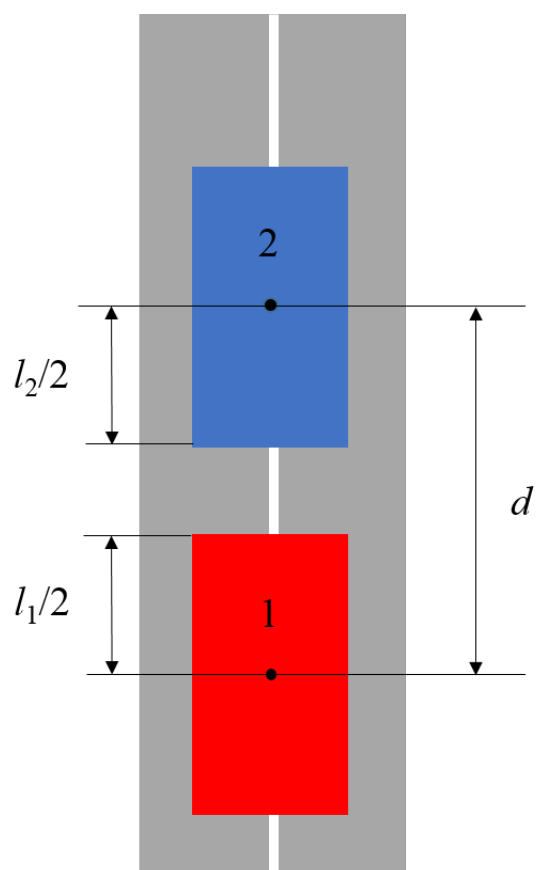


図 14 追突事故の衝突判定

3.6.2 進路変更・正面衝突事故の衝突判定

進路変更、正面衝突事故の衝突判定も基本的には追突と同じ手法で行うが、車体が横にずれるため二次元的な動きになる。よってエージェント 1 とエージェント 2 の距離の x 成分を d_x , y 成分を d_y とし、1,2 それぞれの車体の進行方向の長さを l_1, l_2 、幅方向の長さを w_1, w_2 とすると以下の 2 つの式が成り立つときに、衝突したと判定する。

$$d_x < w_1 + w_2 \quad d_y < l_1 + l_2 \quad (2)$$

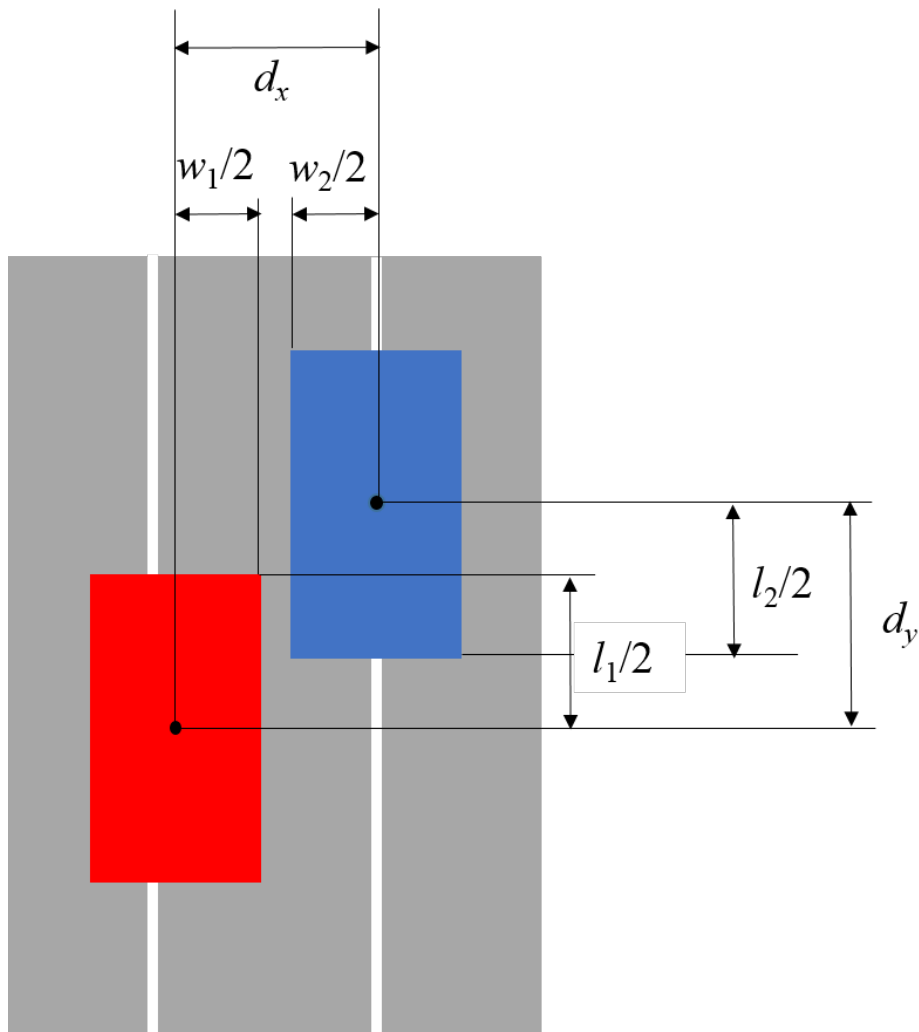


図 15 進路変更・正面衝突事故の衝突判定

3.6.3 出会い頭・右左折事故の衝突判定

最後に出会い頭・右左折事故の衝突判定であるが、この判定では衝突時車体どうしが並行な状態ではないため、進路変更・正面衝突事故のアルゴリズムでは対応しきれない。そのため少々複雑な計算手法を用いた。2つの長方形が接触していないということは、長方形の間に線分が引けるということに言い換えられる。ここでその様な2つの長方形の間の線分に垂直な軸を分離軸とする。また、分離軸に対する長方形の射影を投影線分と呼ぶ。ここでは投影線分が離れていれば交差していないということになる。(図16)

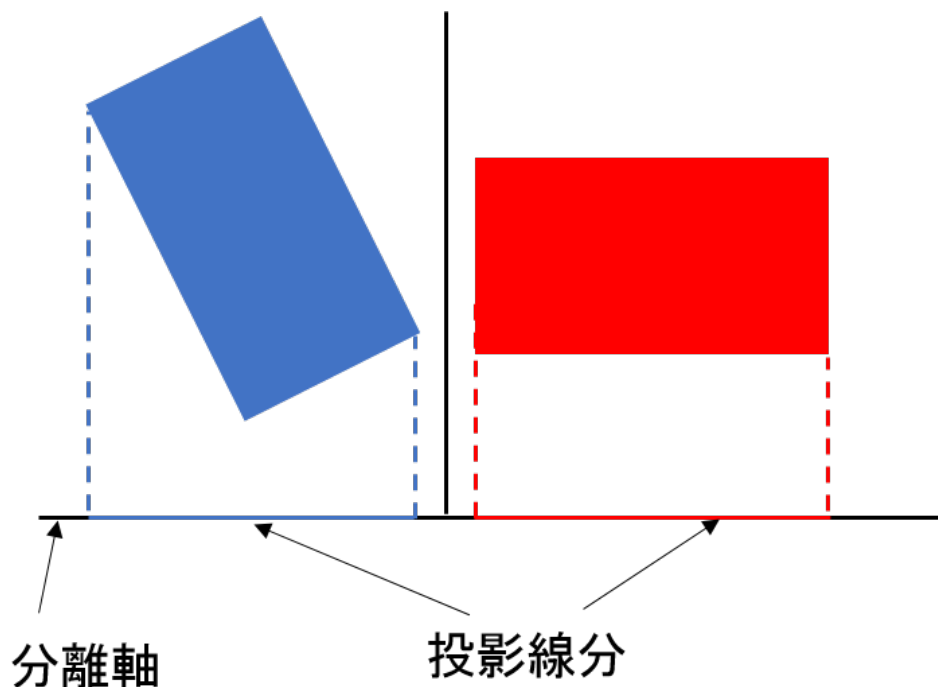


図 16 分離軸と投影線分

長方形が接触しているということは結局辺が交わっていると考えられるため、次に線分の交差について分離軸を用いて考える。またこの概念を線分についても考える。図 17（左）では 1,2 の線分は交わっている。これを分離軸を使って考えると図 17（右）の様に投影線分に対してもう片方の線分の射影を凝縮点と呼ぶ。この凝縮点が互いの投影線分に含まれていれば、2 つの線分は交わっている。図 18 は、交わっていない線分について、表したものである。

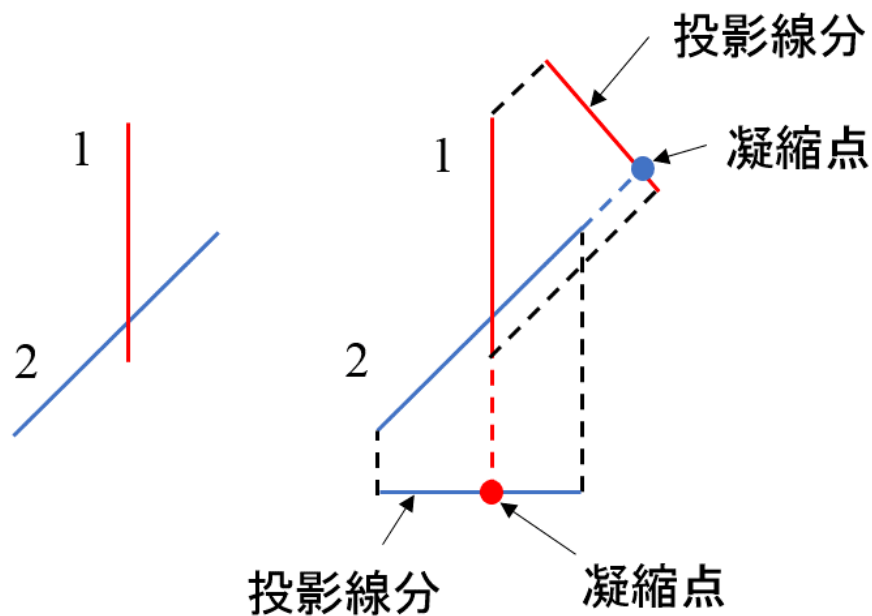


図 17 （左）交差した線分（右）投影線分と凝縮点

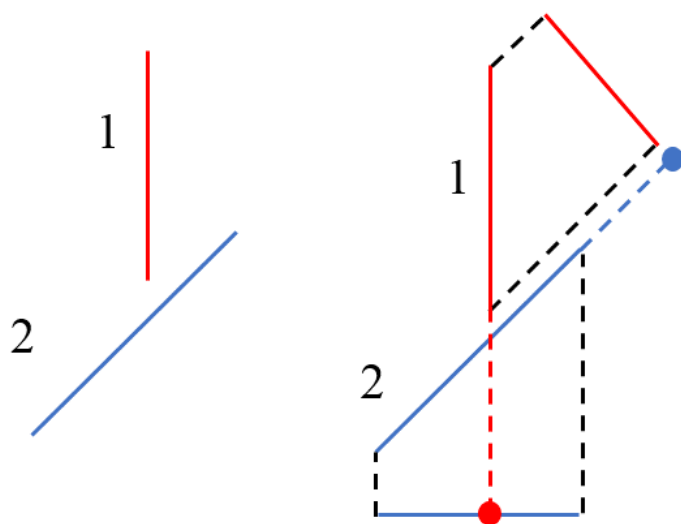


図 18 （左）交差していない線分（右）投影線分と凝縮点

車体が長方形であるため、対辺が平行なため、投影線分は全部で4本だけでよい。これは単純にお互いの辺の交差判定を $(4 \times 4=)16$ 回するよりかなり少ない計算量で済む。図 19 の様に投影線分と凝集点が、接していない時にエージェントどうしは衝突していないと判定できる。図 20 のように計算には 1,2 の投影線分への距離を d 車体の投影線分への射影の半分の r_1, r_2 とすると、以下の式

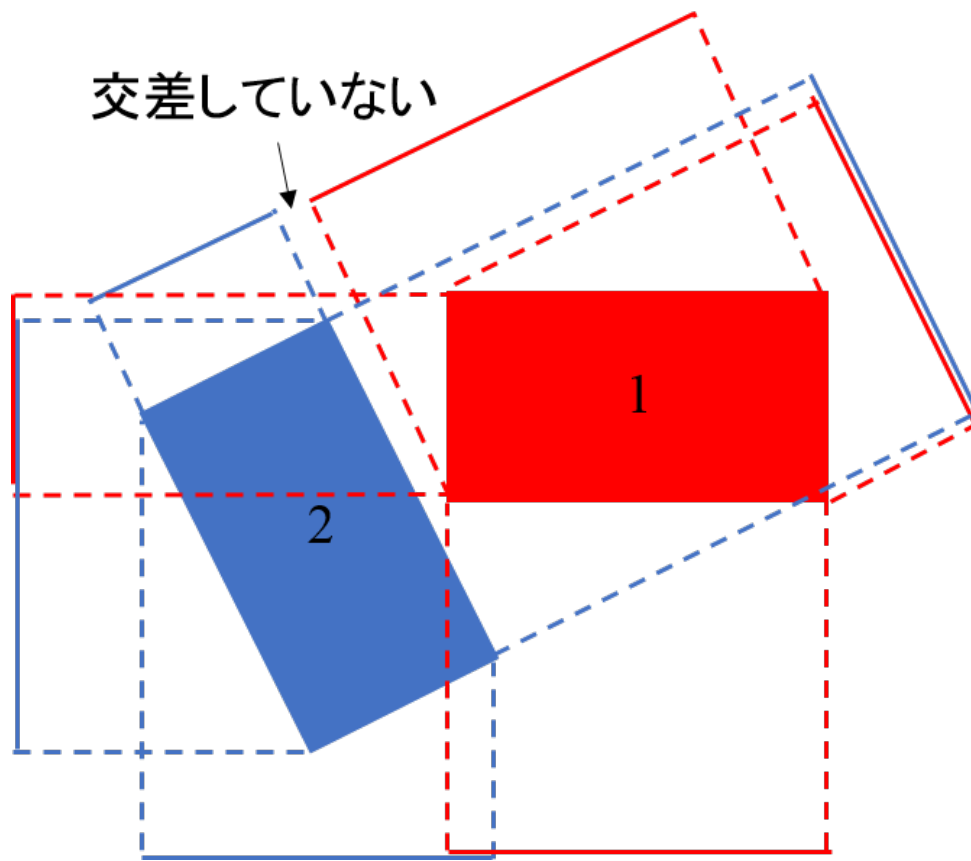


図 19 4つの投影線分

$$d < r_1 + r_2 \quad (3)$$

が4つの投影線分全てについて成り立つときに、エージェントどうしが衝突したと判定する。

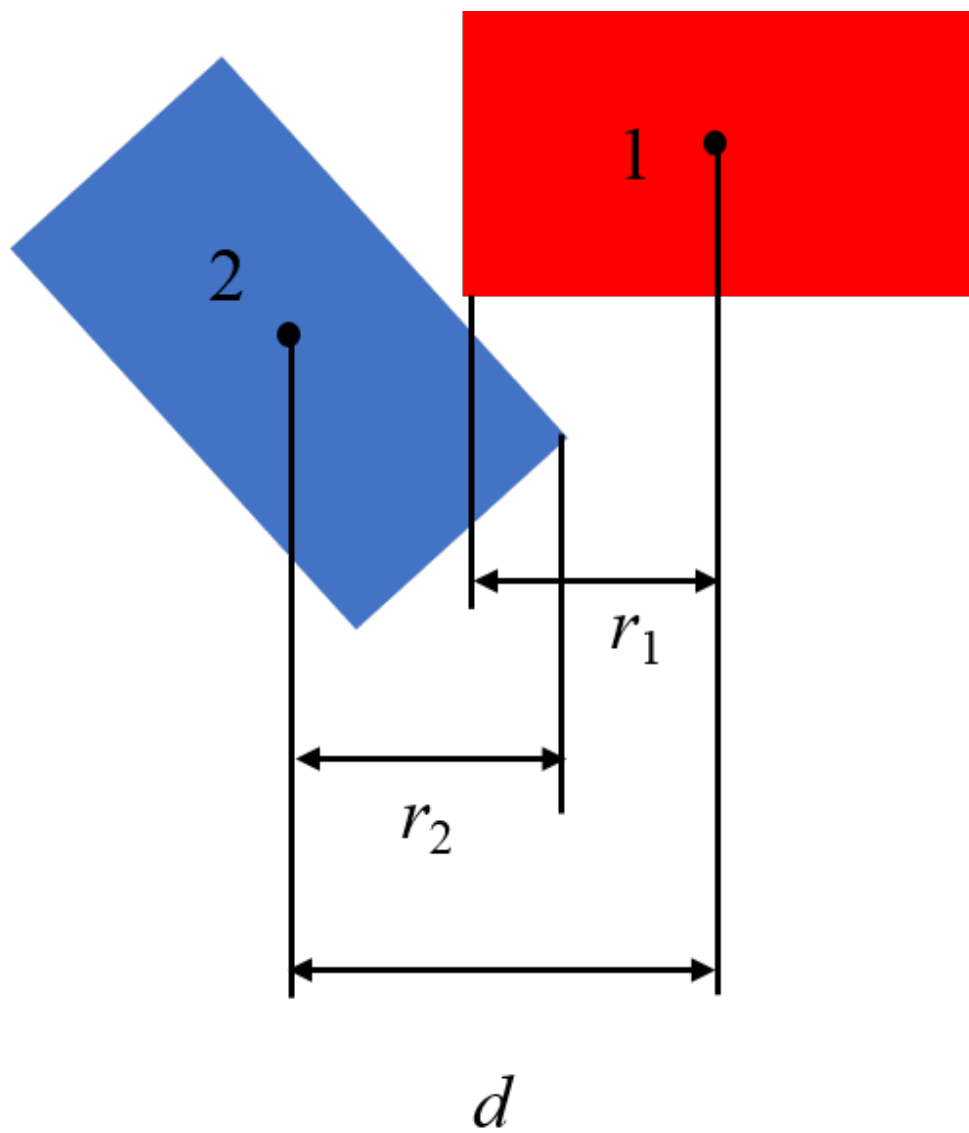


図 20 衝突判定の計算

4 挙動検証と感度解析

実装したそれぞれの事故について、挙動の検証および感度解析を行った。画像について、エラーが発生していないエージェントは青色、エラーが発生しているエージェントは赤色となっている。また、事故が発生したときのエージェントは黒色で表示される。

4.1 追突更事故

4.1.1 挙動検証

4.1.2 感度解析

4.2 出会い頭事故

4.2.1 挙動検証

4.2.2 感度解析

4.3 右左折事故

4.3.1 挙動検証

4.3.2 感度解析

4.4 進路変更事故

4.4.1 挙動検証

4.4.2 感度解析

4.5 正面衝突事故

4.5.1 挙動検証

4.5.2 感度解析

6

5 実都市への応用

6 結論

7 はじめに

本文はただのふつうの文章なのでとにかくふつうに入力する。ソースファイルの中で改行する場所は基本的に自由で、改行が本文に反映されることはない。MS Word だと好きなところで改行できるけど、ふつうの文章で段落の途中で改行する必要とかなないよねというのが L^AT_EX 的思考。改段落には空白行を一行入れる。以降はよく使うコマンドを書いていく。内容は LaTeX コマンドシート一覧 [?] を参考にした。このサイトは非常に便利なので何度も参照するといいと思う。ちなみに、参考文献の参照には相互参照という仕組みを使う。L^AT_EX ではラベルを適当に付けて相互に参照することで、参考文献、数式、図、表の番号がズレることを防止できる。もちろん参考文献の参照を [1] のように直打ちするのもダメではない。

8 空白と強制改行

段落の中での改行は基本的にはやらないが、強制改行することができる。このとき、強制改

行と共に 1 cm 空けるみたいなこともできる。また、水平方向にもで、強制的に空白が作れる。コマンド名前から想像できるように、垂直方向の空

白を空けるコマンドもある。この辺のコマンドは学位論文の表紙を作るときに使える。ちなみに、単位は cm の他に、mm、in (インチ)、em (大文字 M

の幅)、ex (小文字の x の高さ)、pt (ポイント) あたりがよく使われる。特に em と ex は、フォントサイズに対して相対的な大きさなので意外と使える。

9 箇条書きと書体

よく使う書体の一覧を示す。

1. Roman
2. *Italic*
3. **Bold**
4. Sans Serif
5. Monospace

フォントサイズも手動で設定できる。あまり使わないけど。

tiny フォントサイズ
scriptsize フォントサイズ
footnotesize フォントサイズ
small フォントサイズ
normalsize フォントサイズ
large フォントサイズ
Large フォントサイズ
LARGE フォントサイズ
huge フォントサイズ
Huge フォントサイズ

10 数式

数式に関する細かいコマンドは [?] を参照してもらうとして、ここでは数式の基本についてのみ記す。まず、数式は必ず数式モードの中で書く。数式

モードにはふつうの数式と文章中の数式の 2 種類がある。前者は

$$y = \alpha x + y \quad (4)$$

のように書く。後者は $y = \alpha y + x$ のように書く。ちなみに、ギリシャ文字は数式モード内でしか使えないので、文章中では α のように書く必要がある。他には、プリアンブルで定義したコマンドを使って

$$\boldsymbol{q} = \boldsymbol{A} \boldsymbol{p} \quad (5)$$

のように行列やベクトルを書いたり、この式に対して相互参照で式 (5) と参照できたりする。

$$\boldsymbol{z} = \boldsymbol{M}_{-1} \boldsymbol{r}$$

のように数式番号を消すこともできる。

11 表と図

表と図は結構むずかしい。基本的には HTML と同じような思考で取り組むと大体正しい。

11.1 表

表 1 に表の例を示す。

表 1 表のサンプル

右	中央	左
みぎ	ちゅうおう	ひだり
Right	Center	Left

11.2 図

画像は画像ファイルを用意するのがめんどうなのでコメント内にサンプルを書いておく。EPS 形式と PNG/JPEG 形式の例を順に示す。L^AT_EX では

EPSの方がメジャーで、グラフとかポンチ絵とかはできるだけEPS形式で作るのが \LaTeX 脳。グラフはGnuplot、ポンチ絵はInkscapeで描けばいいと思う。写真や可視化の図は仕方がないからPNG形式やJPEG形式で作る。詳しくはラスタ画像とベクトル画像をググってほしい。

12 その他小ネタ

脚注^{*1}が個人的に好き。

参考文献

- [1] 特定非営利活動法人 ITS Japan. ITS とは. <http://www.its-jp.org/about/>
- [2] 国土交通省, ITS スポット http://www.mlit.go.jp/road/ITS/j-html/spot_dsrc/
- [3] 国土交通省, 自動車総合安全情報 衝突被害軽減ブレーキ <http://www.mlit.go.jp/jidosha/anzen/01asv/esc.html>
- [4] 警察庁交通局, 平成 24 年中の事故発生状況, 2013
- [5] 交通事故分析センター, ITARDA Information, No.56, 2005
- [6] 古川修, 安全運転支援システムの効果評価のための UDM を用いた交通流シミュレータの開発, 自動車技術, Vol.63, No.2, pp.104-107, 2009
- [7] 藤井秀樹, 吉村忍, 高野悠哉, マルチエージェント交通流シミュレーションにおける交通事故モデリング, 人工知能学会論文誌, Vol.26, No.1, pp.42-49, 2011
- [8] 自動車技術会, 交通事故予測シミュレーションシステム検証マニュアル, 2013
- [9] 山田真大, 青木優, 日高隆博, 山崎二三雄, 佐藤健哉, 高田広章, 交通事故シナリオに基づく予防安全システムのシミュレーション分析, 情報処理学会研究報告. ITS, [高度交通システム] Vol41, No.2, pp.1-8, 2010

^{*1} ふっとのーと

- [10] 交通事故総合分析センター, 追突事故はどうして起きるのか”, ITARDA INFORMATION, No.43, 2003
- [11] 交通事故総合分析センター, 出会い頭事故における人的要因の分析, ITARDA INFORMATION, No.56, 2005
- [12] 自動車技術会, ドライブレコーダは見た! 自動車技術会「ヒヤリハットデータベース」より, <http://www.jsae.or.jp/hiyari/1009/>
- [13] 勝岡秀明, 四輪車の正面衝突事故発生状況と特徴の分析, 交通事故総合分析センター研究発表会論文集, 2010
- [14] 紳士太郎. ふええの非定常音響解析における減衰振動特性. 日本計算萌え学会論文集, Vol. 2, No. 1, pp. 1–4, 2011.
- [15] T. Shinshi. “Large deformation simulation of golden-brown hair with twin tails.” *Journal of Computational Moe*, **5** (2): 17–24, 2010.