

決定木による自動運転車の 交差点協調制御の検討

片桐 紳太郎¹・田代 むつみ²・三輪 富生³・森川 高行⁴

¹学生会員 名古屋大学大学院 工学研究科土木工学専攻 (〒464-8603 名古屋市千種区不老町)

E-mail: katagiri.shintaro@d.mbox.nagoya-u.ac.jp

²正会員 名古屋大学特任講師 未来社会創造機構

³正会員 名古屋大学准教授 未来材料・システム研究所

⁴正会員 名古屋大学教授 未来社会創造機構

本研究では、交差点における 2 車両の走行位置が重なることで生じる一旦停止を“コンフリクト”と定義し、それを解消するための対象車両の制御方法について検討を行った。すべての車両が自動運転車である状況を想定し、現在の交通状況からシミュレーションによって将来の交通状況を予測する。その結果、近い将来のコンフリクトの発生が予測される場合には、周辺道路の混雑状況や各車両の位置などのリアルタイム情報を入力情報として、予め学習しておいた決定木によりコンフリクト対象車両の適切な制御方法を決定するものである。セル・オートマトンにより T 字無信号交差点の交通流を再現し、その制御効果を検証した結果、8 割程度のコンフリクトで最適な制御パターンを正しく選択でき、総旅行時間を減少させることが示された。

Key Words: cooperative control, autonomous vehicle, conflict, microsimulation, travel time

1. はじめに

近年、急速に進められる自動走行技術の開発と低遅延かつ大容量通信を可能とする通信インフラの整備によって、次世代交通の主角と目される CAV (Connected and Autonomous Vehicle) の実現が近づいている。CAV は車車間や路車間通信によって、高精度かつ広域的な交通情報をリアルタイムに取得・共有することができ、各車両は状況に応じた最適な走行パターンを適宜選択することが可能となると考えられる。中でも、交通渋滞の主な原因となる交差点での合流における車両の協調的走行計画は、従来の信号制御に代わる新たな合流支援手法として、近年注目を集めている。

協調制御を利用した合流支援手法の提案や評価に関する研究は広く行われている。例えば、Meng ら¹⁾は合流制御アルゴリズムを二つに分類し、それぞれの代表的な手法を比較した。一つは交差点をセルに分割し、各車両が交差点進入時刻におけるセルの占有を予約し、それに合うように速度の調整を行う手法である。もう一方は、交差点での走行軌跡の交錯を管理し、車両追従モデルによる速度変更を組み込んだ制御アルゴリズムを用いて車両の速度を調整し全体的な走行計画を立てる手法である。

この研究では、交通量の増加に従い、後者は前者に比べて交通流の整流化効果は高くなるものの、計算量が膨大となる点が指摘された。その他にも、対象範囲内各車両の速度や加速度を変数とする最適化問題を定式化し、最高速度や安全車間距離を制約条件として求解する方法が多く研究されている²⁾⁻⁵⁾。しかし、車両台数の増加とともに条件が複雑化するため、例えば 1 秒後に対象範囲内の全ての車両がとるべき行動の最適解を、それより短い時間で求めることは容易ではない。このため、複雑な最適化問題を解くことなく、対象とする車両の制御方法が得られることが望ましい。田代ら⁶⁾は、交差点に流入する車両の走行予定軌跡が別の車両の走行軌跡と交錯し、一旦停止をしなければならない状況を“コンフリクト”と定義している。その上で、交通情報をリアルタイムに収集する技術と CAV の通信機能が活用できる将来を想定して、近い将来のコンフリクト発生を検知し、道路の優先関係などの既存の交通ルールに縛られない車両制御方法をとることによってそれを回避することで、交通混雑の緩和が可能であると示した。しかし、田代らの研究でも計算負荷が大きな課題であり、検出されたコンフリクトに対して多数回のシミュレーションを実行して最適な制御方法を決定する必要があった。

そこで、本研究では、田代らの研究と同様にコンフリクトを定義し、対象となる 2 車両の最適な制御方法を、多様な交通状況を想定したデータを用いて学習した決定木によってモデルし、計算負荷の軽減を試みる。セル・オートマトンによるマイクロシミュレーションを実行し、周辺車両の旅行時間も考慮して制御効果を分析した。

2. 交差点ネットワークとシミュレータの概要

本研究では、個々の車両の挙動を表現するマイクロシミュレータとして、セル・オートマトンのNaSchモデル⁷⁾によって、信号の無い交差点での合流と車線変更を表現可能な交通シミュレータを構築した。

(1) 道路ネットワークと入力交通量の概要

本研究では、図-1に示す道路ネットワークを用いた。リンク長は全て300mとし、横方向のリンク（ノード1-4間、およびノード2-4間）は片側2車線、縦方向のリンク（ノード3-4間）は片側1車線とした。各リンクは普通車の平均的な車両長（4~5m）と最低車間距離（2.5m）を参考に、一つのセルに一台の車両が配置可能となるよう長さ7.5mのセルで分割した（1リンクは40セル相当）。また、1タイムステップは1秒とした。

各リンクの最高速度は、横方向のリンクの方が縦方向のリンクよりも大きいとし、横方向リンクで3セル/秒、縦方向リンクで2セル/秒とした（それぞれ、81km/h, 54km/hに相当）。また、協調制御が行われない場合は、車線数が多く最高速度が速い横方向のリンクを優先道路、縦方向のリンクを非優先道路とし、優先/非優先制御を行うこととした。

入力交通量は優先リンクへの発生車両台数（ x [台/10分]）と優先リンクから非優先リンクへの流出入交通量の割合（ y [%]）によって定義した。すなわち、優先リンクに接続するセントロイド（ノード 1, 2）から x [台/10分] が発生し、交差点（ノード 4）にて非優先側へ y [%] の車両が流出する。また、非優先側のセントロイド（ノード 3）からは、優先側から流入してくる車両台数と等しいの交通量（ x [台/10分] $\times y$ [%]）が発生する（表-1）。

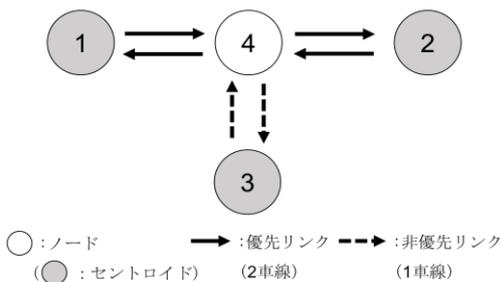


図-1 道路ネットワーク

表-1 $x=100$ [台/10分], $y=30$ [%]でのOD交通量の例 [台/10分]

		着ノード			計
		1	2	3	
発ノード	1	-	70	30	100
	2	70	-	30	100
	3	30	30	-	60
計		100	100	60	260

表-2 決定木構築のための発生車両数（20分間）

		非優先への流出割合 y [%] (非優先からの流入割合)				
		10	20	30	40	50
優先側発生車両台数 x [台/10分]	50	220	240	260	280	300
	60	264	288	312	336	360
	70	308	336	364	392	420
	80	352	384	416	448	480
	90	396	432	468	504	540
	100	440	480	520	560	600
	110	484	528	572	616	660
	120	528	576	624	672	720
	130	572	624	676	728	780
	140	616	672	728	784	840

(2) ローカルルール

ここでは、シミュレータに導入した、各車両の行動を決定するローカルルールを説明する。本研究では、車両発生ルール、速度ルール、車線変更ルール、リンク移行ルールの4種類をNaSchモデルに従って以下のように与えた。

a) 車両発生ルール

前述した車両台数の条件で20分間の車両発生を行った。一様乱数により車両の出発時刻を決定し、発生させたすべての車両が目的のノードに到着するまでシミュレーションを行った。また、決定木学習において多様な交通状況でのコンフリクトやそれを解消する制御方法を学習するため、表-2に示す50パターンの車両発生パターンで実験を行った。

b) 速度ルール

速度の変化は、以下のアルゴリズムによって行う。

```

IF ( $v \geq gap$ )
     $v = gap - p_{noise}$ 
ELSEIF ( $v < v_{max}$ ) THEN
     $v = v + 1 - p_{noise}$ 
ELSE
     $v = v - p_{noise}$ 
ENDIF
    
```

ここに、ある車両について、 v はその速度、 gap は走行車線上の前方車両との距離（前方ギャップ）、 v_{max} は

走行するリンクにおける最大速度である。 p_{noise} は確率 p で 1 となる乱数で、実際の交通行動における不確実性、すなわち前方ギャップが十分あっても減速する“ランダムブレーキ効果”を表現する。ただし本研究では、問題をよりシンプルに考えるため、全車両は自動運転であると仮定し、常に $p_{noise} = 0$ とした。

c) 車線変更ルール

車線変更は真横のセルに移動するものとし、速度更新・前方移動の前に、以下のアルゴリズムで行う。

```

IF (隣の車線のセルが空いている)
  IF ( $gap < v$  AND  $gap_o > gap$ )
    THEN  $w1 = 1$ 
    ELSE  $w1 = 0$ 
   $w2 = \max[v - gap_o, 0]$ 
   $w3 = \max[v_{max} - gap_b, 0]$ 
   $w4 = \max[\frac{d^* - d}{v_{max}}, 0]$ 
   $w5 = \max[d^{**} - d, 0]$ 
IF (交差点を直進予定)
  IF ( $w1 > w2$  AND  $w1 > w3$ ) THEN 車線変更
ELSEIF (交差点を右折/左折予定)
  IF ( $w1 + w4 > w2$  AND  $w1 + w4 > w3$ )
  IF ( $w5 = 0$ ) THEN 車線変更
  ELSEIF ( $w5 > 0$  AND 右端/左端車線にいない)
    THEN 右端/左端車線に車線変更
  
```

ここに、 gap_o は隣の車線の前方車両とのギャップ、 gap_b は隣の車線の後方車両とのギャップ、 d は前方交差点までのセル数で、 $w1 \sim w5$ は車線変更を決定する重みである。複数車線から成るリンクを走行中、次に右折/左折の予定であれば、リンク終端では右端/左端車線に存在する必要がある(直進時は、車線位置の制限無し)。このようなリンク移行のための車線変更の緊急度が、リンク終端に近づくほど増すことを、 $w4$ により表現している。一方で、右折/左折のため右端/左端車線に一度入った後は、交差点に近づくほど、不要の車線変更を避けその車線を維持することが望ましい。本研究ではこの効果を、 $w5$ を導入することにより表現した。 d^*, d^{**} はセル数により与えられるパラメータであり、本研究では、交差点の約 200m 前からこれらの効果が発揮されるように、 $d^{**} = d^* = 26$ と設定した。

d) リンク移行ルール

無信号交差点では、リンクの優先・非優先関係および自車と走行軌跡が重なる他車との位置関係を考慮して、安全なリンク移行を行う必要がある。本研究では、無信号交差点で自車と走行軌跡が重なる他車両の交差点到着時間が 3 秒以上、かつ、移行先リンクの 1 つ目のセルに他車両が存在しない場合に、リンク移行が可能とした。

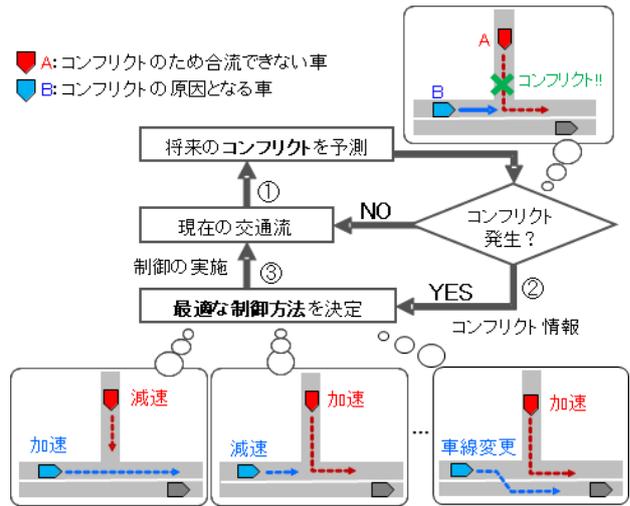


図-2 学習データのための最適制御方法の探索

3. 協調制御方法のための決定木構築の準備

(1) 決定木による協調制御方法の学習の手順

協調制御は、コンフリクト回避のために、通常時の走行から外れた特殊な走行指示を車両に与えることをいう。また、コンフリクトとは、ある車両 A が無信号交差点に進入しようとするときに、同じ交差点に進入する車両 B の存在によって交差点への進入が妨げられる状態である。

本研究では、決定木によって効果的な制御方法を学習することを試みるが、そのためには制御方法とそれに対応した結果に関する情報が学習データとして必要である。そこで、シミュレーション上で、コンフリクトを検出し、各コンフリクトに候補となる制御パターンを全て実施し、結果を比較することで最適な制御方法を決定する。さらに、その際の交通状況などに関する情報と最適な制御方法を出力しておき、これを学習データとして決定木を構築する。図-2 は、このうち、学習データのための最適な協調制御方法の探索について示している。

以降では、コンフリクト予測方法 (2)、最適な制御方法の選択方法 (3)、決定木への適用方法 (4) について、それぞれ説明する。

(2) 将来のコンフリクト予測と制御対象車両の特定

将来の交通状況の予測は、各タイムステップに実行する。現在の各車両の位置や速度等の状況を初期値として、対象範囲内の全車両の近い将来までの挙動をシミュレーションする。この予測は、現在の時刻から α 秒先まで行うこととし、この間にコンフリクトが発生すれば、その時点で予測シミュレーションを中断する。そして合流を妨げられた車両 A と妨げた車両 B を特定し、コンフリクト発生時の各車両の状態(走行リンク、走行車線、位置、速度)を記録する。一方、 α 秒の間にコンフリクトが発

生しなければ、再び現在の時刻に戻り、通常のローカルルールに基づき、次のタイムステップにおける車両挙動を算出する。ただし、コンフリクトが発生すると予測された車両の現在位置が、交差点近傍（9 セル以内）である場合、車両制御をとりうる十分な余裕がないとして、コンフリクト検知の対象外とした。またこれに合わせて、車両が交差点より離れた位置から制御を開始できるよう本研究では α を 10 秒とした。

(3) 最適協調制御方法の選択

コンフリクト回避のために、ある特定の車両の運転行動を制御すると、その周辺車両にも影響が及ぶ。そこで本研究では、コンフリクトを起こす 2 台の制御対象車両に対して複数パターンでの運転制御を行い、周辺車両への影響も考慮しつつ、各パターンにおける将来の交通流をシミュレートすることで、交通流全体で望ましい制御方法を選択する。

制御方法として、本研究では、車線変更 (L)、高速走行 (A)、中速走行 (M)、低速走行 (D)、停止 (S) の 5 種類に加え、制御を行わない場合 (N) (道路の優先・非優先関係に従う) も比較対象とした。そして、これら 6 種類の制御を組み合わせから、表-3 に示す計 26 種類の制御パターンを制御候補とした。各制御はコンフリクト回避を目的としているため、2 車両の制御内容が同じになる組み合わせ (例：車両 A も車両 B も高速走行制御) は対象外としている。また、車線変更の制御は、コンフリクトの原因となる車両 B が優先リンク (片側 2 車線) を走行している場合にのみ適用する。

制御は、対象交差点に直結するリンク内のみで行われるとする。そして、2 台の制御対象車両のうち、どちらか 1 台が対象交差点を通過した時点で、制御を終了するものとした。ただし、制御終了後もシミュレーションを継続し、ネットワークに流入する全ての車両が交差点を通過するまでの総旅行時間を計測した。そして、制御を行わない場合の総旅行時間を基準として、総旅行時間を最も減少させる制御の組み合わせを最適解とした。

なお、制御中の、車線変更、高速走行、中速走行、低速走行、一時停止は、以下のローカルルールに従って行った。

a) 制御時の車線変更ルール

車線変更によるコンフリクト回避は、コンフリクトの原因となる車両 B が以下の全ての条件を満たす時のみ、適用される。

- ① 制御開始時に複数車線から成るリンクを走行中
- ② 制御開始時にコンフリクトを起こす車線を走行中
- ③ 制御による車線変更が予定経路から逸脱させない
ここで、③は、例えば、左折予定の車両がコンフリクト発生時に左端車線にいる場合など、リンク移行のために

存在すべき車線上でコンフリクトが発生した場合は、車線変更制御は適用されないことを意味する。なお、制御による車線変更では、周辺ギャップに関係なく、隣のセルが空いた時点で直ちに移動するとした。また、前方への移動処理は、後で説明する高速走行制御のルールに従い行うものとした。

表-3 2 車両の合流調停における制御パターン

		車両 A				
		高速走行 (A)	中速走行 (M)	低速走行 (D)	一時停止 (S)	制御無し (N)
車両 B	車線変更 (L)	A-L	M-L	D-L	S-L	N-L
	高速走行 (A)	—	M-A	D-A	S-A	N-A
	中速走行 (M)	A-M	—	D-M	S-M	N-M
	低速走行 (D)	A-D	M-D	—	S-D	N-D
	一時停止 (S)	A-S	M-S	D-S	—	N-S
	制御無し (N)	A-N	M-N	D-N	S-N	N-N

b) 制御時の速度ルール

速度変更等の制御は、以下のローカルルールで行う。なお、速度制御中であっても、前方ギャップが不十分の場合は、衝突回避のために減速して前方車両の直後まで移動する。

【高速走行制御】

速度変更のローカルルールに従い、最大速度 3 セル/秒 (81km/h) まで加速する。ただし、非優先リンク (最大速度 54km/h) の車両はこの制御をとらない。

【中速走行制御】

速度変更のローカルルールに従い、中速 2 セル/秒 (54km/h) まで加速減速を行い走行する。

【減速制御】

速度変更のローカルルールに従い低速 1 セル/秒 (27km/h) まで減速し、走行する。

【一時停止制御】

前方交差点までのセル数 d が 4 以内 (30m 以内) に到達した時点から速度 1 を下限として減速した後、もう 1 台の制御対象車両が交差点を通過するまで、交差点直前のセル ($d = 1$) で停止し続ける。制御終了後は、通常の速度ルールに従って、走行を再開する。

(4) 決定木の適用方法

本研究では、上述の 26 パターンの制御の組み合わせから最適であると選択されたものをクラス (離散型の被説明変数) とし、コンフリクトが検知された時点での交通状況予測子 (説明変数) として決定木を構築する。予測子は、車両の走行リンク、走行車線、走行中のセル、速度、進行方向の 5 つの情報を「車両データ」とする。また、ネットワーク内の交通状況を表現するため、車両台数、交通密度、空間平均速度、直進割合、右折割合、

左折割合の 6 つを「交通状況データ」として定義する。今回の学習では、コンフリクトを起こす 2 台の車両データ、コンフリクト車両が存在するセルの前後のリンク終端までの交通状況データ、交差点から範囲 $r(=1, 4, 7, 10, 20, 30, 40$ セル) 以内の交通状況データを予測子とし、計 594 の変数を用いる。ただし、交通状況データを取得する対象のリンクと車線の確認を容易にするため、リンク番号と車線番号、および交通状況データを合わせた、8 つのデータを一つの変数セットとした。

表4 予測子の内容と通し番号

データ種別		対象	開始番号	終了番号
車両データ	走行リンク	車両 A 車両 B	1	10
	走行車線			
	走行セル			
	走行速度			
	進行方向			
交通状況データ	車両台数	車両 A の 前方/後方	11	34
	空間平均速度			
	交通密度			
	直進割合	車両 B の 前方/後方		
	右折割合			
	左折割合			
リンク番号 + 車線番号 + 交通状況データ	リンク番号	r=1 以内	35	114
	車線番号	r=4 以内	115	194
	空間平均速度	r=7 以内	195	274
	交通密度	r=10 以内	275	354
	直進割合	r=20 以内	355	434
	右折割合	r=30 以内	435	514
	左折割合	r=40 以内	515	594

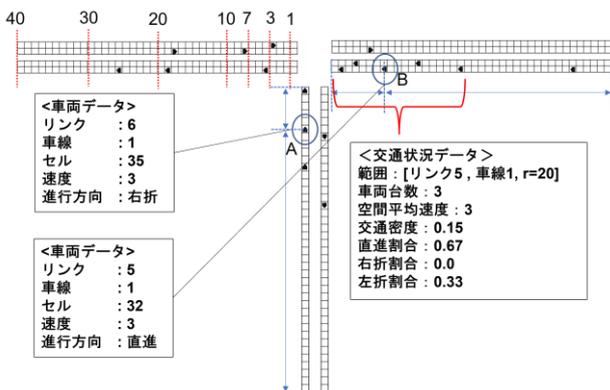


図3 予測子の取得イメージ

4. 決定木の構築と制御効果の分析

(1) 決定木の構築

決定木モデルは二分木構造による解の予測モデルである。親ノードに含まれるデータをある分類条件によって子ノードへ分類し、最終的に到達した葉ノードに含まれるクラスが予測結果となる。決定木学習では、各ノードの分類条件を教師データから決定する。教師データは、表2に示した 50 の交通量パターンごとに乱数によって車両の出発時刻を変化させつつ 1,000 回ずつシミュレーション (計 50,000 回) を行い、各シミュレーションで発生したコンフリクトのうちランダムに一つを抽出した。ただし、優先側発生交通量 x が小さく、かつ非優先側への流出割合 y も小さいケースではコンフリクトが発生しない場合があったため、取得できたデータ数 (コンフリクト数) は 49,842 となった。

CART (Classification and Regression Trees) アルゴリズムに従い、ジニ不純度 G を基準に木の作成を行う。

$$G = 1 - \sum_{i=1}^c p^2(i) \quad (1a)$$

$$p(i) = \frac{n_i}{N} \quad (1b)$$

$$N = \sum_{i=1}^c n_i \quad (1c)$$

ここで、あるノードについて、 c はそのノードに含まれるクラス数、 n_i はクラス i のデータ数、 N はそのノードの総データ数である。このジニ不純度 G は、ノード内のクラス数が少ない、つまり純粋であるほど 0 に近づくため、ジニ不純度 G を小さくするように以下の手順でノードを分割する。

【ノード分割手順】

1. 予測子 (説明変数) x_k の要素 w_{kj} を昇順で並べえる。
2. x_k の各要素 w_{kj} について、ノード t に含まれる観測値を 2 つの子ノード t_R, t_L へ分割する。
3. 親ノード t , 子ノード t_R, t_L のジニ不純度 G を求める。
4. 2 と 3 を全ての予測子と要素について実行し、不純度ゲイン ($\Delta I_{kj} = G_t - G_{t_R} - G_{t_L}$) が最大となる x_k, j を求める。
5. 求めた k, w_{kj} によってノード t を分割する。
6. ノードのジニ不純度が 0 またはデータ数が下限値に到達するまで 1~5 を繰り返す。

本研究で採用した終了条件は、 $G_k = 0$ もしくは、ノード t を分割しようとした際に、子ノードに含まれるデータ数が下限値を下回る場合とし、その場合は、ノード t

を葉ノード（終端ノード）としてそれ以上分割を行わないとした。下限値の設定によって木の深さを制御することが可能であり、過学習を防ぐことができる。本研究でも、葉ノードのデータ数の下限値を変化させて、最適な木の深さの探索を行った。

木の深さ探索と決定木モデルの構築は、MATLAB®を用いて行った。MinLeafSize（ノードに含まれるデータの下限値）の違いによる決定木モデルの性能を k 分割交差検証によって推定される分類誤差で評価した。ここで、分類誤差は、検証用データにおいて、分類を誤ったデータの割合である。また、k 分割交差検証では、教師データを k 個に分割し、そのうちの一つをテストデータとし、

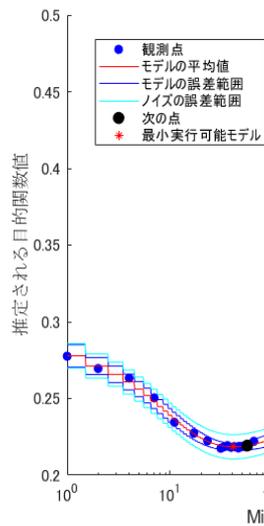


図-4 葉ノードのデータ数を基準とした木の深さ探索

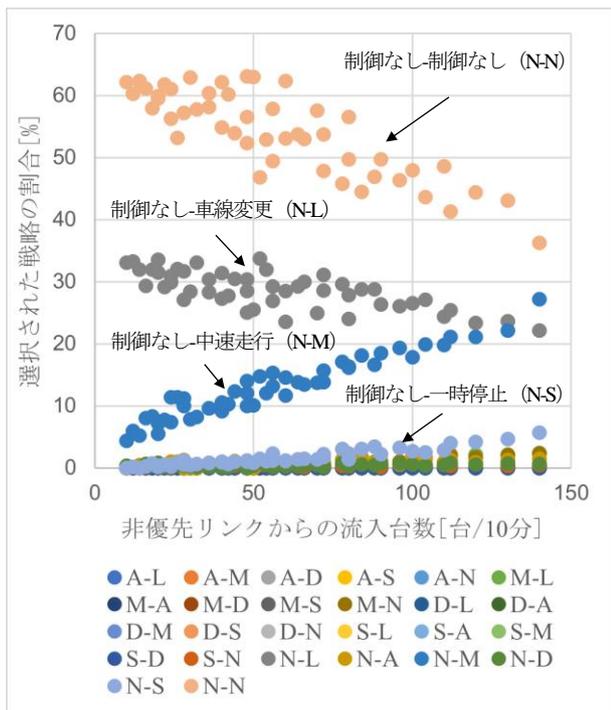


図-5 最適な協調制御方法の選択割合（教師データ）

残りのデータでモデルを構築し検証を行う。テストデータを変えつつ k 回検証を行い結果の平均をとることで、モデルの分類誤差の推定値が得られる。本研究では、k = 5 として深さ探索を行ったところ、図-4 に示すように MinLeafSize=39 で分類誤差が 21.9%と最も小さくなると推定された。

a) 教師データの協調制御の分析と学習結果

教師データに含まれる最適な協調制御方法と、そのデータを用いて学習した決定木から出力される協調制御方法を比較する。また、優先側発生車両台数 x [台/10分]と非優先リンクへの流出割合（非優先からの流入割合）y[%]から非優先リンクからの流入台数[台/10分]を算出し、50の交通量パターンをこれを用いて整理した。

図-5は、教師データにおける最適制御方法の割合を、非優先リンクからの流入台数でグラフ化したものである。

この図より、非優先側の交通量によって、最適な制御方法が変換することが分かる。交通量が少ない状況では制御なし-制御なし (N-N)（協調制御を行わず、優先・非優先関係のローカルルールに従って各車両に行動させる）や、制御なし-車線変更 (N-L) のように車両が減速を行わずコンフリクトを解消する制御方法が最適となることが多い。非優先側からの交通量が増えると、制御なし-中速走行 (N-M) や制御なし-一時停止 (N-S) の制御が最適とされる割合が増える。これは、非優先側の交通量が多いと非優先リンクに滞留が生じるため、優先リンクの車両が減速や停止をし、非優先側の車両を先に交差点に進入させることで、全体の旅行時間を小さくできるためと考えられる。また、非優先側からの交通量が多いほど、制御なし-制御なし (N-N) が選ばれる割合が減ることから、混雑する状況ほど協調制御を行う必要性が高くなることもわかる。

決定木学習では各葉ノードに分類された教師データの内、ノードに存在する制御方法の中で最も割合が高い制御方法でそのノードを代表する（代表クラス）。表-5は、構築した最適な深さの決定木の葉ノードの代表クラスとそのクラスの分類されたデータにおける割合を示した。

表-5 葉ノードの代表クラスとその割合

協調制御方法 (代表クラス)	葉ノード の数	代表クラス が占める割合 (平均値)
制御なし-制御なし (N-N)	120	0.66
制御なし-車線変更 (N-L)	55	0.57
制御なし-中速走行 (N-M)	50	0.66
中速走行-制御なし (M-N)	1	0.56
制御なし-高速走行 (N-A)	1	0.51
計	227	-

葉ノードの代表クラスがデータ内に占める平均的な割合は、いずれも 5~6 割程度であった。つまり、決定木モデルを用いて予測した制御方法が最適である割合が 6 割程度しかないと言える。

(2) 決定木による協調制御の検証

決定木モデルによる協調制御の効果を検討するため、教師データと同様の交通量パターンに対して、異なるランダムシードを用いてそれぞれ 50 の車両発生シナリオをシミュレートし、検出したコンフリクトに対して決定木を用いて協調制御を行うことで、その効果を検証した。なお、この決定木を用いた制御シミュレーションにおいては、1 つのコンフリクトを制御している間であっても、次のコンフリクトが見つければそれを制御することができる。

a) 決定木モデルによる最適な制御方法の選択

決定木を用いたコンフリクト制御シミュレーションにおいて各制御方法が選択された割合を図-6 に示す。

図より、教師データで示された最適な制御方法が概ね再現できていることが分かる。教師データでは、ある車両発生シナリオの 1 回のシミュレーションからランダムに選ばれた 1 つのコンフリクトを対象としていたが、決定木による制御シミュレーションにおいては検出された全てのコンフリクトを制御している。このため、必ずしも選択された制御方法の割合を正確に比較できない。また、決定木による制御シミュレーションにおいては既に制御中となっている車両など、学習データには含まれない状

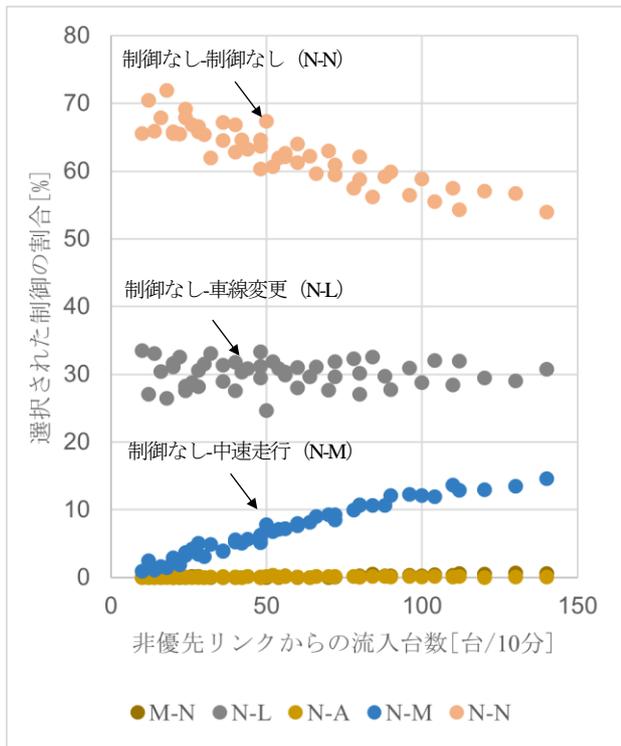


図-6 最適な協調制御方法の選択割合 (決定木)

況で決定木を適用したため、交通状況が異なる可能性がある。しかしながら、制御なし-制御なし (N-N) が選択された割合の平均値が、教師データでは 54.2%であったのに対して、決定木モデルによる出力結果では 62.6%となっており、制御を行わない割合が期待していたよりも高くなった。また、制御なし-高速走行 (N-A) や中速走行-制御なし (M-N) の制御がほとんど選ばれておらず、教師データの段階であまり選択されない制御方法には十分に対応できていない可能性がある。

b) 旅行時間削減効果

決定木を用いたコンフリクト制御シミュレーションの結果から、協調制御による旅行時間削減効果を以下の式で計算した。

$$R_a = \frac{1}{50} \sum_{i=50} \left(\frac{T_{0i} - T_{1i}}{n_a} \right) \quad (2)$$

ここで、 a は交通量パターン、 i はシナリオ番号 (1~50)、 n_a は交通量パターン a における発生車両台数、 T_{0i} [秒] はシナリオ i において協調制御を行わない場合の総旅行時間、 T_{1i} [秒] は決定木で選択された協調制御を適用した場合の総旅行時間であり、 R_a [秒/台] は各交通量パターンにおける平均旅行時間削減量である。

図-7 に旅行時間削減量を優先側発生車両台数と非優先への流出割合で整理した結果を示す。この際、非優先リンクへの流出台数は、非優先リンクからの流入台数と等

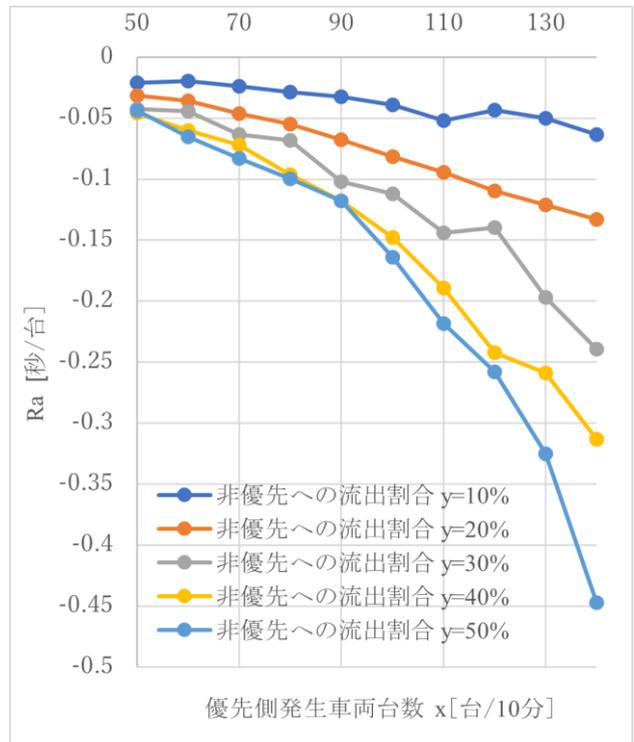


図-7 旅行時間削減効果

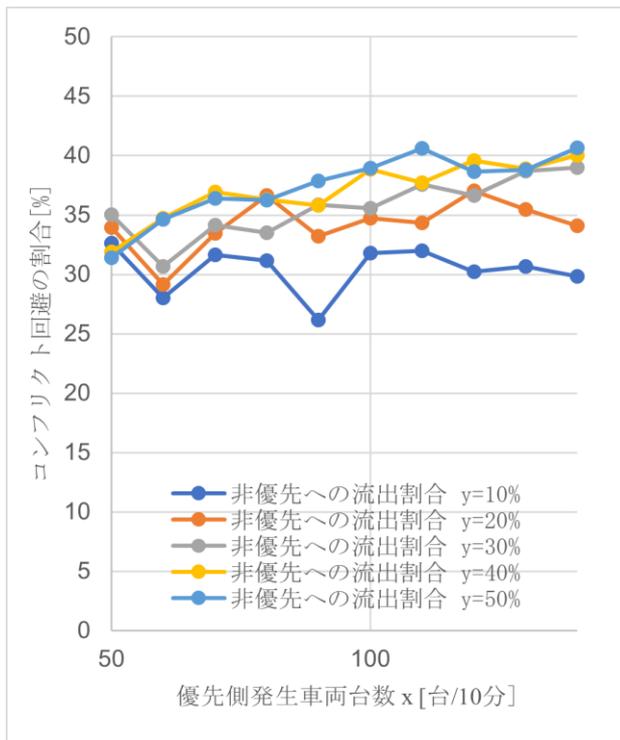


図-8 コンフリクトの回避効果

しいことに注意されたい。図より、発生台数が多く、非優先道路との流出割合が高い状況ほど、協調制御による旅行時間削減効果が高くなることが分かる。つまり交通量が多く混雑した状況ほど、コンフリクトの解消が周辺交通流に与える影響が大きく、協調制御の効果も高くなるといえる。

d) コンフリクトの回避の効果

図-8に決定木による協調制御によってコンフリクトが回避できた割合を示す。ここで、コンフリクトの回避は交差点進入を妨げられる側の車両が制御をしない場合の交差点進入時刻よりも早く交差点に進入できた場合としている。この結果から、協調制御により、約 25~40%のコンフリクトを回避できることが分かった。しかし、半数以上のコンフリクトは回避することが出来ておらず、これには制御なし-制御なし (N-N) の選択割合が高いことが影響していると考えられる。

また、本研究の決定木モデルは、制御対象が常に 2 台のみで他車両は必ず通常のローカルルールに従い走行する環境下で学習データを作成している。一方で、決定木による制御シミュレーションでは、複数のコンフリクトを同時に制御しているため、一つのコンフリクトの制御中に、それより前に発生が予測されたコンフリクトに関連して既に制御されている車両が存在している。このため、学習データにおける交通状況と、決定木を適用した際の交通状況が必ずしも一致していないことも、精度低下を招いている可能性がある。

5. 結論と今後の課題

本研究では、無信号 T 字交差点におけるコンフリクト回避のための協調制御を決定木により学習し、シミュレーションによりその効果を検証した。また、様々な交通量パターンで学習データを作成し、構築したモデルの精度検証協調制御の効果の分析を行った。

分析の結果から、交通量の少ない非優先道路から交差点に進入する車両台数によって、最適な協調制御方法が変化することが示された。また、学習した決定木モデルは 8 割程度の分類精度を有し、決定木によって協調制御を行った結果、車両の旅行時間を減少させることが可能であることが分かった。一方で、検出したコンフリクトのうち、回避できたコンフリクトの割合が高くとも 4 割程度しかなく、半数以上の状況で協調制御を行うよりも、優先・非優先関係のローカルルールに従って走行することが最適とされた。これは、決定木のカテゴリ精度が十分でないためと考えられ、学習精度の向上や、制御方法の定義の見直しが必要である。

学習精度に関しては、今回用いた T 字のネットワークで 2 割のカテゴリ誤差が発生しているため、十字交差点などのより車両のコンフリクト関係が増える複雑な交通流ではさらに予測精度が低くなると考えられる。そのため、決定木学習の精度向上は大きな課題であり、今回得た知見を参考に、教師データの取得方法や、説明変数の設定についても見直しを行う必要がある。また、ランダムフォレストなどのアンサンブル学習や他の機械学習手法の適用も検討する必要がある。さらに、本研究では全ての車両が自動運転車であると仮定しているが、制御できない一般車の存在を考慮した分析も必要である。

謝辞：本研究は JSPS 科研費 19H02260、および JST の研究成果展開事業「センター・オブ・イノベーション (COI) プログラム」の助成を受けたものです。ここに記して謝意を表します。

参考文献

- 1) Meng, Y., Li, L., Wang, F., Li, K. and Li, Z.: Analysis of Cooperative Driving Strategies for Nonsignalized Intersections, IEEE Transactions on Vehicular Technology, Vol. 67(4), pp. 2900-2911, 2018.
- 2) Dresner, K. and Stone, P.: Multiagent traffic management: a reservation-based intersection control mechanism, Proc. AAMAS, New York, NY, USA, pp. 530-537, 2004.
- 3) Li, L. and Wang, F.: Cooperative driving at blind crossings using intervehicle communication, IEEE Transactions on Vehicular Technology, vol. 55, no. 6, pp. 1712-1724, 2006.
- 4) Lee, J. and Park, B.: Development and evaluation of a cooperative vehicle intersection control algorithm under the

- connected vehicles environment, IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 1, pp. 81-90, 2012.
- 5) Kamal, M.A.S., Imura, J., Hayakawa, T., Ohata, A. and Aihara, K.: A vehicle-intersection coordination scheme for smooth flows of traffic without using traffic lights, IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 3, pp. 1136-1147, 2015.
- 6) 田代むつみ, 野崎泰誠, 三輪富生, 森川高行: 無信号交
 差点における最適合流制御方法の検討, 土木計画学
 研究・講演集, Vol.59, CD-ROM, 2019.
- 7) Nagel, K. and Schreckenberg, M.: A Cellular automaton model for freeway traffic, Journal de Physique I, France, vol. 2, no. 12, pp. 2221-2229, 1992.
- 8) MathWorks HP, https://jp.mathworks.com/?s_tid=gn_logo (アクセス日: 2021年9月28日)