

## 1 F - 2

知識表現システムFracks/TPと  
交通信号制御への適用について杉本 勉, 野中 哲, 大濱寛樹  
(NTTデータ通信株式会社)

## 1 はじめに

将来の交通信号制御として予測制御が期待されている。予測制御は、道路網の挙動推定(交通挙動推定)をする機能とそれから制御計画をたてる機能の2つに分けることができる。そこで我々は、交通挙動推定への知識処理技術の適用について検討を行っている。

現在は、交通モデル構築のための第一歩である交差点の信号制御を実現する表現方法について検討しており、本稿ではその方法について主に述べる。また、モデル化には我々が並行して開発している知識表現システムFracks/TPを用いている。

## 2. 知識表現システムFracks/TP

## 2.1 概要

知識表現システムFracks/TP(FRAME and Constraint based Knowledge representation System for Traffic Prediction)は大きく以下の3つの特徴を持つ。

- (1) フレーム理論に基づいている。フレームもスロットもすべて同じ構造を持つので、対象フレーム内の各スロット自身もフレーム(スロット・フレームと呼ぶ)として扱うことができる。
- (2) 制約概念を導入し、フレーム間の関係記述を強化している。Fracks/TPにおいて制約とは複数フレームのスロット間の関係(等しい(==), 等しくない(!=))を記述するものである。
- (3) 階層関係として「クラス-インスタンス」関係と「全体-部分」関係の2種類を提供する。制約は「全体-部分」関係の「全体」フレームに記述する。

Fracks/TPでは交通工学で使われている方法に沿って信号制御ができるよう、そこで要求される機能は次の2つである。

1. 基本フレーム表現(「フレーム-スロット-値」の3項表現)とその操作述語 `get_value` と `add_value`。
2. 「あるフレームのスロット値とあるフレームのスロット値が等しい(==)」という制約の記述、および `get_value` や `add_value` で利用できる検索/更新機能。

2.2 と 2.3 では、フレーム操作述語である

Knowledge Representation System - Fracks/TP -  
and its Application to Traffic Lights Control  
Tsutomu SUGIMOTO, Satoru NONAKA, Hiroki OOHAMA  
NTT Data Communications Systems Corp.

`get_value` と `add_value` への制約概念導入方法と導入による利点についてそれぞれ述べる。

2.2 `get_value` と制約

通常、目的の値が直接得られないとデフォルト値を参照したり、デーモンを起動している。制約指向概念を導入し、制約伝播を利用することによりデーモンの記述は必要なくなる。しかし、Fracks/TPでは処理速度や記述力の点からデフォルトやデーモンの記述も残り、その上に制約指向概念を導入している。

2.3 `add_value` と制約

`add_value` の場合、値の変更が伴うため整合性チェックなどデーモンの役割は重要であり、複雑になる。しかし、制約記述は `get_value` にも `add_value` にも利用できるため記述量を低減させることができる。また、システム側が整合性のチェックだけではなく整合性を保つための他の値の変更機能まで持っているため、基本的にユーザは制約記述のみでよい。

## 3. 交差点信号制御への適用

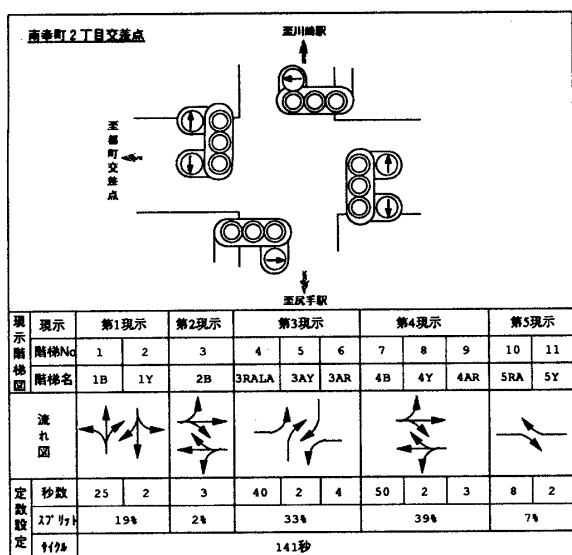
## 3.1 従来方法の問題点

[1]では、各信号間の制約として「等しい(==)」と「等しくない(!=)」を利用し、交差点の中の一つの信号機の状態を変え、制約伝播によって信号制御を実現した。しかし、この方法では次の2つの短所がある。

- (1) 現在の交通工学で行われている方法が使えない。すなわち、交通工学ではいくつかの信号制御パラメータ(表1)と交差点の説明図(図1)[2]を使っているが、[1]の方法ではこれらを利用していない。
- (2) 信号間の制約記述だけでは、通常の4枝交差点や簡単な時差式信号を含んだ交差点しか表現することができない。

表1 信号制御パラメータ

現示 通行権が与えられている交通流または同時に通行権が与えられている交通流の一群をいう。通常の4枝交差点では2現示制御と呼ばれる。また、同一現示内における各信号機の変化に着目した分類を「階梯」という。
サイクル 青, 黄, 赤信号またはこれに変わる一連の信号表示が一巡するのに要する時間のこと。
スプリット 1サイクル中で各現示に割り当てられる時間をこと。これには黄信号も含めることがある。



記号の意味

記号	B	RALA	RA	Y	AY	AR
意味	青	---	-	黄	全黄	全赤

図1 南幸町2丁目交差点の現示階梯

3.2 適用例

前節に記した問題点(1),(2)に対して、次の2つの改善案をそれぞれに適用した。

- ・信号制御パラメータ「現示」と「階梯」に注目。
- ・実際にある交差点の信号制御の実現。

本節ではそれらの方法について述べる。

実際にある交差点の例としてJR川崎駅西口近くにある南幸町2丁目交差点を選んだ。この交差点はすべての信号に青矢印の点燈があり(内2つは左右の青矢印がある)、5現示制御というかなり複雑な動きをする交差点である。従ってこの交差点の信号制御を実現できればこの表現方法は汎用性があるといえる。

信号制御を実現するためのフレームとして「南幸町2丁目交差点」と各信号機4つ、それにそれぞれのクラスに対応するフレームを考える。また、「南幸町2丁目交差点」フレームと各信号機フレームは「全体部分」関係である。「南幸町2丁目交差点」フレームには「現在の階梯番号」や「現示」といったスロットが、各信号機には「(ある時点の信号機の)色」スロットがある。

この交差点の信号機は互いに向かい合っているもの同士同じ形で同じ動きをする信号機であるため、次の2つの制約を与えることができる。

- 制約1: 信号機1@色:==信号機3@色
- 制約2: 信号機2@色:==信号機4@色

この制約記述は、get\_valueの時は値の検索パスとして、add\_valueの時は整合性を保つための更新パスとして働く。

例えば、get\_valueである信号機の色を取得する場合、信号機の色は全ての信号機に記述がある必要

はなく、:=で結ばれたもののどちらかにあればよい。自分自身に記述がなければ、制約を解析して検索パスをたどって値を取得する。

add\_valueを利用した、交差点の階梯の更新は、「信号機1」と「信号機2」の更新(図3ウ add\_values)が明示的に行われ(図2ウ-1,ウ-2)、「信号機1」には「制約1」が「信号機2」には「制約2」が起動して制約伝播が起こりそれぞれ「信号機3」と「信号機4」の「色」が更新(add\_value)される(図2ウ-3,ウ-4)、ことにより実現できる。

この方法により、「現示」や「階梯」の情報を利用し、現示数などの複雑さとは関係なく、信号制御を実現することが可能となる。

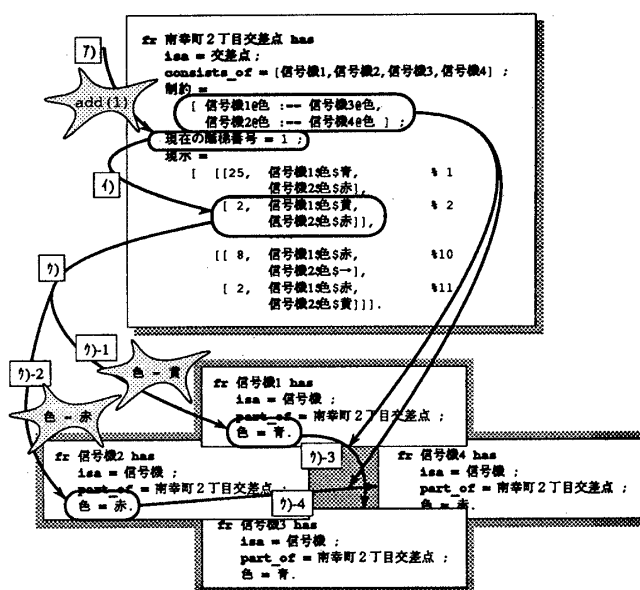


図2 フレーム構造およびデータの流れ

```

next kaitei(Inter,Next) :-
  ( add_value(Inter, 現在の階梯番号, add(1)); ... 7)-1
  add_value(Inter, 現在の階梯番号, 1) ); ... 7)-2
  get_value(Inter, 現在の階梯, Next); ... 4)
  add_values(Next). ... 7)
    
```

図3 次の階梯を求めるプログラム

4. まとめ

知識表現システムFracks/TPの概要とその交通信号制御への適用について述べた。現在3~5ヶ所程度の交差点を対象に交通挙動推定のためのモデルを構築している。

参考文献

- [1] 杉本, 制約概念を導入した知識表現言語, 情報処理学会第41回全国大会, 1990.
- [2] (社)交通工学研究会編, 交通信号の制御技術, 1989.