

実時間意志決定機構の検討

1C-4

河野 隆志 加藤 秀一

NTTヒューマンインタフェース研究所

1. まえがき

人間の問題解決能力は非常に強力で、それをコンピュータに代行させようとする、様々な困難な問題に出会う。例えば、人間は困難な状態に遭遇した場合、とりあえずその状態を避ける行動をとりながら、その間により効果的な行動を考えたりする。このような実時間の問題解決能力を持つことは、実世界の環境で人間の行なっていることを補佐または代行するシステムを作る上で本質的な問題になっている。本稿の目的はそのための手法を示すことである。

2. 問題の設定

実時間で問題解決することが必要なとき、考慮しなければならない環境条件は様々だが、特に次のことが重要である。

[環境条件]

- (1) 許容時間内に解を出す必要がある。
- (2) 複数の対処すべき重要な事態が発生する。
- (3) 入力データが時々刻々、非同期に変わる。
- (4) 問題解決に必要な情報がすべて揃う保証はない。

[問題設定]

まず手始めとして、実時間で、複数の対処すべき状態を認識し、それに対応した行為を許容時間内に行なうことを目的とする問題を念頭におく。

入力情報を、系を構成する各要素の入力 s_i ($1 \leq i \leq n$)の集合で表すことにすると、系の状態は $S = \{s_1, s_2, \dots, s_n\}$ で表される。この系の状態 S の部分集合 $S_i = \{s_{i1}, \dots, s_{ik}\}$ を部分状態と呼ぶ。(ここで、各部分の状態 S_i はそのとり方に特に制約はなく、オーバーラップもあり得るとする。)

ここでは、部分状態 S_i に対する条件判定関数 C_i を定め、起動条件 $C_i(S_i)$ と、その結果としての行為 A_i とからなるルール R_i を導入する。すなわち、ルール R_i を

$$\langle C_i(S_i) \rightarrow A_i \rangle$$

で表す。このルール R_i の集合 R を知識ベースとする実時間意志決定機構を検討する。

3. 手法の検討

2で示した環境条件の中で(1)に対応する手法を示す。

許容時間は問題解決に許された時間である。問題解決にかかる時間の最大値は許容時間以下でなければならない。このための戦略を次に示す。

- 1) 問題解決にかかる時間をできるだけ小さくする。
- 2) 許容時間内で、最適ではないかも知れないが適当な解を見つける。そのためには、許容時間、問題解決時間が見積れることや途中結果が有効な問題解決方式が必要である。

従来は1)の研究が主に行なわれてきたが、実時間の問題解決の本質は2)であるので、本稿では2)に重点を置く。

[手法]

想定システムの構成を図1に示す。(1)~(3)の各部分に対する手法を以下に述べる。

- (1) 重要な事態への優先対応

重要な事態や許容時間の短い事態へは優先的に対処する。

そのために重要度 I_i 、緊急度 E_i という項目を用意する。これらの項目はルール毎に与える。すなわち、ルール R_i を

$$\langle C_i(S_i) \rightarrow A_i, I_i, E_i \rangle$$

で表す。

重要度 I_i は時間の推移とは関係のない事態へ対処することの利害の程度を表す値で、ルール R_i に固有の値である。

緊急度 E_i は事態への対処の時間的緊急性を表す値である。

更に、緊急度 E_i を正確な緊急度 U_i ととりあえずの緊急度 V_i に分ける。

正確な緊急度 U_i は壊滅状態に陥る時間の逆数に比例する値で、 S_i の部分集合 S_{iu} を引数とする関数 $u_i(S_{iu})$ として与える。

とりあえずの緊急度 V_i は正確な緊急度 U_i の近似値で、 $S_{i,u}$ の部分集合 $S_{i,v}$ を引数とする関数 $v_i(S_{i,v})$ として与える。すなわち、緊急度 E_i を

$$\langle u_i(S_{i,u}), v_i(S_{i,v}) \rangle$$

で表す。

これらの項目の使用戦略の一例を示す。複数のルールが候補に上がった場合、次の①から④を順番に実行してルールを絞り、候補が1つになるまで続ける。

- ① 重要度があるしきい値より高いルール群を候補とする。
- ② とりあえずの緊急度をルール群の中で比較して、最も高いルール群を候補とする。
- ③ 緊急度をルール群の中で比較して、最も高いルール群を候補とする。
- ④ 重要度をルール群の中で比較して、最も高いルール群を候補とする。

(④で一意に決まらない場合、ルール群の初めの要素を選ぶ)

(2) とりあえずの行為と行為の最適化

ルール R_i の結果部である行為 A_i は部分状態により決定される。このとき、最も適当な行為をしていく上で必要な部分状態を S_j とする。部分状態 S_j を $S_{j,n}$ と $S_{j,p}$ に分割する。ここで、 $S_{j,n}$ はとりあえずの行為を決定するための部分状態であり、 $S_{j,p}$ は $S_{j,n}$ とともに用いて適当な行為を決定し、その行為を行なうために重要な順に列べた順序集合である。

部分状態から行為を決定する関数 $a_i(S_a, S_b, n)$ ($0 \leq n \leq N$, $N: S_b$ の要素数)は S_b の第 n 要素までとり、残りをデフォルト値として計算する。このように関数 a_i を定義すると、行為 A_i は

$$\{a_i(S_{j,n}, S_{j,p}, n) \mid 0 \leq n \leq N\}$$

で表される。

とりあえずの行為はとりあえずの緊急度 V_i があるしきい値 V_0 より大きい場合に、 $n=0$ とし、 $a_i(S_{j,n}, S_{j,p}, 0)$ を実行して行なわれる。

適当な行為は $a_i(S_{j,n}, S_{j,p}, n)$ を $n=1$ から $n=N$ ($N: S_{j,p}$ の要素数)まで随時、実行して行なわれ、最も適当な行為に近づいていく。

とりあえずの緊急度 V_i がしきい値 V_0 以下の場合には、最も適当な行為 $a_i(S_{j,n}, S_{j,p}, N)$ を行なう。

(3) 状態を認識する時間の短縮

状態の認識は重要度 I_i の大きいルールから順にその起動条件 $C_i(S_i)$ を調べ、成立していればそのルールを選択ルールの候補とする。

状態を認識する時間を短縮するため重要な状態は並列的に認識する。ここで、重要な状態とはあるしきい値 I_s より大きい重要度 I_i をもつルール群 $R_s\{\dots, R_i, \dots\}$ の部分状態群 $\{\dots, S_i, \dots\}$ である。しかし、これに当てはまる全ての入力を並列に認識することは不可能なので、ルール R_i の起動条件 $C_i(S_i)=c_{i1}(s_{i1}) \wedge \dots \wedge c_{in}(s_{in}) \wedge \dots \wedge c_{im}(s_{im})$ の中で、ルール群 R_s 中の他のルールに共有されることのも最少ない入力 s_{ij} を引数として持つ条件判定 $c_{in}(s_{ij})$ を並列に実行する。そして、この条件判定が成立した時点で、起動条件 $C_i(S_i)$ が成立しているか調べに行き、成立していればそのルール R_i を選択ルールの候補とする。

4. むすび

実時間問題解決方式の実現への手始めとして、複数の対処すべき状態を認識し、それに対応した行為を許容時間内に行なうための一手法を示した。

今後、重要度、緊急度の決定方法や途中結果が有効な(最適な解に向かって連続的に解を出し続ける)問題解決方式の検討とともに、本手法の有効性を試作システムの作成により検証していく。

<参考文献>

- (1) T.J.Laffey, P.A.Cox, et al. : Real-Time Knowledge-Based Systems, AI MAGAZINE, SPRING, pp27-45, (1988).
- (2) 加藤：リアルタイムエキスパートシェルの基本検討, 信学秋全大, D-147, (1989).

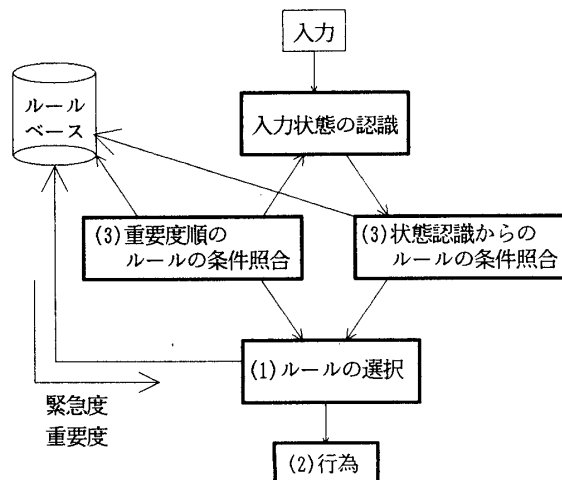


図1：想定システムの構成