

オートマタ表現によるタスク系列生成に関する研究

5 S-3 — 複数のタスク集合のクラスタリング —

成瀬 継太郎 嘉数 侑昇
北海道大学 工学部

1. はじめに

本研究では、変動する環境におけるタスク系列生成問題を取り扱う。ここで扱うタスクとは、タスクを処理するとそのタスク固有の利得が得られるが、その利得はタスクを処理してはじめて解るものであり、このようなタスクを継続的に処理し、得られる利得をを最大にするタスクを選択する問題をタスク系列生成問題とする。この利得はある固定値を取り続けるのではなく、時刻によって変動し、いつ利得が変化するかは明示的にはあたえられない。これを環境の変化としてとらえる。環境が変化すると同じタスクを処理しても得られる利得が異なるので、環境に応じた選択方法が必要となる。

このような問題に対して、タスクの選択問題を確率オートマトンで処理することを目的とする。まず、環境の変化に対応するために、環境の変化のクラスタリングを行う。そして、各クラスタとオートマトンの内部状態を対応させ、各内部状態に対する出力関数を学習によって更新する。また、その有用性を簡単な計算機実験によって確かめる。

2. タスク集合

ここで扱うタスクとは実行可能な仕事の最小単位であり、タスク間の制約条件は、タスク間の半順序として表現する。

【定義 2. 1】

全てのタスクからなる半順序集合を $(T = \{t_i | i \in I\}, \preceq)$ と表す。

ここで、 t_i はタスク、 I は添字集合である。

【定義 2. 2】

タスク集合 T におけるタスクの列 $p = (t_{i_0}, \dots, t_{i_n})$ ($n \geq 0$) に対して、 $t_{i_0} \in T_{in}$, $t_{i_n} \in T_{out}$, $t_{i_j} \preceq t_{i_{j+1}}$

($j = 1, \dots, n-1$) が成り立つときこの列 p をタスク集合のパスと呼び、すべてのパスからなる集合をパス集合と呼び、 P で表す。ここで、 T_{out} , T_{in} はそれぞれ (T, \preceq) の極大元・極小元からなる集合である。

【定義 2. 3】

関数 $f: P \rightarrow \mathcal{R}$ をパス集合 P における評価関数と呼ぶ。すなわち、変動する環境は異なる評価関数として表現され、タスクの選択は (T, \preceq) の中から $f(p)$ を最大にする p を選択する解探索問題になる。

3. オートマタの学習

3. 1 オートマタネットワーク

タスク集合をオートマタネットワークとして表現することを考える。タスク $t_i \in T$ に関するオートマトンを A_i とし、その出力関数を G_i とする。

【定義 3. 1】 オートマタネットワーク AN は以下の 2 項組である

$$AN = (A, E) \quad (1)$$

$$A = \{A_i | i \in I\} \quad (2)$$

$$E = \{(A_i, A_j) | t_j \preceq t_i, A_i, A_j \in A\} \quad (3)$$

AN におけるタスクの選択は、 A_i が結合している A_1^1, A_1^2, \dots の中からある一つのオートマトンを選択することである。

オートマタネットワークに対して環境 Env を考える。ネットワークへの入力 T_{in} と出力 T_{out} は、 Env を通じて行われる。また、パスの評価も Env によって行われる。

3. 2 オートマタネットワークの学習

ここでの学習 [NR91][NA89] の目標は、最適なパスをただ一つ選択するのではなく、評価値の期待値を最大にするように各々のオートマトンの出力確率を定めることである。ここで考える出力関数は、確率分布の形をとるものとする。 A_i の出力関数 G_i として、 A_i から A_1^1, A_1^2, \dots に結合しているとき、それぞれを選択する確率を p_1^1, p_1^2, \dots として、 $G_i = \mathbf{p}_i = (p_1^1, p_1^2, \dots)$ である。この \mathbf{p}_i を A_i の選択確率ベクトルと呼ぶ。

まず、各オートマトン A_i にある時刻 t におけるシステムに対する有効性を表す強度 [HO89] というパラメータを

割り当て、 Str_i と記す。

タスクの処理は、これらのオートマトンに対して、確率的に一つを選択し、選択されたオートマトンに強度に利得が付加されると表現する。すなわち、

$$Str_i(t) = \begin{cases} Str_i(t) + f(A_i), & A_i \text{ が処理されたとき} \\ Str_i(t), & \text{それ以外のとき} \end{cases} \quad (4)$$

ここで、 $f(A_i)$ は t_i を処理することによって得られる利得である。

このとき、 $G_i = p_i$ を次の式で更新する。

$$p_i^j(t) = \frac{Str_i^j(t)}{\sum_k Str_i^k(t)} \quad (5)$$

すなわち、選択確率は強度に関して比例配分される。

また、各オートマトンの初期強度はすべて等しい、つまり選択確率はすべて等しいとする。

4. タスク集合のクラスタリング

環境の変化を認識するためにここでは、利得の変化を異なるタスク集合ととらえ、選択確率ベクトルとその時間変化に基づくタスク集合のクラスタリングを行う。そして、各クラスタでの望ましい選択確率ベクトルを決定する。

ある時刻 t での選択確率ベクトルを $p(t)$ 、その時間変化を $\dot{p}(t) = p(t) - p(t-T)$ とする。また、 i 番目のクラスタのテンプレートを C_i, \dot{C}_i とする。ここで、 C_i は確率ベクトルに関するテンプレート、 \dot{C}_i は確率ベクトルの変化に関するテンプレートである。このとき、最近隣法に基づくクラスタリングを行う。まず、既存のクラスタに最も近いクラスタ c を次の式で判定する。

$$(C_c, \dot{C}_c) = \min_i (d((C_i, \dot{C}_i), (p(t), \dot{p}(t)))) \quad (6)$$

ここで $d(x, y)$ は距離関数である。

そして、 d_i をしきい値として $d((C_c, \dot{C}_c), (p(t), \dot{p}(t))) < d_i$ のときクラスタ c に分類され、それ以外のときは $(p(t), \dot{p}(t))$ をテンプレートとする新しいクラスタを生成する。

次に各クラスタに望ましい選択確率ベクトルを決定する。これには様々な方法が考えられるが、ここでは \dot{C}_i を正規化したものをクラスタ i の望ましい選択確率ベクトルとする。

5. 計算機実験

3種類のタスクの選択問題に対して、実験を行う。環境の変動は、100単位時間ごとに利得ベクトル (1.0, 2.0, 3.0) (3.0, 2.0, 1.0) を交互に繰り返し、初期強度をそれぞれ 10.0 とし、 $d_i = 0.15 T = 10$ とする。

図1に学習とクラスタリングの結果を示す。それぞれのタスクの選択確率を黒線で、生成されるクラスタに番号を割り付け、クラスタの推移を灰線で表す。環境の変動に対して選択確率が変化し、選択確率とその変化量に基づきクラスタが生成されている。

クラスタリングの評価として期待利得を考える。図2に期待利得の推移を示す。実線はクラスタリングの結果を用いない方法、破線はクラスタリング結果の理想確率ベクトルを用いた方法を示す。クラスタリングの結果を用いない方法では、環境が変化するたびに期待利得の値が減少し、そこから緩やかに上昇するが、クラスタリングの結果を用いる方法では、環境の変化による期待利得の減少期間は短く、そこからの上昇も鋭い。

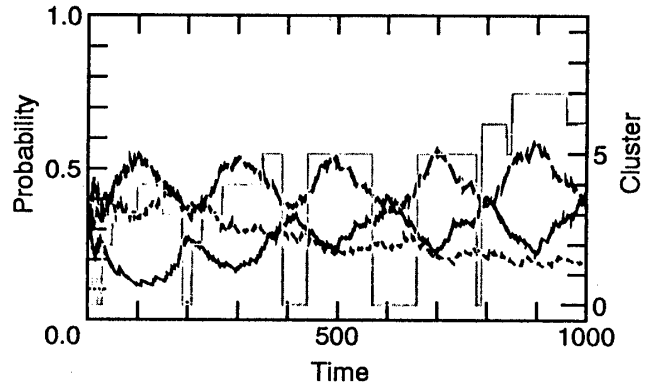


図1 選択確率とクラスタリング

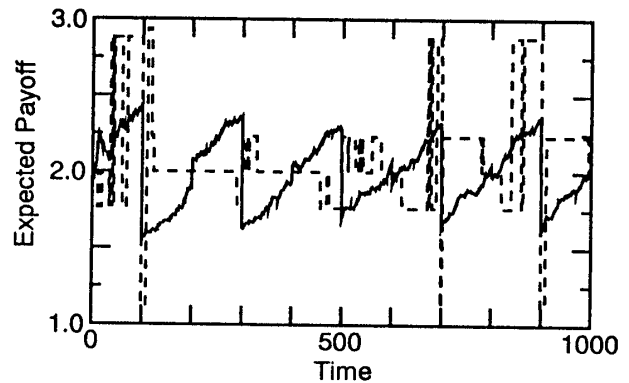


図2 期待利得

6. おわりに

変動する環境におけるタスク系列生成問題に対して、タスクの選択問題を確率オートマトンで処理し、その選択問題を出力関数の更新による学習によって行った。このとき、確率変化に基づく利得の変化のクラスタリングを行い、クラスタとオートマトンの内部状態を対応させ、環境の変化に応じて、異なる出力関数を実現した。また、その有用性を簡単な計算機実験によって確かめた。

参考文献

- [NR91] 成瀬, 嘉数: オートマタ表現によるタスク系列生成に関する基礎研究, 第43回情報処理学会全国大会講演論文集.
- [NA89] Narendra, K.S., and Thathachar, M.A.L.: *Learning Automata - An Introduction*, Prentice Hall, 1991.
- [HO89] Holland, A.H. et al: *Induction*, MIT Press, 1989.