Position Paper for Macc97.

信頼度に基づく要求詳細化マルチエージェントモデル

有馬 淳 (Jun Arima)
(株) 富士通研究所 (Fujitsu Laboratories LTD.)

Abstract: This paper is concerned with a multi-agent model for general problem solving in computer network characterized by *openness*, *locality*, and *diversity* of available knowledge. We are devising a robust multi-agent system in which, given a generally abstract plan, the agent repeatedly refine it into a more concrete one until executable. In this process, to make up for the lack of knowledge, the agent can ask other agents how to refine the partial plan, according to the social evaluation about who has refined the concerning plan most successfully.
In this general system, this paper focuses on a *credit assignment problem*(Minskey, 1961): If a task is fulfilled by a sequence of activities of agents, how can credit or blame be accurately assigned to early rules that set the stage for the final result? To this problem, we introduce a logic-based model. Unlike the previous researches, this model is devised to specify which agents logically contribute in fulfilling a task. A multi-agent system based on this model has been tested with a simple simulator through an electronic purchase scenario.

1 はじめに

本稿では、サービスにおいて多様な質を持つエージェント群を想定し、過去の実績に基づいた信頼度の高いエージェントを選択することを考え、特に**信頼度割り当て問題** (Minsky,1961) に焦点を当て現状を報告する。この問題は強化学習 (Holland, 1986; Grefenstette, 1988) で特に扱われている。強化学習では環境に適合する新しいルール集合の生成と獲得が本来の目的になるが、ここでは、ルール（エージェント）集合はオープンであるものの与えられるものとし、ある環境で成功しやすいルールを選択させることが目的である。本モデルは従来のＡＩで蓄積のある論理型問題解決モデルをベースにしており、既存の強化学習モデルに見られる信頼度割り当てとは異なり、論理的に貢献したエージェントの特定を可能にしたいという動機がある。これによってより適切な信頼度割当と強化学習効率の改善を目指す。

2 想定している問題解決のモデル

エージェントが問題解決の過程で必要に応じ**主問題**を他エージェントに送信し、受信したエージェントはその主問題を**副問題列**に変換可能な場合に、その（主問題および）副問題列を返信する。このコミュニケーションを通じてプランの部分的 / 段階的詳細化を行なうことで問題を解決するタイプのエージェントをここでは**要求詳細化エージェント** (*Demand-Refinment Agent*:DRA) と呼ぶ。

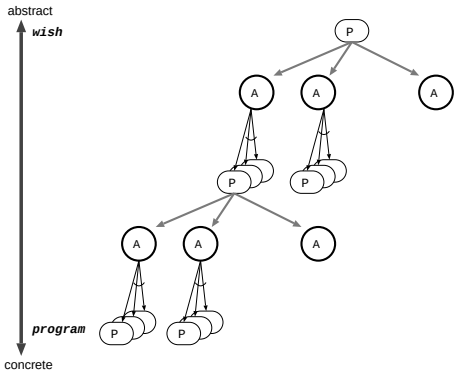

図 1: 要求詳細化エージェントにおける AND-OR 木
[図中] P: 問題、A: エージェント

ここでは、以下のように論理プログラムの記法、概念を借りたモデル化を行なう。主問題を副問題列に分解する知識を**（詳細化）ノウハウ**と呼び $P \longrightarrow P_1, ..., P_n. \quad (n \geq 0)$ で表す。原始式で表された**問題** P' の詳細化依頼文は `how([`P'`])` で表し、問題 P' と P が単一化可能なノウハウ $P \longrightarrow P_1, ..., P_n.$ を受信エージェント A が持つ時に限り、A は `rep([`Q`],[`$Q_1, ..., Q_m$`])` $(m \geq 0)$ を返信することができる。ここで Q は P, P' の単一化 $(Q = P\theta)$ であり、$Q_1, ..., Q_m$ は $(P_1, ...P_n)\theta$ または、それらをさらに詳細化した結果である。詳細化依頼を出した問題に関し、有限時間内に詳細化されない場合は失敗とみなされる。

3 信頼度付与問題の論理的処置

信頼度付与問題に関する要求詳細化型エージェントの最大の強みは、問題詳細化空間が副問題列は AND 接点で、主問題を詳細化可能なエージェント群は OR 接点で結ぶ古典的な AND-OR 証明木になぞらえることができる点である（図 1）(Lesser,1990)。詳細化プロセスは証明過程に他ならない。これによって論理的な考察、制御が可能になる。以下は、信頼度付与問題に関し、成功に貢献したエージェントの効果的な選択を試みる。

i) OR 接点： 詳細化に失敗した探索木は、最終結果に論理的に寄与しないことが明らかなため、成功した証明木上のエージェントにだけ信頼度を増加させるポリシーがとれる。この点、実際の成功に結び付かない試行錯誤中の行動列と実際に貢献した行動列との間の区別がない従来の強化学習モデルと大きく区別される。

ii) AND 接点: AND 接点内にも貢献がないと考えるべきエージェントが入る可能性がある。次のノウハウを考える。

```
buy_min_cost(Item) ⟶
        find_best_DST_shop(Shop,Item),
        ask_user(buy(Item,Shop)), ···
```

最初の副問題に注目する。これを詳細化する方法として図2のノウハウが考えられる。ここでメッセージ Mes の A への送信を send(A,Mes)、A からの受信を receive(A,Mes) で表している。この場合、shop1, shop2, shop3 の各エージェントのうち I に関して最低価格を提示する店エージェント S (minimal([_,S],I,[shop1,shop2,shop3])) の結果を導くのに、各店の返答が実際に使用される。しかし、再安値を示した店だけをこの場合強化したい。図2のノウハウはネットワークの特性からも重要な欠点がある; 1) 店のリストはオープンであるべき（[shop1,...]）。ノウハウとして固定されるべきでない。2）最安値の店が最終的な解に貢献するとは限らない。例えば、次に続くユーザーの判断 (ask_user) でこの店は拒絶されるかも知れない。この場合、他のどれも最安値ではないので、この副問題列が成功することはできない。ロバスト性に問題がある。

このタイプのエージェント選択はコントラクトネットに代表されるようにネットワークにおける問題解決では典型的なものと考えている。そこで、各エージェントの返答を評価順に並べることを基本機能として導入し、その表現を**ステップ**と呼ぶ。ステップ部分でエージェント選択部分は問題 P の詳細化に応じて任意の評価尺度 E でソートされることになる。ステップは

$$A :: P//E$$

```
find_best_DST_shop(S,I) ⟶
        minimal([_,S],I,[shop1,shop2,shop3]).
minimal([Price,Shop],I,[Shop])⟶
        send(Shop,how([estimate(I,_)])),
        receive(Shop,rep([estimate(I,Price)],[])).
minimal(Min,I,[S|L]) ⟶
        minimal([Price,Shop],I,L),
        send(S,how([estimate(I,_)])),
        receive(S,rep([estimate(I,P)],[])),
        (Price ≥ P,Min = [P,S]
        ; Price < P, Min = [Price,Shop]).
```

図 2: 再安値の店を探す (1)

```
find_best_DST_shop(Shop,I,_) ⟶
        Shop::estimate(I,P)//P.
```

図 3: 再安値の店を探す (2)

で表される。ここで A は P を詳細化したエージェントであり、E は一般に A、P を引数に持つ関数である（‘A ::’ と ‘$//E$’ は省略可）。ノウハウにおいて、右辺の任意の問題をステップに替えたものもノウハウと扱う。これを用いると先の最安値の店を求めるノウハウは図 3 のように大幅に簡略化される（E が A を引数に持つのは A の信頼度を加味した順序づけを行なう場合を想定している。例では、信頼度に依存せず、単に評価値の低いものから順にならべることを仮定している）。

要求詳細化エージェントは他エージェントの助力が必要なステップ $A :: P//E$ にあたると、問題 P に関する各エージェントの信頼度をもとに確率的に選択したエージェント群に対し詳細化要求を発し、詳細化の結果得られた P およびその行為者 A に基づく E によって詳細化ステップ列を順序づけし、その順に $A :: P$ を置き換え、さらに詳細化が不十分なステップを詳細化する。このプロセスで置換している詳細化列を提供したエージェント A の情報は、次の候補が求められない限り保持される。 問題がすべて解かれた時（どんなステップも残っていない時）、その詳細化に関わったすべてのエージェントに対し信頼度を上げる。（要求詳細エージェント全体の詳細な記述は別な機会に述べる。）確率的にエージェント群を選ぶのは、知識がオープンであり、また計算資源の有限性から局所的にしか処理し得ないことを考えると妥当な解の一つと思われる。

4 今後の課題

現在、主要部分の実装が終わり、論理部分の動作確認は終了している。今後、具体的な信頼度付与に関する理論的な考察を行い、実験を行なう予定である。