

多データベースシステムに対する 多エージェントシステム

2F-2

矢羽田 千哲 濱田 賢 滝沢 誠
東京電機大学

1 はじめに

グループウェア等の分散型情報システムは、通信網によって結合された自律的なデータベースシステム(DBS)を含む。多種多様なDBSを含む大規模なシステムでは、利用者が、どのようなDBSを含んでいるか、各DBSの状態について十分に知ることは困難である。したがって、利用者かわりに、必要となるDBSが何であるかを明確にし、これを操作する機能が必要となる。本論文では、このような機能を提供するために、協調的データベースシステム(CDBS)[HASE92, YAHA92]を提案する。CDBSとは、通信網で相互結合された複数のエージェントから構成されるシステムである。本論文では、エージェント間での交渉方式について述べる。

本論文の構成は以下の通りである。第2章では、協調的データベースシステムの基本概念について述べる。第3章では、知人について定義する。第4章と第5章では、協調動作を行なうための交渉と学習について述べる。

2 協調的データベースシステム(CDBS)

2.1 システム構成

CDBSは通信網で相互接続された複数のエージェントから構成される[図1]。

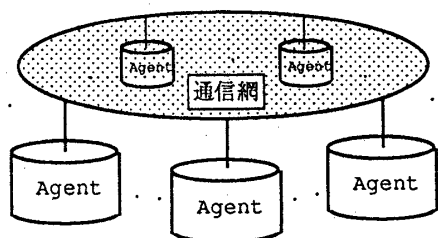


図1: 協調的データベースシステム(CDBS)

エージェントとは、利用者の要求を代行して実行できるシステムである。各エージェントAは、 DB_A とメタデータベース MDB_A を提供する。 DB_A は、オブジェクトの集合である。 MDB_A は、抽象化(is-a)と集約化(part-of)により構造化された項の集合である。項は、Aが持つオブジェクトと、他のエージェントが持つオブジェクトの意味を示す名前である。CDBSでは、エージェント間の協調動作によって、必要なDBSを持つエージェントを見つけ、それに対するデータ操作を行なう。さらに、各DBSが自律性を持つことから、利用者の要求を実行できるか、実行できるとき、どのように実行できるかについて交渉を行なう。

2.2 自律性

CDBSは、自律的なエージェントから構成される。以下についての決定を、エージェント自身が行なえるとき、

Multi-Agent System for Multi-Database System

Chiaki YAHATA, Satoshi HAMADA and Makoto TAKIZAWA
Tokyo Denki University

エージェントは自律的である。

1. DBSのデータ構造。
2. 外部からの操作演算を受け付けるかどうか。受け付けた時の演算の実行順序。
3. 外部からの通信要求に対するの応答方法。
4. 他エージェント内のオブジェクトとの相互関連。

本論文では、各DBSの自律性に対処するために、DBSのエージェント間での交渉手順を考える。これにより、各DBSの動的な変化にも対応し、利用者が求める情報を得ることができる。

3 知人

ここで、A、B、Cはエージェントを示し、tは項を示すとする。 MDB_A が項tを含むとき、Aはtについて知っているとする。エージェントの協調動作は、以下に定義する知人間で行なう。

[定義] MDB_A がtを含むならば、Aはtを知っているとす。また、tについてのオブジェクトを DB_A が含むならば、Aはtを直接知っているとする。Aはtを知っているが、直接には知らない場合、Aはtを間接的に知っているとする。□

Aがtについてのデータ操作要求を受け付けたとき、tを直接知っているならば、 DB_A の操作を行なえる。Aがtを間接的に知っている場合を考える。例えば、他のBの MDB_B がtを含んでいることを、Aが知っている場合には、Aは、tについての操作要求をBに依頼できる。このようなとき、BをAの知人とする。Aの知人でないエージェントを、Aの他人とする。

[定義]Aが「Bはtを知っている」ことを知っているとき、BをAの知人とし、 $A \xrightarrow{t} B$ と書く。 $A \xrightarrow{t} B$ で、 $B \xrightarrow{t} C$ であり、 $B \xrightarrow{t} A$ でなく、 $A \xrightarrow{t} C$ であるとき、AはCを推移的に知るとする。これを、 $A \xrightarrow{t} C$ と書く。 $A \xrightarrow{t} B$ であり、これが推移的でないとき、AはBを直接に知るとし、 $A \xrightarrow{t} B$ と書く。□

AがBをtについて推移的に知っているとき、AはBを直接に問い合わせできず、 $A \xrightarrow{t} C \xrightarrow{t} B$ なるCに、依頼せねばならない。 $A \xrightarrow{t} B$ のとき、AはBにtについて直接問い合わせを行なえる。

4 交渉手順

各エージェントの協調動作は、エージェント間の交渉手続きによって行なわれる。本論文において、DBSに対する操作として、検索演算を考えることにする。

まず、エージェントUは、利用者から検索要求Rを受け取る。Rは、検索条件Qと、プレファランスPから成る($R = (Q, P)$)。Qは、項集合 $\{t_1, \dots, t_m\}$ ($m \geq 1$)である。各項 t_i について、 t_i が示すオブジェクトの集合を $obj(t_i)$ とする。このときRの意味は、オブジェクトの集合 $obj(t_1) \cap \dots \cap obj(t_m)$ とする。Pは検索に対する優先率

項 $(p_1, \dots, p_h) (h \geq 0)$ のリストである。優先事項とは、 Q を満足するオブジェクト集合の検索方法が複数ある場合に、どの方法を選ぶかを与えるものである。 U は、以下の手順により、 R の処理を行なう。

1. R の詳細化を行なう。詳細化とは、 U と利用者の相互作用で、 Q と P を構成していくことである。
2. R を、副要求 R_1, \dots, R_n に分割する。 U は副要求 R の中で、自分で応答を得れない副要求 R_i を、知人に依頼する。
3. 各 R_i に対して知人 A_i を決定する。 U は、 R_i を A_i に送る。
4. A_i に R_i の実行についての交渉を行なう。

以下に U と A_i 間の交渉手順を示す [図 2]。

[交渉手順]

1. U は A_i に R_i を送信する。ここで、 R_i に対して、 A_i が応答可能である条件とプレファランスを、 Q_i' 、 P_i' とする。 $Q_i \supset Q_i'$ は、 A_i は R_i の一部の回答を得れることを示す。 $P_i \supset P_i'$ は、 A_i は P_i の一部の優先事項を持つ場合である。 Q_i と Q_i' 、 P_i と P_i' には、以下の組合せが可能である。

- (a) $Q_i = Q_i'$ 、 $P_i = P_i'$ (b) $Q_i \supset Q_i'$ 、 $P_i = P_i'$
 (c) $Q_i = Q_i'$ 、 $P_i \supset P_i'$ (d) $Q_i = Q_i'$ 、 $P_i \cap P_i' = \phi$
 (e) $Q_i \supset Q_i'$ 、 $P_i \supset P_i'$ (f) $Q_i \supset Q_i'$ 、 $P_i \cap P_i' = \phi$
 (g) $Q_i \cap Q_i' = \phi$

(a) のとき、 A_i は R_i の条件 Q_i を満足する解を P_i を満足するように得れる。 A_i は *Yes* を U に返す。

(b) のとき、 A_i が R_i の一部の解が得られることを、 U に通知する。このとき U は以下の選択を行なう。

- (i) U は R_i をさらに分割し、交渉をやり直す。
- (ii) R_i に対して、他の候補 A_i' がある場合には、 A_i' に交渉を行なう。
- (iii) R_i に対して、候補がない場合には、 R の分割をやり直す。このとき、他に依頼した副要求の実行はアポルトする。

(c) と (d) のとき、 A_i はプレファランスを満たすことができないことを、 U に通知する。 U は利用者と交渉を行ない、承諾が得れば、 A_i にそれを通知する。承諾を得れない場合、上述の (ii) 又は (iii) を行なう。

(e) と (f) のとき、 A_i は R_i の一部のみ得られることを U に通知する。 U は利用者とプレファランスについて交渉を行ない、承諾が得れば上述の (i)、(ii)、(iii) の中から選択を行なう。承諾を得れない場合、上述の (ii) と (iii) から選択を行なう。

(g) のとき、 A_i は実行できないことを U に通知する。このとき U は、上述の (ii) 又は (iii) の選択を行なう。 A_i が実行できるならば、 *Yes* を U に返し、実行準備を行なう。

2. U は、全てから *Yes* を受けたならば、 *Do* を各 A_i に送信し、 A_i に R_i の実行を依頼する。
3. 各 A_i は、 *Do* を受信したならば、 R_i の実行を行なう。 R_i の実行に成功したならば、解を *Done* に含ませて、 U に送信する。 R_i が DB_i の更新を行なう時には、更新結果を安全記憶に格納する。 A_i が R_i を実行できなかった場合には、 *Fail* を U に通知する。

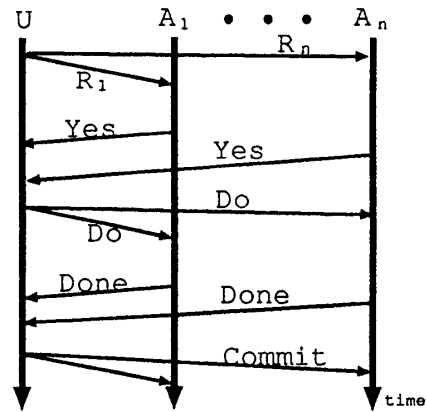


図 2: 交渉手順

4. U は全ての A_i から *Done* を受けたならば、全 A_i に *Commit* を送る。 U が A_i から *Fail* を受けた場合には、上述の (ii) と (iii) から選択を行なう。
 5. A_i は *Commit* を受けたならば、 A_i は、安全記憶内のデータを用いて DB_{A_i} の状態を変化させる。
- R が検索演算のみであれば、4 と 5 の手順は不要である。4 と 5 は、2 相コミット手順である。

5 学習

エージェント間の交渉で、エージェントは自分の知らない項と、項間の関係を得て、これを MDB に記憶することを学習とする。学習によって、以上の他に、項が示す情報を持つエージェントの情報も得れる。これにより、他のエージェントに演算を依頼する時、項が示す情報を持つエージェントと直接交渉を持つことが可能となり、交渉コストを低減できる。

MDB は容量などの物理的な制約が存在するため、交渉で得た項全てを記憶することは困難である。このために、 MDB が満杯で、新しい項又は項間の関係を記憶できないときには、利用頻度の低い項、より詳細な項を MDB から除去することを、忘却とする。

6 おわりに

本論文では、利用者の要求を代行するエージェントから構成される $CDBS$ を提案した。さらに、エージェントの自律性に対処するためのエージェント間の交渉手順を述べた。今後の課題は、エージェントの判断手順の確立と、自律的なエージェントの対処方法の考察がある。

参考文献

- [HASE92] Hasegawa, M. and Takizawa, M., "Group Work System in Research Environment (GRO W)," *Proc. of the 7th JWCC*, 1992, pp.171-178.
- [TAKI91] Takizawa, M., Hasegawa, M., and Deen, M., "Interoperability of Distributed Information System," *Proc. of The First International Workshop on Interoperability in Multidatabase Systems*, 1991, pp.239-242.
- [YAHA92] Yahata, C., Hamada, S., and Takizawa, M., "Cooperating Database Systems," *IPSJ SIG Notes*, Vol.92, No.76, pp.121-128.