

人間-エージェント共生空間での利用を目的としたエージェント間協調プロトコル実現方式

藤田茂[†] 菅原研次[‡] 木下哲男^{††} 白鳥則郎^{††}

E-mail: fujita@cs.it-chiba.ac.jp

[†]千葉工業大学 情報工学科

[‡]千葉工業大学 情報ネットワーク学科

^{††}東北大学電気通信研究所

あらまし

ADIPS フレームワークを人間-エージェント共生空間実現のために利用する場合に対象領域毎の協調プロトコル定義が困難であるという問題があった。本稿では、エージェントと人間が協調して作業を進めるための共生空間において利用される協調プロトコルを対象となる作業の種類に応じて定義するための方式として ADIPS フレームワークのための、エージェントのアーキテクチャについて述べる。人間-エージェント共生空間向けのエージェントでは、これまでのエージェントアーキテクチャの要素であった協調機構と領域知識ベースのメカニズムをエージェント設計者毎に定義可能とすることで、対象領域毎の協調プロトコル処理を実現している。

キーワード

マルチエージェントシステム 人間-エージェント共生空間 ADIPS

An Approach to Describe the Inter-Agent Cooperation Protocol on Human-Agent Symbiotic Space

Shigeru Fujita[†] Kenji Sugawara[‡] Tetsuo Kinoshita^{††} Norio Shiratori^{††}

E-mail: fujita@cs.it-chiba.ac.jp

[†]Department of Computer Science, Chiba Institute of Technology

[‡]Department of Network Science, Chiba Institute of Technology

^{††}Research Institute of Electronics Communication, Tohoku University

Abstract

There is an one problem to use Agent-based Distributed Information Processing (ADIPS) framework on the Human-Agent Symbiotic Space, which is too hard to define the inter-agent cooperation protocol. On the Human-Agent Symbiotic Space, many inter-agent cooperation protocols are required to define on the each domain. In this paper, we proposed new ADIPS agent architecture to define inter-agent cooperation protocol in cooperation mechanism. The new design of cooperation mechanism and domain knowledge-base are shown.

key words

Multi-agent System Human-Agent Symbiotic Space ADIPS

1 はじめに

計算機ネットワークを介して、電子商取引、バーチャルエンタープライズを実現するなど、“ネットワーク社会”を支援するソフトウェアとして、人間と協調可能な“エージェント”が注目されている [4, 5].

我々はこれまで分散システムの自律的構成/再構成を目的として、エージェントによる分散処理システム構成を実現するフレームワークとして、ADIPS: Agent-based Distributed Information Processing System を提案、実装、評価してきた [1, 2, 3].

また、ネットワーク社会のモデルとして、人間とエージェントが透過的に協調していると捉えることが可能な、“人間-エージェント共生空間”の提案を行い、ADIPS フレームワークによる試験の実装を行ってきた [6]. この利用経験から、従来利用してきた ADIPS フレームワークでは、人間とエージェントとの間で発生する協調作業を動的に定義し、この定義に従ってエージェントを動作させることが困難であるという問題が明らかになった。そこで、本稿では、人間-エージェント共生空間での利用を目的として、エージェント間協調プロトコルを定義利用するための ADIPS フレームワークにおける新しいエージェントアーキテクチャの設計について述べる。

2 ADIPS フレームワーク

ADIPS フレームワークは、利用者に対してサービスを提供する分散システムとして動作するエージェント組織の集合が動作する環境である“動作環境”，これら分散システムを構成する部品としてエージェントが保存される“リポジトリ”，部品としてのエージェントを開発する“開発環境”から構成される。これら3つの要素は、ネットワークを介して互いに接続、利用される。図1にADIPS フレームワークの概念図を示す。

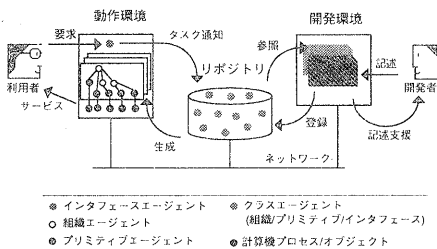


図1: ADIPS フレームワーク概念図

リポジトリには、プリミティブエージェント、組織エージェントの2種類のエージェントが存在する。プリミティブエージェントは分散システムの構成要素である計算機

プロセス/オブジェクトを設計、利用するための知識を持ち、他のエージェント、利用者からの要求に対してサービスを提供する要素として動作する。組織エージェントは、プリミティブエージェントあるいは他の組織エージェントを利用して、利用者要求を充足する分散システムを構成するための、システム/サブシステムを実現する。

動作環境には利用者要求を充足する目的で構成された分散システムと、利用者要求をリポジトリへ伝える利用者インタフェースとなるインタフェースエージェントが存在する。利用者要求は、ウィンドウ、コマンド等を介して、インタフェースエージェントに伝達される。インタフェースエージェントは、この利用者要求を“タスク通知”としてエージェントに理解可能な形に変換し、リポジトリへ伝達する。リポジトリではこのタスク通知を受け取り後述する“ADIPS 構成プロトコル”を用いて、利用者要求を充足する分散処理システムを構築する。

ADIPS フレームワークでは、ある一つの分散処理システムを構成しているエージェント組織を、“ADIPS”と呼称している。これら ADIPS は動作環境上に複数動作することが可能である。個別の ADIPS はそれぞれ動作環境を介して、ネットワークのトラフィックや計算機負荷状況を獲得し、構成要素である個別のエージェントがもつ知識によって決定される動作により、ネットワーク状況に応じたサービスを利用者に提供する。

リポジトリ、動作環境では“ADIPS 構成プロトコル”を用いて、利用者要求を充足する分散システムを動的に構成する。“ADIPS 構成プロトコル”は契約ネットプロトコル [7] を分散システムの構成/再構成に利用するために拡張定義したプロトコルである。この“ADIPS 構成プロトコル”を用いることによって、ADIPS フレームワークは、(1) 利用者要求による自律的システム構成、(2) 利用中のエラーなどのイベント駆動による自律的システム再構成、(3) 設計知識に基づく自律的システム動作パラメータ調整、(4) 既存分散処理システムを構成する計算機プロセス/ソフトウェアモジュールの系統的再利用の4つの特徴を実現している。

開発環境は、ADIPS 構成プロトコルを利用して、エージェントの知識記述者が個別の分散処理システムの設計/運用知識を記述するための環境である。複数のエージェントが同時に動作する環境を対象にしたエージェントの知識記述は、必ずしもエージェントの専門家でない分散システムの設計者にとっては困難であるために、ADIPS フレームワークでは“ADIPS 構成プロトコル”を利用し、エージェントの開発途中で利用可能なエージェントを実際に利用するリポジトリを対象に検索したり、検索されたエージェントの知識記述を参照可能にするブラウザ、エディタによる支援を提供している [3].

3 エージェントアーキテクチャ

3.1 分散処理システム向けアーキテクチャ

ADIPS フレームワークにおけるエージェントはリポジトリ上にあるものを“クラスエージェント”，動作環境上にあるものを“インスタンスエージェント”と定義している。また，それぞれ計算機プロセス/オブジェクトを対象にしているエージェントをプリミティブエージェント，分散システム/サブシステムを対象にしているものを組織エージェントと呼んでいるが，そのアーキテクチャは図2に示すように共通である。以下，本文中ではこれらの分類を特に付けずに“エージェント”として取り扱う。

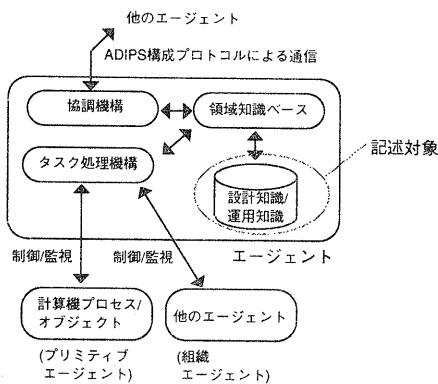


図 2: エージェントアーキテクチャ(旧)

協調機構はエージェント間協調プロトコルに基づいて，他のエージェントとの通信を行い，マルチエージェントシステムを構成する要素として動作するための機構である。領域知識ベースは設計知識/運用知識/運用方針等の領域依存な知識を保持利用してエージェントの動作を決定する推論を行う。タスク処理機構は，プリミティブエージェントにおいては，対象となる計算機プロセス/オブジェクトをインタフェースを介して制御監視する。組織エージェントにおいては，分散システム/サブシステムを構成する組織内部のエージェントと“ADIPS 構成プロトコル”に基づいて制御監視を実施する。

これまでこのアーキテクチャに基づく ADIPS フレームワークは Java 言語を用いて実装され，領域知識ベースにおける処理での一部協調プロトコル処理を目的としてプロダクションシステムの導入が行われた [8] が，プロトコル定義部と領域知識が同一の記述箇所に出現するために，エージェントの知識記述が困難になるという問題があった。次節では，これまでの ADIPS フレームワークを構成するエージェントのアーキテクチャを包含する形式で，人間-エージェント共生空間向けアーキテクチャ

の拡張設計を行う。タスク処理機構については，計算機プロセス/オブジェクトへのインタフェースとしての定義の拡張を行っているが，本稿で述べる協調プロトコルとは直接関係しないのため，協調機構，領域知識ベースの設計についてのみ述べる。

3.2 人間-エージェント共生空間向けアーキテクチャ

図3に人間-エージェント共生空間向けの ADIPS フレームワークにおけるエージェントアーキテクチャを示す。これまでの ADIPS フレームワークでは，エージェント間プロトコルは“ADIPS 構成プロトコル”に固定されていたために，協調機構はプロトコルエンジンとしての処理に専念するのみであった。新たに ADIPS フレームワークを，人間-エージェント共生空間への拡張に適応するために，協調プロトコルが共生空間の領域毎に定義利用されることから，協調機構は推論機構とプロトコル定義部に再設計される。ここで協調機構はメッセージのパフォーマティブ¹とその出現順序を規定したプロトコル定義にしたがって，メッセージへの返答を返すために，領域知識ベースへの問い合わせを行う。

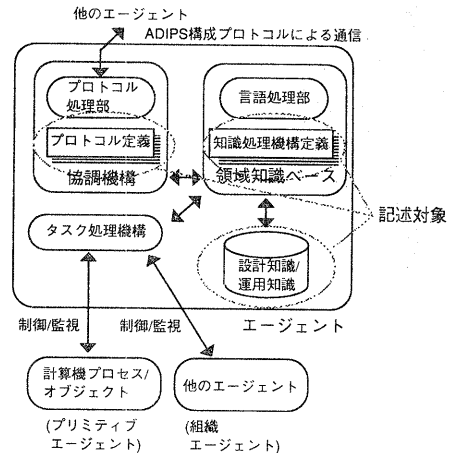


図 3: 人間-エージェント共生空間向けアーキテクチャ

一方，領域知識ベース側は協調機構からの問い合わせに対して，記述された知識に基づく推論を行って返答を返す。これまでの領域知識ベースでは，“ADIPS 構成プロトコル”処理のために分類された知識記述形式として定義されたスクリプト形式に，手続き記述部とプロダクションルールを付加した知識記述形式に制限されていたために，分散処理システム領域以外の人間との協調処理

¹ADIPS では KQML [9] のメッセージ形式を用いている

を実現するようなエージェントの知識記述が困難であるという問題があった。そこで、協調機構と同様に、領域知識ベースを問題解決方式を定義する部分と、個別の領域知識を具体的に記述する箇所に分離した形式へ再設計する。問題解決方式の定義部はプロダクション、プランニング等の知識処理系を実現し、個別の領域知識記述部はそれぞれ定義された形式での具体的な知識処理を実現する。

3.3 協調プロトコル実現とオントロジー

領域知識ベースの知識記述に対しては、設計知識の記述、協調作業知識の記述はオントロジー [10] の一致を設計者間でとることが困難であるという問題がある。文献 [11] では、オントロジーの役割の中でも代表的なものとして、(1) メタモデルとしてのオントロジー、(2) 設計意図を明らかにすること、(3) 標準化の3つをあげている。(1)はモデルとしての表現、(3)は知識処理の高度化、高能率化を対象にしており直接 ADIPS フレームワークにおけるオントロジー利用とは関係しない。一方(2)に関して、これまで ADIPS フレームワークのエージェント記述においては、“ADIPS 構成プロトコル”とプロトコルに依存した知識記述による表現によっていたが、表現能力の不足から必ずしも設計者の意図が知識記述中に表現されているとは言えなかった。エージェントの知識記述者の設計意図が明確に記述できない場合、異なる設計意図をもつエージェントが互いに異なる知識を背景として、協調作業を行うことになり、それぞれのエージェントが想定しない動作をすることになり、人間-エージェント共生空間での利用を目的として不十分であると言える。

エージェントの知識記述者が標準化されたオントロジーを用いて新たなエージェントに対する知識記述を行うことで、明確な設計意図に基づく動作をするエージェント同士での協調が実現されることが期待できる。しかしながら、現在利用可能な分散システムの設計活動を対象にした標準オントロジーが存在しないことや、日々進歩している領域を対象にした標準化の活動が困難であろうことが予測できることや、各分散処理システムの設計活動主体である、企業/コミュニティ毎にオントロジーが異なると予想される。

このため分散処理システムの設計者にとって本来の業務ではない設計意図記述のために、他の設計活動主体との整合性をとったオントロジーによる、エージェント記述作業を行うことは、な作業負担が大きいと予想される。また、標準化されるオントロジーは複数の活動主体のオントロジーを包含して厳密な定義を与える必要があることから、大規模化は避けられず、エージェントが常に標準オントロジーを参照し現実的な時間で処理することが

困難になると予想される。

エージェントの知識記述者にとって最も自然な定義をもち、理解が容易なオントロジーとは、通常の業務で他の作業者と共通して利用されるものであり、これを用いてエージェントの知識を定義、利用することが必要である。しかしながら、知識記述者が必ずしも他の作業者と同一のオントロジーを用いて常に業務を遂行しているとは言いきれない。このため、作業員間で合意がとれていないことが発覚すると、新たにオントロジー定義のフェーズが発生し、新たに合意に達したオントロジーに基づく協調作業が継続することになる。

また作業員にとって明確なオントロジーであっても、計算機処理可能な形式で定義利用可能である必要がある。またエージェントはリポジトリに保管され、時間間隔をもって再利用されることから、オントロジーの保守をエージェントに支援させる必要がある。

新しく人間-エージェント共生空間向けに設計する、協調機構、領域知識ベースを利用する場合、これまでの ADIPS を実現したアーキテクチャと異なり、協調プロトコル定義者と問題解決方式の定義者、個別の領域知識記述者の3者間でオントロジーの不一致による問題が発生することを前提としている。この問題発生をエージェントの動作中に検出し、エージェント間のオントロジー不一致に帰着するか否かをエージェント間の協調により判定する。

4 人間-エージェント共生空間での利用

4.1 エージェント利用モデル

図4に人間-エージェント共生空間のモデル図を示す。人間-エージェント共生空間では、作業員・エージェントが互いに協調してある作業を遂行する。この作業のために構成される空間概念が“人間-エージェント共生空間概念”である。

この人間-エージェント共生空間モデルを、ADIPS フレームワークを利用して生成する手順を以下に述べる。

1. 作業手順、用語定義をリポジトリに保存されたものを利用し、新たに共生空間として記述する。
2. リポジトリ内のエージェント、作業員のリストに対して新たに定義された作業への参加を呼びかける。
3. 必要な作業員、エージェントが決定され、作業員端末には定義された共生空間が準備され、作業員は共生空間を視覚化し表示されたインタフェースを通じて実作業を行う。
4. 共生空間での作業中に発生する協調活動は、1で定義された手順に従って行われる。

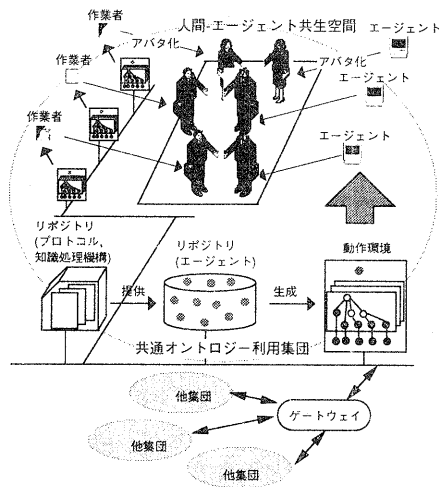


図 4: 人間-エージェント共生空間モデル

問題が発生しなければ作業が終了する。この過程で利用されたエージェント、作業者の活動記録はリポジトリに保存される。

5.4で検出されるオントロジーの不一致に起因する問題の発生は1での作業の問題として、再記述を促す。

上述の作業手順では、(1)作業手順の再利用、(2)用語定義の再利用、(3)エージェントの再利用を示している。次にエージェントを再利用する効果とその適応規模に応じた ADIPS フレームワークと共生空間の関係について述べる。

4.2 エージェント再利用の効果

分散システム開発/人間-エージェント共生空間でのエージェント利用による利点として、過去の活動の再利用が図れるという点がある。しかしながら、積極的に再利用される分散システムの要素は、従来技術であるオブジェクト指向技術等の標準化や文書による標準化とその利用が十分機能していること、人間-エージェント共生空間の論理的な基盤としての現実社会には複雑ながらも、法律や商慣習という標準が存在していることから、全ての分散システム/社会活動を対象として再利用をエージェントで行わせるのは現実的でない。

エージェント利用による利点として期待される対象領域は、分散システムの設計活動作業の記述/蓄積や複数の作業者の存在する協調作業の定式化とその利用の過程である。この対象領域では、(1)新しく作業用のオントロジーを定義する活動、(2)実際にオントロジーを利用して作業を定義する活動、(3)定義された活動の継続の

3つの活動が行わる。この3つの定義をもっとも再利用するのは、その活動をおこなった集団である。そこで図4に示すように、動作環境やリポジトリをグループ化し、グループを越えたエージェント間通信にはゲートウェイを経由することとし、大規模なオントロジーベースの利用を前提としない形式としている。

5 おわりに

本稿では、人間-エージェント共生空間での利用を目的として、協調プロトコルを定義するために、従来あった ADIPS フレームワークを実現するエージェントのアーキテクチャを拡張設計する方式について述べた。

新たに設計されたエージェントは、従来固定であったエージェント間プロトコルを新規に定義利用するための協調機構と領域知識ベースでの推論方式を定義することを可能にした領域知識ベースから構成される。

今回設計されたエージェントはプロトコル定義、推論方式定義をそれぞれ計算機処理可能な形式で行うことにより、エージェントに対する知識記述の際に問題となるオントロジーの不一致による問題の発生を検出させることを目的としている。この目的を実現することにより、人間-エージェント共生空間を記述するという作業を支援している。

今回エージェントのアーキテクチャについて述べたが、これに基づいてエージェント間プロトコル記述言語、推論方式定義言語とそれらに従って知識記述をおこなうことを支援する環境を与える必要と、作業者・エージェント間で利用されるオントロジーの整理を支援する環境を与える必要がある。後者については既にオントロジー整理のためのツール [12] などがあるが、オントロジー整理を行う実際の作業支援を目的としているために、エージェントも直接オントロジーを利用しその定義を変更するような、ADIPS フレームワークにおけるオントロジー導入に適應することは困難であるが、開発環境に同様の機能を持たせることを現在検討している。

参考文献

- [1] 藤田, 菅原, 木下, 白鳥, 分散処理システムのエージェント指向アーキテクチャ, 情報処理学会論文誌, Vol.37, No.5, pp.840 - 852, 1996.
- [2] 菅沼, 藤田, 菅原, 木下, 白鳥, マルチエージェントに基づくやわらかいビデオ会議システムの設計と実装, 情報処理学会論文誌, Vol.38, No.6, pp.1214 - 1224, 1997.

- [3] 原, 藤田, 菅原, 木下, 白鳥, ADIPS フレームワークのための知識記述支援ツール, 情報処理学会論文誌, Vol.39, No.11, pp.3142 - 3145, 1998.
- [4] 丸山文宏, ECにおける商品情報検索のためのエージェント技術, 情報処理学会誌, Vol.39, No.11, pp. 1101 - 1106, 1998.
- [5] 服部文男, エージェント通信技術の動向, 情報処理学会誌, Vol.39, No.11, pp. 1118 - 1123, 1998.
- [6] 今野, 他, 人間 - エージェント共生空間の試作, 情報処理学会マルチメディア通信と分散処理研究会報告, Vol. 89-1, pp. 1 - 6, 1998.
- [7] R. G. Smith, "The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver," *IEEE Trans. on Computers*, Vol.29, No. 12, pp. 1104 - 1113, 1980.
- [8] 分根, 他, プロダクションモデルを利用したADIPS エージェント記述言語の拡張, 電子情報通信学会技術報告人工知能と知識処理, AI98-22, pp. 9 - 14, 1998.
- [9] T. Finin and R. Fritzon, KQML - A Language and Protocol for Knowledge and Information Exchange, Proc., 13th International Distributed Artificial Intelligence Workshop, pp. 127 - 136, 1994.
- [10] 溝口理一郎, 池田満, オントロジー工学序説 -内容指向研究の基礎基盤と理論の確立を目指して-, 人工知能学会誌, Vol.12, No.4, pp. 559 - 570, 1997.
- [11] 溝口理一郎, オントロジー工学への道, 人工知能学会誌, Vol. 13, No.1, pp. 9 - 10, 1998.
- [12] 山口高平, 樽松理樹, 法律オントロジー, 人工知能学会誌, Vol. 13, No.2, pp. 189 - 196, 1998.