

自律分散型ロボットシステムのための 通信機能の設計と通信シミュレータの開発

石田慶樹* 浅間 一** 尾崎功一*** 松元明弘*** 遠藤 勲**

Design of Communication System and Development of a Simulator for an Autonomous and Decentralized Robot System

Yoshiki ISHIDA Hajime ASAMA Koichi OZAKI
Akihiro MATSUMOTO Isao ENDO

An autonomous and decentralized robot system, ACTRESS, is being developed as an intelligent robot system, which performs complex tasks individually or cooperatively. In this paper, the concept of ACTRESS is introduced. ACTRESS consists of various kinds of robotic agents, for example, robots, equipments, and computers. Then, communication framework, between them is proposed. In ACTRESS, communication is indispensable function, therefore communication framework is one of the most important topics in ACTRESS. This communication framework is designed for both achieving reliable communication and exchanging diverse information. In order to analyze and evaluate implementation of communication framework, a communication simulator has been developed, which simulates parallel behavior of robotic agents and communication between them. This simulator is useful for designing and developing the system, especially its communication function. Finally, by using this simulator, the optimal distribution of function is evaluated. Functional distribution is one of the main characteristics of ACTRESS, therefore the configuration of functional distribution among robotic agents should be designed carefully. The simulator is also useful for evaluating optimality of functional distribution from the viewpoint of communication.

Key Words : Autonomous and Decentralized Robot System, Communication Framework, Protocol, Communication simulator, Functional distribution

1. はじめに

自律分散型ロボットシステムは、複数の自律性を持つロボットなどのエージェントにより構成される。自律性は各エージェントがセンシング機能と判断力を持つことにより実現される。一方、分散型システムにおいては、複数の要素を組み合わせるによりある目標を実現する。自律分散型ロボットシステムは、制御を分散し各要素が自律したロボットシステムの構築を目指すものであ

る。

ある目標を複数の自律ロボットにより実現する場合には、各々の自律ロボットには協調性が必要となる。ところが、自律性と協調性は本来は相反する概念であり、このことにより様々な問題が派生することが従来より指摘されている¹⁾。たとえば、油田ら²⁾は複数の自立・自律ロボット間の協調のために「遠慮的協調」を提案している。また、坂根ら³⁾は分散型センサシステムにおいて契約ネットワークプロトコル (Contract Net Protocol)⁴⁾ に基づいた協調方法を提案している。

本論文においては、自律分散型システムにおいて協調

原稿受付 1991年9月6日

* 東京大学工学部 ** 理化学研究所 *** 東洋大学工学部

性を実現するために必要となる機能の一つである通信機能に着目し、その基本設計について述べ、設計結果を評価するために開発したシミュレータについて紹介する。このシミュレータを用いることにより、自律分散システムでの各エージェントに対する機能の配分の形態の評価を行うことも可能である。

2. 自律分散型ロボットシステムACTRESS

現在我々は自律分散型ロボットシステム ACTRESS (Actor-based Robots and Equipments Synthetic System) の研究開発を行なっている。ACTRESS は大規模プラントの保全作業など、高度で複雑な作業のロボット化を目指している。ACTRESS は Fig. 1 に示すように、ロボター (Robotor) と呼ばれる自律性を持つ複数のロボット要素 (agent) から構成される。ロボット要素としては、移動ロボットなどの様々な機能を持つロボット、既存の種々の設備や機器、視覚などのセンサおよびコンピュータなどがある。各ロボット要素は通信により情報 (message) の交換を行ない、各々が自律的あるいは協調的に問題解決を行いながら、全体としての目標を遂行する。

この ACTRESS の特徴は、

1. 自律分散性
2. 多様性
3. 協調性

の3点であり、システム全体の設計理念としては「機能分散」と「協調」の2つの概念に集約できる。

ACTRESS に代表される自律分散システムの長所としては以下の2点が挙げられる。

- ・システム全体の挙動が決定的に規定されていないために、未知の状況や単体では解決することが不可能

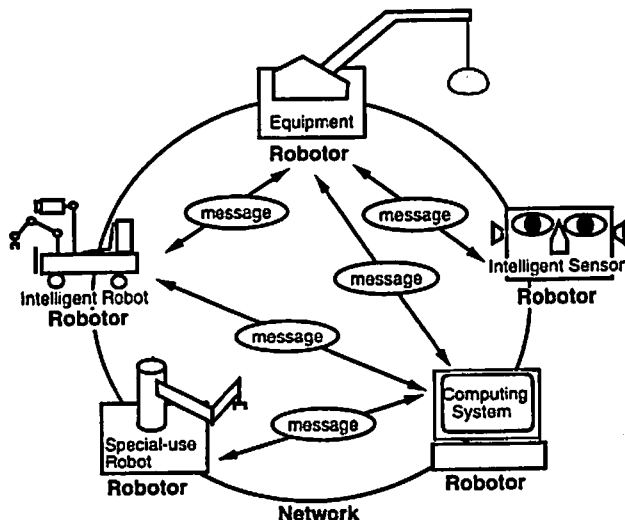


Fig. 1 Concept of ACTRESS

な事態にも対応できるなど、柔軟性に富む。

- ・ある目標を達成するための方法が複数存在し、しかも冗長性を持つために、不測の事態が生じても何らかの試行を行うことが可能で、システム全体としての頑健性が向上する。

一方、短所としては次の2点が挙げられる。

- ・各要素が自律的に問題解決を行うために、システム全体としての最適性が保証されない。
- ・意志疎通を密に行うことが現実的には不可能であり、そのためデッドロックなどが発生する可能性がある。

ACTRESS においては保全作業における多様な作業に対応するために、作業に対しての最適性よりも頑健性を重視している。また、通信機能や各自律要素が持つ問題解決の戦略にデッドロックを解決するための枠組を用意することにより、全体として整合性をとれるシステムにすることを狙っている。この時、デッドロックについては、解決するための枠組として幾つかのレベルを設定し、最終的には人間の判断により問題解決を行うという枠組⁹⁾で検討を進めている。

3. 通信機能の設計

通信を実現するために要求される仕様としては、

1. 信頼性のある通信の実現
2. 多様な形態や内容への対応

がある。この2点を満たすように、通信のための規約であるプロトコルの設計を行う。

通信の実現形態としては、Fig. 2 に示すように (a) 有線通信型、(b) 無線 (電波、光) 通信型、(c) 有線・無線通信併用型の3つの形態が考えられる。ACTRESS においては、このうち (c) の形態を取るものと

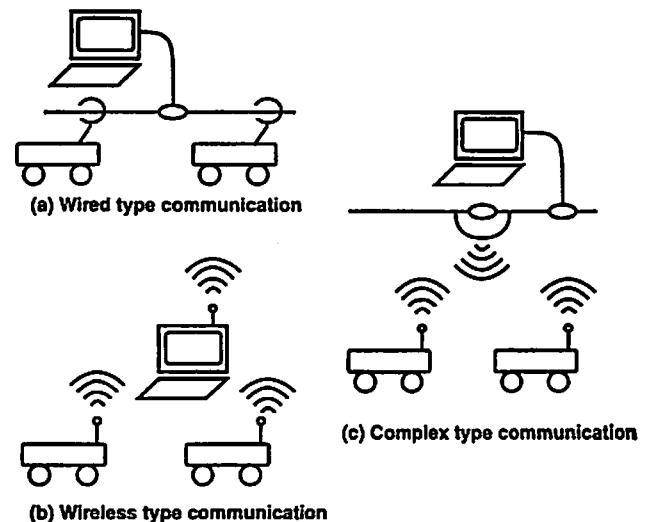


Fig. 2 Types of Communication

	<i>loosely coupled</i>	<i>closely coupled</i>	Robot Language
High	Knowledge Level	Concept Level	↔ Task Level
↑			
	Physical Level	Procedural Level	↔ Object Level
↓	Control Level		↔ Command Level
Low			

Fig. 3 Categories of Information in Robotics

するが、これは (a) (b) の形態を包含している。
次に通信に必要なプロトコルとして、通信形態を規定するプロトコルと、通信内容に関するプロトコルの2つに区分する。前者を通信プロトコル、後者をメッセージ・プロトコルと呼ぶ。

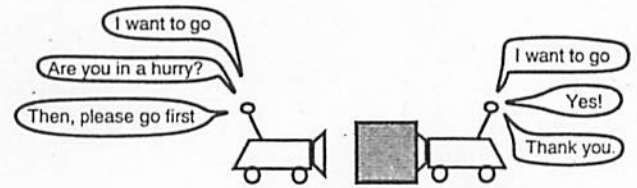
通信プロトコルはコンピュータ・ネットワークにおけるプロトコルに相当するものであり、通常はレイヤ(層)により構造化される。レイヤによる構造化は階層構造化と呼ばれる。階層構造化に関しては OSI の7層参照モデルが代表的である。この階層構造化により通信の信頼性が保証される。しかし、レイヤ構造ではオーバーヘッドが大きくなるという欠点もある。

メッセージ・プロトコルは、特に自律分散型ロボットシステムにおいて必要となる情報交換に着目して設計を行う。これは領域依存のプロトコルであり、先の OSI 参照モデルでは応用層(第7層)のプロトコルに相当する。このプロトコルを設計するために、ロボットシステムの情報交換を考察する。複数のロボット間で交換される情報は内容によっていくつかのレベルに分割されることは、すでにロボット言語の分野を中心として報告されている。一方、ロボット間の情報交換が生じる形態については従来は注意が払われてはいなかった。ロボット間の情報交換が生じる場合を考察すると次の2つの場合が起こり得る。

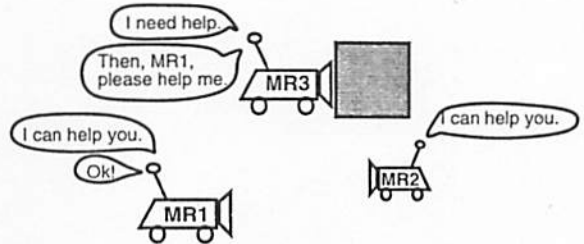
1. 複数のロボットがある共通の目的を実現するために協調するなどロボット間で密な関係を保っている場合。
2. 複数のロボットがそれぞれの個別の目的を実現するのに必要な最小限の情報交換を行っているなどロボット間で疎な関係を保っている場合。

To:	Receiver's Address
From:	Sender's Address
Control:	Field for Message Control
Class:	Message Class
Type:	Message Type
Messages:	Message Body

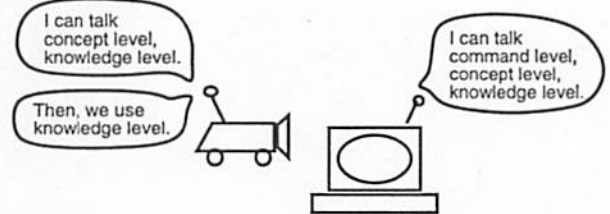
Fig. 5 Framework of Message Protocol Core



(a) Task Negotiation



(b) Cooperation Negotiation



(c) Communication Negotiation

Fig. 4 Three Types of Negotiation

これは先のロボットの言語のレベルとは独立である。これらによりロボット間の情報交換を2つの軸に分けた結果を Fig. 3 に示す。

この分類を基に、メッセージ・プロトコルについても分類を行い、全体をクラス分けする。このクラスを「レベル」と呼び、レベルとしては以下の5種類を用意する。

1. 制御レベル：制御信号やセンサデータなど電気的な信号を交換するレベル。
2. 物理レベル：位置、速度、力など単位を持つ物理量に関する情報を交換するレベル。
3. 手続きレベル：ロボットを動作させるためのコマンドや手続きなどを交換するレベル。
4. 知識レベル：各エージェントが持っているデータや知識を交換するレベル。
5. 概念レベル：意図や目標などの概念を交換するレベル。

ただし全てのエージェントがこれらの全レベルを実装するわけではなく、各エージェントが持つ機能・特性に合わせていくつかのレベルのみを実装することになる。これにより各エージェントが持つ通信機能を過度に重くす

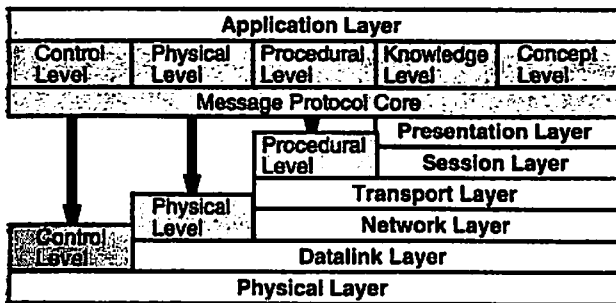


Fig. 6 Protocols in ACTRESS

する必要がなくなる。

しかしこのような場合には、各レベルとは独立にACTRESSを構成するエージェントが、メッセージ・プロトコルとして必ず実装している部分が必要となる。この部分を「メッセージ・プロトコル・コア」あるいは単に「コア」と呼ぶ。コアには、情報の交換と様々な交渉を行う機能が必要である。実際にコアに用意したものとしては、以下の5種類がある。

1. Negotiation : 様々な取り決めを行う (交渉)。
2. Announce : ある情報を広範囲に知らせる (通知)。
3. Inquiry : 特定の情報を獲得する (問合せ)。
4. Offer : 特定の情報を提供する (提供)。
5. Synchronize : 協調動作の同期をとる (同期)。

さらに Negotiation には次の3種類を用意する。

- (a) Task Negotiation : ある動作を行っている時に資源の競合が生じた場合に解消の方法について交渉する。
 - (b) Cooperation Negotiation : 協調作業を行う場合にその形態について交渉する。
 - (c) Communication Negotiation : メッセージ・プロトコルにおいてどのように通信するかを交渉する。
- この Negotiation の例を Fig. 4 に示す。(a) の Task Negotiation ではお互いに相手のいる方向に走行し、衝突が生じる可能性がある状況下で、双方が通信でどちらが先かを交渉し決定する。(b) の Cooperation Negotiation ではある作業が1台のロボターで不可能な時に、協力者を探してどのロボターと協調作業を行うかを交渉し決定する。(c) の Communication Negotiation ではメッセージ・プロトコルのレベルの中でどのレベルを用いて情報交換を行うかを交渉し決定する。

Fig. 5 にコアの枠組を表わす。各メッセージは送信先となる To フィールド、発信元の From フィールド、メッセージの緊急度を表わす Control フィールド、メッセージのクラスつまりどのレベルもしくはコアであるかを表わす Class フィールド、その中での型を表わす Type フィールドから構成される。

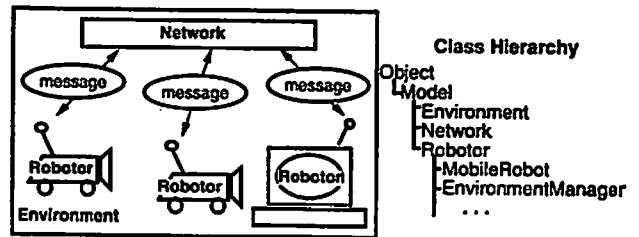


Fig. 7 Objects in the Simulator

ACTRESS の通信機能として、通信プロトコルとして OSI プロトコルを用いた場合を Fig. 6 に示す。階層構造化した通信プロトコルではオーバーヘッドがあるために、特に高速な情報交換が要求されるメッセージ・プロトコルの下位のレベルにおいては、レイヤの省略を行う。このレイヤの省略は予め固定的に行う場合と、Communication Negotiation により行う場合の両方が起こり得る。

4. 通信シミュレータの開発

ACTRESS において、通信機能の設計あるいは機能分散や協調の戦略を決定するためには、純粋な理論的な解析だけでは不十分であり、様々な実験による評価が必要である。このような実験を行うための土台の一つとして、現在ソフトウェアによる通信シミュレータを構築し、様々な状況下での評価を行っている。

本通信シミュレータに要求される仕様としては以下の点が挙げられる。

1. ロボター間のメッセージ交換を容易に実現できること。
2. 通信の様々な特性を設定することができること。
3. 環境との相互作用をシミュレートできること。
4. 並行動作を容易に記述できること。
5. 様々な視点に基づく評価が可能となっていること。

これらの仕様を満たすために、本シミュレータはオブジェクト指向型言語 Smalltalk-80⁶⁾ によりインプリメントしている。Smalltalk-80 を利用した理由は以下の通りである。

- ・オブジェクト指向型言語では、オブジェクト間のメッセージの受け渡しにより計算が進行する。ロボターをオブジェクトとすると、ロボター間のメッセージ交換をオブジェクト間のメッセージ交換としてそのまま実現することができる。
- ・シミュレータを構成する全ての要素をオブジェクトとすることによりそれぞれの特性の変更が容易である。
- ・マルチプロセス OS に用意されているプロセスを並列に実行するための機能を擬似的に実現することができるので、並行動作の記述が比較的容易である。

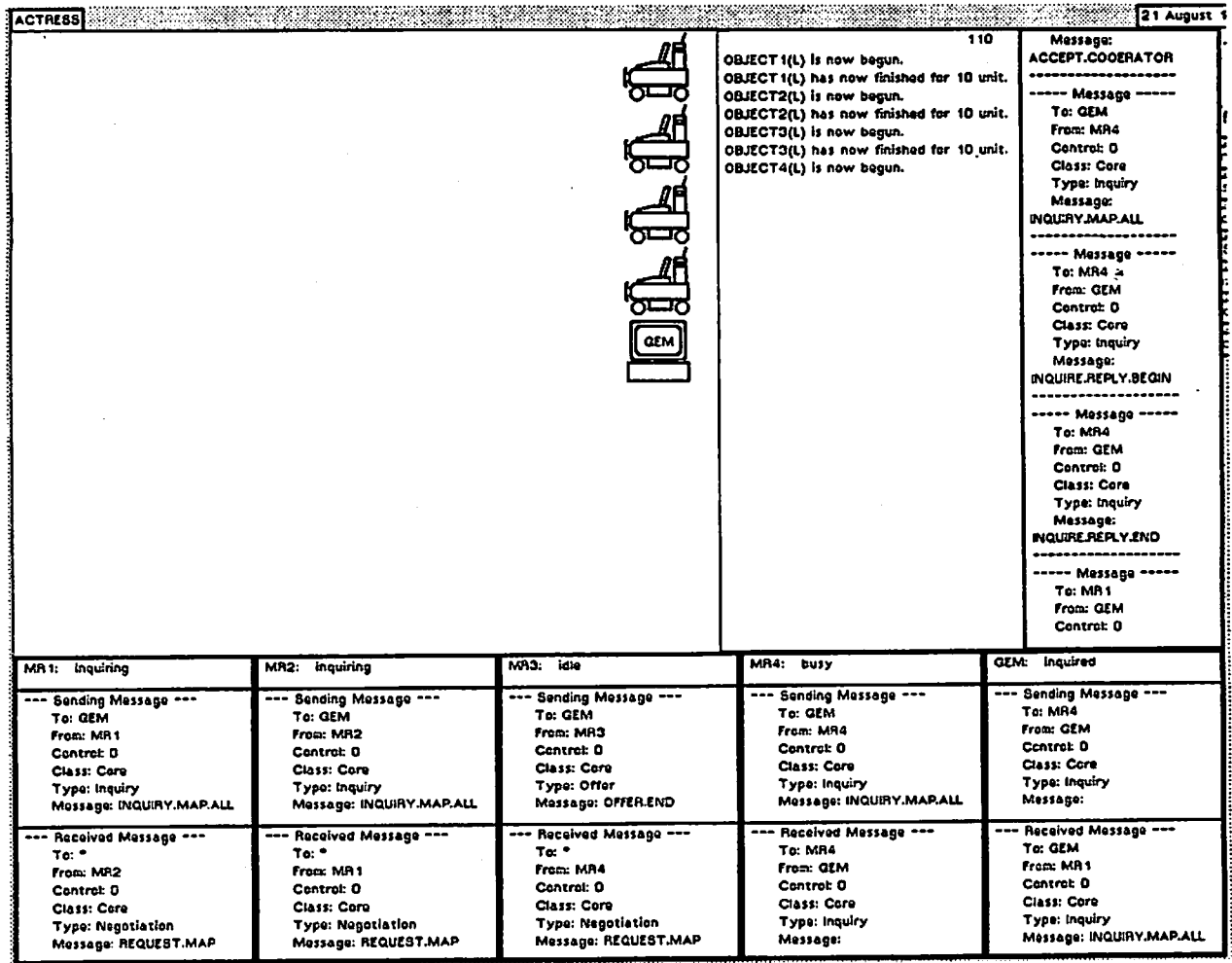


Fig. 8 Hardcopy of the Simulator

・Smalltalk-80 は統合的プログラミング環境が提供されているため、簡単にユーザ・インターフェースを構築できる。

Fig. 7 にシミュレータにおけるオブジェクトの構成を示す。本シミュレーションでは並列に動作する複数の Robotor (Mobile Robot, Environment Manager など)、通信媒体を抽象化した Network, および Robotor との物理的な干渉を行う Environment がオブジェクトとして存在している。個々の Robotor はメインループとなるメソッドを持ち、そのメソッドは Robotor 毎で独立

なプロセスとなり、Smalltalk-80 の並列プロセスの機能を用いて並列に実行される。メインループの中で、各 Robotor は Network を介した通信により他 Robotor とメッセージ交換を行いながら、Environment に作用することで作業を実行していく。Fig. 8 はシミュレータの実行画面である。

本シミュレータを用いることにより、ロボタ一群への機能分散の形態や各ロボタが持つ問題解決の戦略により全体がどのような挙動を示すかを、通信量などのいくつかの視点から評価することが可能になる。

Table 1 Comparison of amount of exchanged information

Type	Constructing initial models	Exchanging environmental information for path planning	Reporting pushing object	Total
(A)	m	m^2	m	$m^2 + 2m$
(B)	nm	0	$(n - 1)m$	$2nm - m$
(C) (if $n \geq m$) (if $n < m$)	$m + nm$	0 $(n - 1)(m - n)$	m	$nm + 2m$ $2nm + m - n^2 + n$

m : the number of objects n : the number of mobile robots

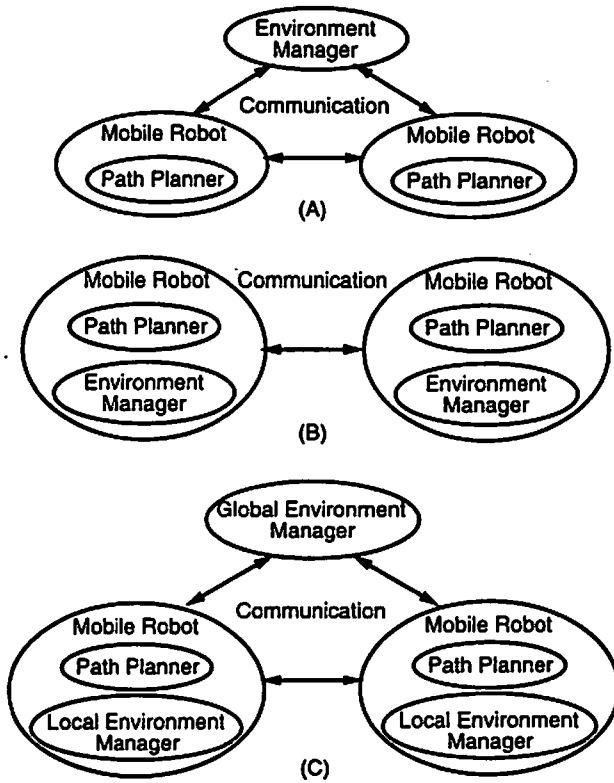


Fig. 9 Typical Configuration of Distributed Functions

5. 通信シミュレータを用いた評価

ここでは通信シミュレータを用いることにより解析した機能分散の評価結果について述べる。機能分散の評価対象は移動ロボットのための環境情報の管理形態である。環境情報の管理形態としては集中型管理と分散型管理が考えられる。

このとき、機能分散の形態としては、Fig. 9 に示すように

- (A) 環境モデラが集中的に環境情報を管理する。
- (B) 全ての移動ロボットが環境情報を個々に管理する。
- (C) 大域的な環境を環境モデラが管理すると同時に、各移動ロボットは個々に局所的情報を管理する。

の3種類を考えることができる。ここで、様々な作業の実現のために全ての要素間での通信が可能であるとする。

ここで、対象とする作業を荷片付けとする。すなわち、作業空間内に複数個の荷物が存在し、これらを複数台の移動ロボットにより片付けるものとする。荷物が m 個、移動ロボットが n 台存在し、1個の荷物について状態変化が生じた場合について1回の情報交換が行われるとした時に、(A)(B)(C)のそれぞれについて通信量(通信回数)を解析した⁷⁾。ただし、ここで求めた通信量は環境情報の交換だけに必要な通信量である。

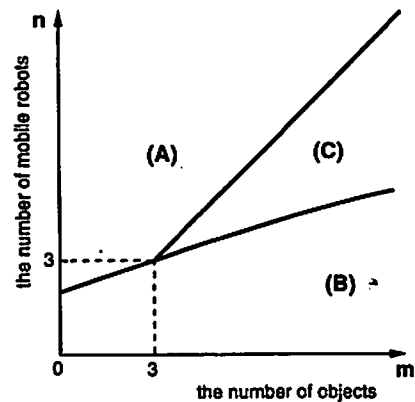


Fig. 10 Evaluation of Minimum Information Transmission

まず環境情報を初期化するために、(A)、(B)、(C)のいずれの場合においても、すべての環境マネージャに荷物 m 個の情報を伝達する必要がある。このためには1台の環境マネージャに対して m の通信量が必要である。初期化のために必要な通信量は、(A)、(B)、(C)でそれぞれ m 、 nm 、 $m+nm$ となる。次に経路計画時には、(A)の場合では1個の荷物を移動するための計画を行う毎に環境モデラから移動を計画している移動ロボットに全ての荷物の情報を伝達する必要があるため m の通信量が必要となり、 m 個の移動を行うために必要な通信量は m^2 となる。(B)の場合は各移動ロボットが環境情報を持っているために情報交換は必要ない。(C)では、移動ロボットの台数が荷物の個数より多い時には($n \geq m$)、最初の移動で全ての荷物の移動が実行され、この経路計画には初期情報だけで済むために、情報交換は必要ない。移動ロボットより荷物が少ない時には($n < m$)、最初の移動で残った $m-n$ 個の荷物のそれぞれを移動する場合に、移動を計画している移動ロボット以外の移動ロボット $n-1$ 台が移動した荷物 $n-1$ 個の情報が必要となるために、合計で $(n-1)(m-n)$ の通信量が必要である。最後に1個の荷物の移動が終了すると、それを全て自分以外の環境マネージャに伝達する必要があるために、 m 個の荷物では(A)、(B)、(C)でそれぞれ m 、 $(n-1)m$ 、 m の通信量が必要となる。

この解析をまとめたものをTable 1に、また、この結果について m 、 n をパラメータとして通信回数が最小となる場合をグラフ化したものをFig. 10に示す。この解析によると、移動ロボットの台数が荷物の個数よりも多い場合には(A)の形態が最小となり、荷物の台数に比べて移動ロボットが極めて少数である場合には(B)の形態、それ以外の場合には(C)の形態が最小となることがわかる。実際には、移動ロボットの台数は荷物の個数より少なく、しかも移動ロボットの台数自体はある程

Table 2 Result from the Simulation

	m											
	2			6			10			14		
	(A)	(B)	(C)	(A)	(B)	(C)	(A)	(B)	(C)	(A)	(B)	(C)
n=2	24	10	24	84	30	72	176	50	136	300	70	216
n=4	24	26	30	90	78	86	182	130	154	306	182	238
n=6	24	42	32	96	126	96	188	210	172	312	294	260
n=8	24	58	36	96	174	108	194	290	190	318	406	282
n=10	24	74	40	96	222	120	200	370	200	324	518	308

unit: the number of packets

m: the number of objects n: the number of mobile robots

度存在することが考えられるために、(C)の形態が最も好ましい。また、情報の整合性のある管理を考えると、(A)>(C)>(B)の順に有利であるので、この通信量と整合性の両者を満たすものとして(C)の形態が好ましいと結論が得られた。

この結果を通信シミュレータを用いて評価を試みた。ここで、各移動ロボットは作業開始時に最新の環境情報を持たない場合には、まずそれを獲得する。協調者である環境情報の管理者が誰であるか知らない時には、“Cooperation Negotiation”を行い、協調者を捜し出す。そして協調者に対して、“Inquiry”により環境情報を問い合わせる。作業終了時に“Offer”により終了報告を行う。実際の通信量にはこれらが含まれる。

この評価を行うために次のような仮定をおいた。まず、通信においてエラーが生じないものとし、通信量は通信回数(パケット数)で評価することとした。次に、作業は一定時間で1台の移動ロボットにより実行されるものであるとし、また荷物の取り合いなどの作業の競合は生じないとする。

この仮定下での評価を Table 2 に示す。また、これを

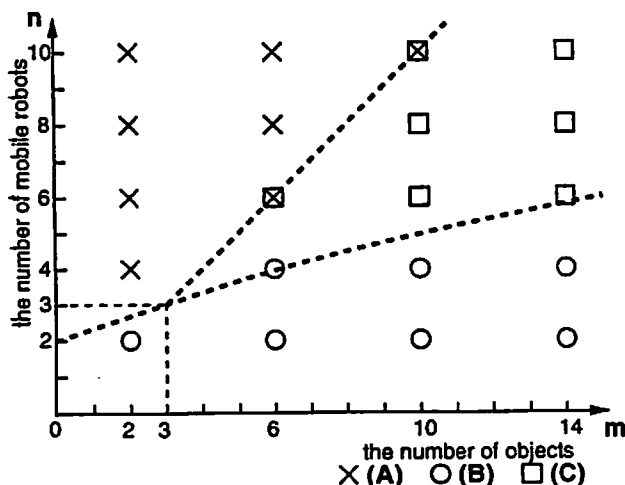


Fig. 11 Result of the Evaluation

グラフ上にプロットしたものが Fig. 11 である。

これらの結果により、以下のようなことが考察として得られる。

- ・シミュレーションによる評価結果は理論的解析と傾向として一致している。
- ・結果は全体のインプリメンテーションに依存する。
- ・実際に地図情報の管理の形態によって結果が異なる場合もある。例えば(C)が個々に持つ情報の更新を頻繁に行った時は(A)の場合と同じになる。
- ・(B)や(C)の形態を実装するためには情報の整合性を保つことに注意する必要がある。これらは分散データベースの主要な課題と同様である。
- ・パケット数より通信量の評価を行ったが、実際には通信内容によりパケットの大きさが異なってくる。特に地図情報の形式によっては結果に影響を与える可能性がある。

またこれによりシミュレータを検証することが可能である。

- ・本シミュレータは通信機能を中心としてロボターの様々な機能をシミュレートすることが可能である。また本システムを用いることにより様々な評価が可能である。
- ・容易に実装形態を変更することが可能であるため、様々な状況毎の評価が可能となる。
- ・シミュレータのインプリメントを実際のロボターに移植することも可能である。
- ・本シミュレータは ACTRESS ならびにロボターの設計と開発にとって有効である。

ACTRESS の設計と開発のためには様々な評価が必要である。ここでは通信の視点から評価するためのシミュレータを構築し、それを用いて機能分散の評価を行った。これにより、機能分散の最適性の評価ばかりでなく、シミュレータの適用性についても検証できた。

6. おわりに

本論文では、はじめに自律分散型ロボットシステム ACTRESS の概要について紹介した。ACTRESS は「機能分散」と「協調」を基本概念としたものである。

次に ACTRESS の通信機能を設計した。通信機能についてはまず形態を定め、通信のためのプロトコルを設計した。プロトコルとしては通信形式を規定する通信プロトコルと通信内容を規定するメッセージ・プロトコルにわけ、さらにメッセージ・プロトコル内にレベルとコアを用意する。

通信機能の評価を行うために通信シミュレータを開発した。通信シミュレータは Smalltalk-80 により記述さ

れ、複数のロボターの挙動とメッセージ交換をシミュレートできる。

最後に通信シミュレータを用いて機能分散の評価を行った。これにより機能分散が評価できたばかりでなく、通信シミュレータの適用性についての検証も行った。

今後は、実際の通信の実現、通信に基づくデッドロックの解消、協調作業の実現などに取り組む予定である。

参 考 文 献

- 1) 油田, プレムウッチェ: "複数の自律移動ロボットの協調行動のアーキテクチャ", 第1回ロボット学会ロボットシンポジウム予稿集, pp.67-72, 1991
- 2) 油田, プレムウッチェ: "複数の移動ロボットの協調に関する検討—遠慮的協調の提案—", 第8回ロボット学会学術講演会予稿集, pp.887-890, 1990,
- 3) 坂根, 佐藤, 丸矢: "ロボット視覚のための分散協調センシングシステム", 第8回ロボット学会学術講演会予稿集, pp.907-908, 1990
- 4) Smith, R. G. and Davis, R.: "Frameworks for Cooperation in Distributed Problem Solving", IEEE

- Trans. Sys., Man and Cybernetics, Vol SMC-11, No.1, pp.61-70, 1981
- 5) 浅間, 尾崎, 板倉, 松元, 石田, 遠藤: "通信及びルールに基づく複数の移動ロボットの衝突回避", 第1回ロボット学会ロボットシンポジウム予稿集, pp.73-78, 1991
- 6) Goldberg, A., and Robson, D.: "Smalltalk-80: the Language", Addison-Wesley, Massachusetts, 1989
- 7) Asama, A., Habib, M. K., Endo, I., Ozaki, K., Matsumoto, A. and Ishida, Y.: "Functional Distribution among Multiple Mobile Robots in an Autonomous and Decentralized Robot System", Proc. of the 1991 Int. Conf. on Robotics and Automation, pp. 1921-1926, 1991



石田慶樹 (Yoshiki ISHIDA)

1961年10月18日生。1988年東京大学大学院工学系研究科精密機械工学修士課程修了。1988年東京大学工学部助手、現在に至る。分散型ロボットシステムの通信、コンピュータ・ネットワークの研究開発に従事。WIDEプロジェクトのメンバ。IEEE, 精密工学会, 情報処理学会などの会員。

(日本ロボット学会正会員)



浅間 一 (Hajime ASAMA)

1959年1月18日生。1984年東京大学大学院工学系研究科精密機械工学修士課程修了。1986年理化学研究所化学工学研究室研究員補、現在研究員。分散型知能ロボットシステム。バイオプロセスの知能化技術の開発に従事。工学博士。IEEE, 精密工学会, 日本機械学会, 化学工学会などの会員。

(日本ロボット学会正会員)



尾崎功一 (Koichi OZAKI)

1967年6月10日生。1992年東洋大学大学院博士課程前期課程修了。現在埼玉大学大学院博士後期課程在学中。自律分散型ロボットシステムのための通信, 移動ロボットとその協調に関する研究に従事。

松元明弘

(Akihiro MATSUMOTO)



1958年6月8日生。1983年東京大学大学院工学系研究科精密機械工学修士課程修了。1983年東京大学工学部助手, 1988年東洋大学工学部講師, 1990年同大助教授, 現在に至る。工学博士。ロボット言語, 自律分散型移動ロボットシステムの研究に従事。精密工学会, 日本機械学会, IEEE の会員。(日本ロボット学会正会員)



遠藤 勲 (Isao ENDO)

1940年8月14日生。1965年東京大学工学部化学工学科卒業。1970年東京大学大学院工学研究科博士課程卒業, 理化学研究所化学工学研究室勤務, 現在, 同主任研究員, 生物工学, 特にバイオプロセスエンジニアリングの研究に従事, 1978年化学工学協会論文賞受賞, 化学工学会, 発酵工学会, 日本農芸化学会, 計測自動制御学会, アメリカ化学工学会の会員。

(日本ロボット学会正会員)