

分散協調機能をもつ自律移動ロボットシステムの試作 (I)

3W-6

大分大学工学部

副島 匡暢 宇津宮 孝一 肥川 宏臣 児玉 利忠

1 はじめに

マルチロボットによる協調問題の研究は、人工知能やロボティクスなどのさまざまな分野で豊富な概念試行の環境を与えている [1]。

近年、マルチロボットを用いた研究は盛んであるが、これらの研究の多くは仮想環境による概念シミュレーションによるものが多い。これは、適切なハードウェアが使用できないことや、リアルタイム性の高い実験が難しいことによる。完全なロボットシステムは複雑であり、必要なハードウェアの開発もかなりの時間を要する。

以上のことを踏まえて、我々はマルチロボットシステムのテストベッドとして、Gung-Ho("work together"の意味)システムを構築している。このシステムは、協調動作の基本概念を評価するためのマルチロボットシステム開発を支援するものである。特に、Gung-Hoシステムでは、ロボットの制御にUNIXワークステーションを利用することにより、UNIX上のソフトウェア開発ツールやマルチタスク機能、TCP/IPやXウィンドウシステム等のネットワークコミュニケーション機能やユーザインタフェース機能などを最大限に利用できる。これらの環境により、マルチロボットシステム開発における仕様確認の迅速性や変更の柔軟性を達成することができる。

2 Gung-Ho システム

我々はマルチロボットシステムに関する研究のなかでも、簡易ロボットを用いた分散協調アルゴリズムの開発などに重点を置いている [2]。Gung-Hoシステムの概要を図1に示す。これらのロボットは、既知の環境上で自律協調動作できるように設計する。それぞれのロボットは、システムの柔軟性と簡易性のため、移動部と制御部を分離した構成とした。制御部はUNIXワークステーション上におき、この制御部を介して移動部を遠隔制御する。

- 移動部 各ロボットは直径20cm、高さ25cm大で、移動部にオンボードプロセッサ(MC6809)、左右車輪用の2つのモータ、前後2つの衝突感知用タッチセンサおよび前後左右の4方向距離測定用の超音波センサをもつ。オンボードプロセッサは制御部からの制御コマンドに対するモータとセンサの動作制御やセンサ情報のフィードバックなどを行う。制御コマンドとセンサ情報は移動

Gung-Ho: A Testbed for Prototyping Distributed Cooperative Autonomous Robots(I).  
Masanobu Soejima, Kouichi Utsumiya, Hiroomi Hikawa and Toshitada Kodama, Oita University.

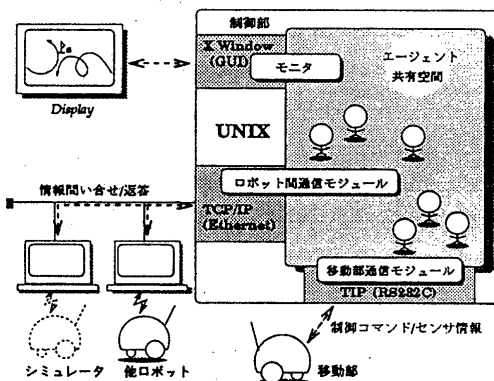


図1 "Gung-Ho" マルチロボット開発システム

部と制御部の間を結んだ無線通信機構により交換される。移動部は制御コマンドにより、移動や障害物までの距離計測などのようなプリミティブなタスクを実行する。

- 制御部 制御部は包括アーキテクチャ [3], [4] を基にUNIXワークステーション上にソフトウェアで構築する。各ワークステーションは各ロボットを制御し、イーサネットを通じてコミュニケーションがとられる。制御部内のコミュニケーションは、UNIXの共有メモリ機能を用いて行われる。ロボット動作は制御部内の各エージェントにより生成され、エージェント間のコミュニケーションとともにアプリケーションとして定義される。エージェントにより生成されたロボット動作は、各モジュールにより処理され、任意のワークステーション(他ロボットの制御部)や自分の移動部へとの情報交換を行う。

- シミュレータ・モニタ 開発や実験のためのテストベッドとしてのシステムは、ソフトウェア開発やデバッグもうまく支援すべきである。Gung-Hoでは、移動部をUNIX上の仮想的なプロセスで置き換えることにより、シミュレータとしての機能をもつ。シミュレータはロボットと同様のコマンドを受け取り、環境のマップ情報を用いて実際にロボットがいるべき環境の情報を返す。モニタプロセスは、共有メモリ上の位置情報やセンサ情報などをXウィンドウシステムを用いてグラフィカルに描き出す。これらにより、共有メモリを通して実際のロボットとシミュレータの共存する実験環境の構築が可能である。我々は、これらの二つの機能がシステム開発における初期段階のテストやデバッグに非常に有効なものになると考えた。

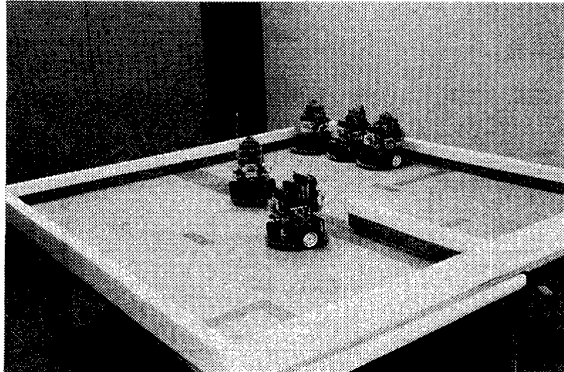


図2 実験風景

### 3 動作例

ここでは、以下のようなエージェントを用い、ロボット同士は位置情報を交換しながら衝突を避け目的地まで移動していく。

- ・目標追跡エージェント：エージェントは、与えられた目標に対して近づくようなコマンドを与える。
- ・衝突回避エージェント：エージェントはセンサ情報や他ロボットの位置情報を監視し、ロボットが衝突を回避しなくてはならない場合にコマンドを与える。
- ・通信エージェント：必要な情報を監視し、他ロボットに問い合わせや返答をするためのコマンドを与える。

これらのエージェントは共有空間（エージェント間の共有メモリ）を通じて様々な情報交換を行う。エージェントにより生成されたコマンドは、ひとつのコマンドに統合されてロボットもしくはシミュレータに送られる。

これらのエージェントはすでに実現され、シミュレータとモニタによりテストされている。そこで、これらのエージェントを実際のロボット群（図2）を用い実験を行った。

この例においては2台のロボットが使用されているが、1台は物理的に存在するロボットで目標を追尾する。もう1台は移動部をシミュレータに置き換えたシミュレーションロボットで物理的には存在しない。現実環境では、ロボットはあらかじめ与えられた目的地に向かって、物理的には存在しないシミュレーションロボットを避け軌道を変更しながら動作した。図3は、両方のロボットが動作した軌跡を示すモニタ画面である。図3で観察するとおり、ロボットは実線に沿ってほぼ正確に動作し、あらかじめ設定された目的地（[dest]）へシミュレーションロボット（破線）を避けながら到達することができた。

この例を行うためのサンプル制御プログラムは1人で約半月間で開発できた。このように、自律協調動作を行なうために必要な機能は、UNIXを用いることにより簡易的に実現することができる。Gung-Hoシステムはテストベットの構築に効果的なプログラミング環境を与える。

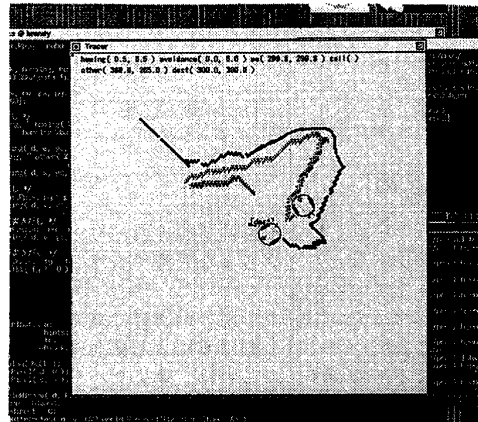


図3 モニタ画面

### 4 結論

我々は、マルチロボットシステムのテストベット開発に関して、その検討と試作を行なった。

ロボット処理系をソフトウェアにより構築することは、迅速性、柔軟性、拡張性の点で非常に有利である。このシステムはUNIXワークステーションを基盤にすることにより、UNIXが支援するマルチタスク機能、ネットワーク機能およびユーザインタフェース機能およびプログラミング環境などを有効に利用することができる。ワークステーションは制御部として移動部を遠隔操作することにより、これらの構造は包括アーキテクチャを用いたマルチロボットシステムの構築について非常に有効である。このことにより、その他のさまざまなアーキテクチャに柔軟に対応していくことが可能であろう。

このサンプルプログラムの開発により、マルチロボット制御システムの開発ツールとしての効果性を示すことができた。我々はさらに、協調マルチロボットシステムのテストベットの基礎開発を行っていく。

### 参考文献

- [1] S. Premvuti, S. Yuta : Consideration on the Cooperation of Multiple Autonomous Mobile Robots, *IEEE Int. Workshop on Intelligent Robots and Systems*, pp. 219-223, 1990.
- [2] 副島, 宇津宮, 児玉, 凍田, 吉田 : 協調機能を持つ分散ロボットシステムの検討と試作, *情報処理学会第44回全国大会論文集*, Vol.1, pp. 283-284, 1992.
- [3] R. A. Brooks : A Robust Layered Control System for a Mobile Robot, *IEEE Journal of Robotics and Automation*, Vol.RA-2, No.1, pp. 14-23, 1986.
- [4] Jonathan. H. Connell : Minimalist Mobile Robotics, *ACADEMIC PRESS*, 1990.