

曲げ加工ロボットのシミュレーションシステムの開発

東京農工大学 工学部 機械システム工学科

教授 遠山茂樹

(平成6年度研究開発助成 AF-94016)

キーワード：自律分散システム

1. 研究の目的と背景

1. 1 目的と背景

産業用ロボット単体としての作業能力も限界に近づき、最近では複数のロボットを集中管理し作業する方法がとられるようになってきた。

このような方法は一般的にシステムの最適化や高効率化を図りやすい一方、システムの一部の故障が全体に影響を及ぼしたり、設計の変更にフレキシブルに対応できなくなるという問題も生じてしまう。そのため、現在のシステムの方向性は最適化と高効率化から信頼性（対故障性）と柔軟性重視へと変わりつつある。このような背景からシステム全体を統合する管理機構を持たない自律分散システムが注目され、盛んに研究されるようになってきた。

そこで本研究では、比較的単純作業が多くあまりロボット化されていないいうえ、システムの柔軟性が要求される曲げ加工を対象に自律分散システムを開発した。

1. 2 自律分散システム

自律分散システムとはシステム全体を統合する管理機構を持たず、各要素がシステムの目的及び環境などに基づいて自らの行動を決定することにより要素間の協調を図り、全体としての秩序を生成・維持するシステムのことである。

システムを構成する各要素は個々に他の要素の挙動とシステムの目的及び外部環境に関する情報を何らかの方法で取得し、各要素間でその情報を互いに交換しあい、自らの行動を自立的に決定する。

このようなシステムの機能として次のものが挙げられる。

- システム全体を統合する管理機構を持たないので、一部の要素の故障によりシステム全体が停止することがない。これによって信頼性の向上が望める。
- 故障に備えての予備系の用意がなくなるため、コストダウンにつながる。
- 各要素が周りの環境に対応しながら全体としての秩序を生成するため、変化の激しい作業環境にも柔軟に対応できる。

このようなシステムの構成図を図1に示す。

自律分散システムは、まだ確立された技術に基づいて構成さ

れるまでには成熟していない。それは解決すべきいくつかの問題点が残されているためである。

- 自律分散システムでは、各ロボットが互いに譲り合い協調し合って全体の調和を維持し、さらにシステムの目的を達成しようとする強い力が働くなければならない。しかし、互いに譲り合ったり、自分の主張をとおそうとしたのでは、システムはたちまち膠着状態、デットロック状態に陥ってしまう。この問題はシステムが複雑になるほど重大になる。
- 複数のロボットで作業を分担する場合、各ロボットが担う作業量はそれぞれの能力に応じて公平に分担することが必要になる。また、あるロボットの故障などにより、各ロボットに能力以上の負荷をかけてしまわないような工夫も必要になる。
- ロボット間の通信には一般にコストがかかる。システムが複雑になり多くの相手と通信を交わしたり、通信の内容が複雑になったりすると設備費がかさむし、通信のための時間もかかるてしまう。

以上のような問題をいかに解決するかがこのシステムの定着、発展の鍵となる。

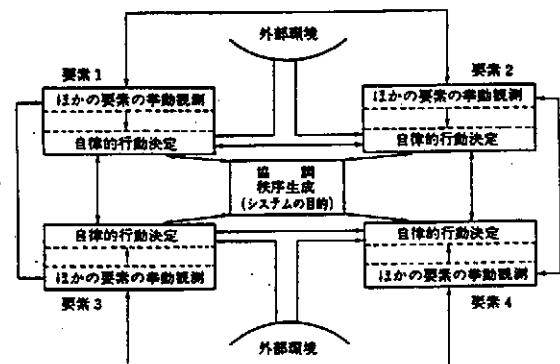


図1 自律分散システムの構成図

2. 実験方法

2. 1 システムの構成

パソコン上で動くロボットシミュレータを用い、2種類のシステムを構成する。それぞれのシステムには2組の要素を用意し、それぞれ別の作業をさせる。各要素間の通信は全て、単純な信号のやりとりによって行い、故障のタイミングは設定できるようにした。

TYPE1：2台のロボットを1組とし、1台がワークの搬送、保持をし、もう1台が曲げ加工をする。システム稼働中、任意のタイミングで2台までが故障しても作業を続行する。

TYPE2：TYPE1を拡張し、1台の曲げ加工機と1台のロボットを1組とし、ロボットはワークの搬送、保持をする。1台の加工機は一度に一種類のワークしか加工できず、加工機が故障した場合、TYPE1に切り替え作業を続行する。

ここで用いるロボットは、工場で部品などを搬送する無人カートにマニピュレータを乗せたもので、その構造をなるべく簡単にするため、特別なセンサーなどは取り付けず、他の要素との通信機能だけを持たせることにした。図2にロボットの外観を示す。通信内容も作業プログラムの判断が終了した時や、加工開始と終了時及びロボットの故障が発生した時にそれぞれ固有の信号を発信または受信するようにした。

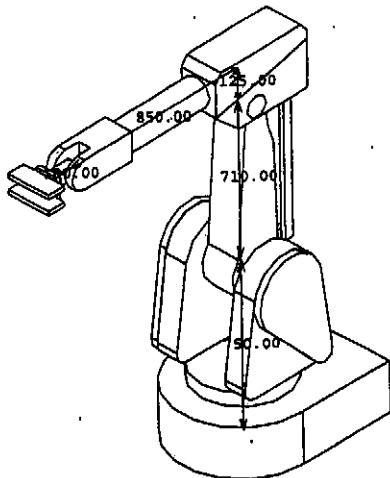


図2 ロボットの外観

また、作業プログラムと加工の開始、終了のタイミングをワークからの指示により判断させるようにした。こうすることによりワークを認識、識別させるための特別な機構を組み込む必要がなくなる。さらに、同じタイミングで2つ以上の指示ができないよう次のワークは前のワークがロボットに作業プログラムの指示を出し終わるまで待機しているようにした。

図3にワークがだす指示の例、図4に作業プログラムの例を示す。

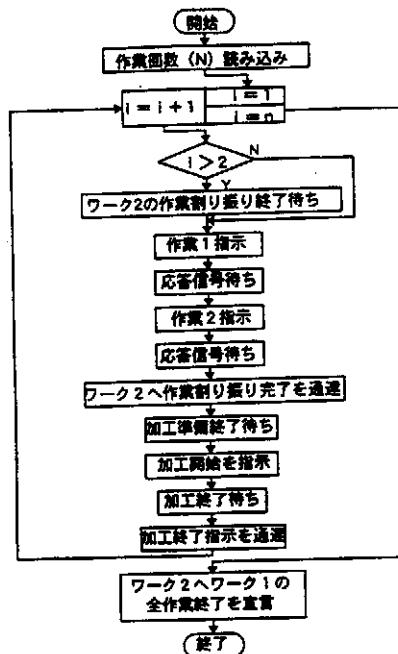
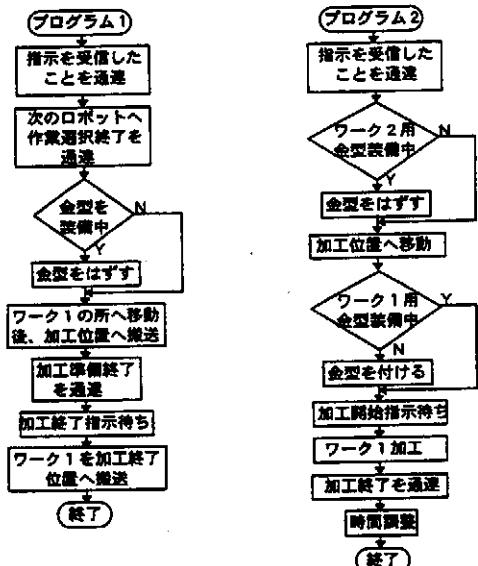


図3 ワークがだす指示の例



(A) 作業プログラム1

(B) 作業プログラム2

図4 作業プログラムの例

次にこのロボットの特徴を示す。

- ロボットには優先順位があり、その順位に基づき作業プログラムの判断を開始する。これは複数のロボットが同時に同じ作業プログラムを選択しない様にするためである。
- ロボット同士の衝突回避は移動を開始するタイミングをずらすことによっておこなう。
- ワークの搬送と加工は別々のロボットに行わせる。これは各ロボットの作業量を均一にするためである。

TYPE1とTYPE2の作業場の全体像を図5、6に示す。

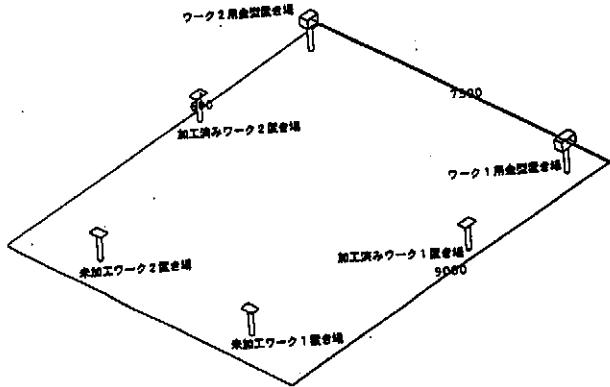


図5 TYPE1の作業場の全体像

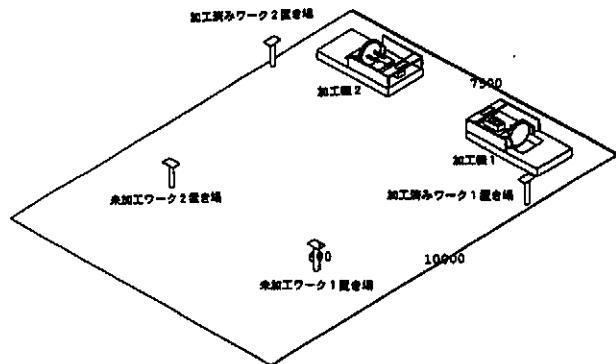


図6 TYPE2の作業場の全体像

2. 2シミュレーション方法

ワークは2種類とし、その形状を図7, 8に示す。

TYPE1ではそれぞれ7枚ずつ加工する。途中、それぞれの組から1台ずつ2台のロボットが異なるタイミングで故障する。

TYPE2ではそれぞれ10枚ずつ加工する。途中、加工機1, 2、ロボット1, 2の順で故障する。

以上の条件でシミュレーションする。

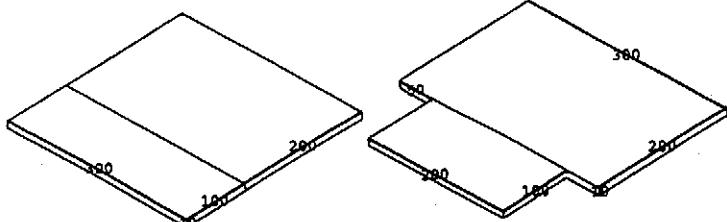


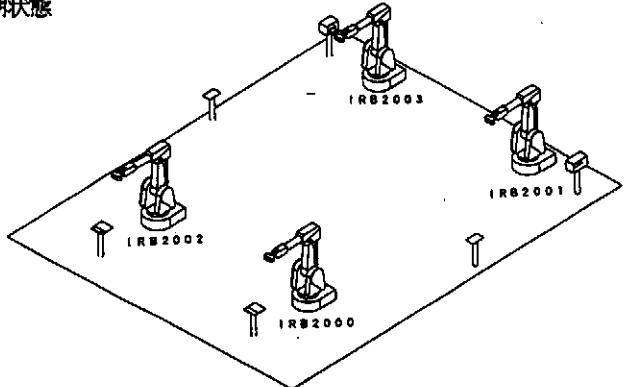
図7 ワーク1

3. 実験結果および結論

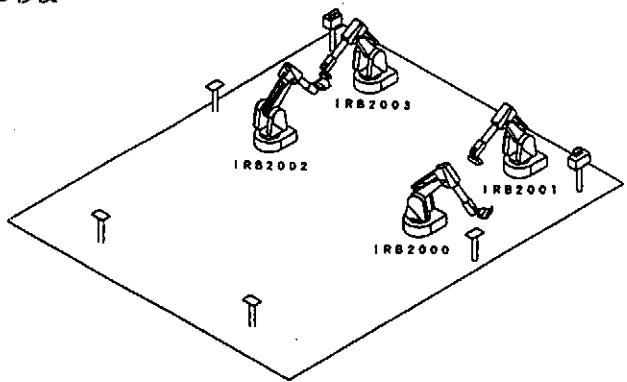
3. 1 シミュレーション結果

図9にTYPE1のシミュレーション結果を、図10にTYPE2のシミュレーション結果を示す。

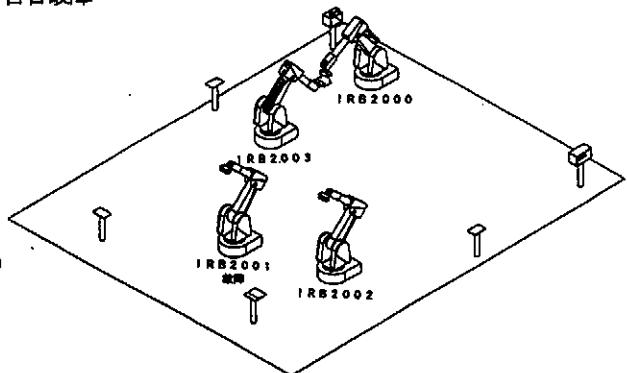
初期状態



4.3秒後



1台目故障



2台目故障

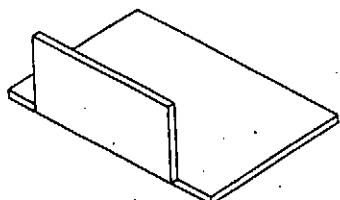
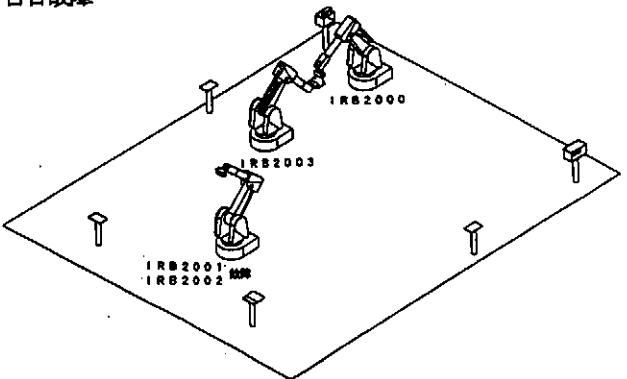


図8 ワーク2

図9 TYPE1のシミュレーション結果

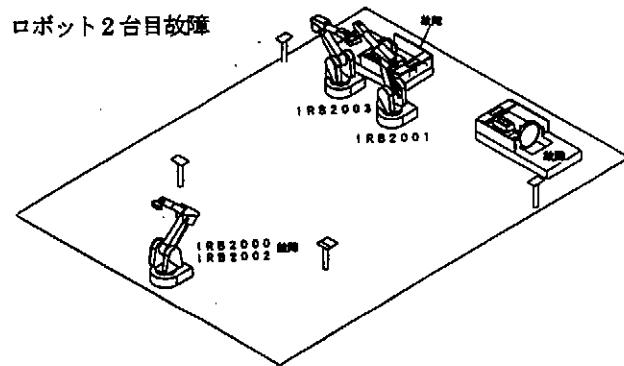
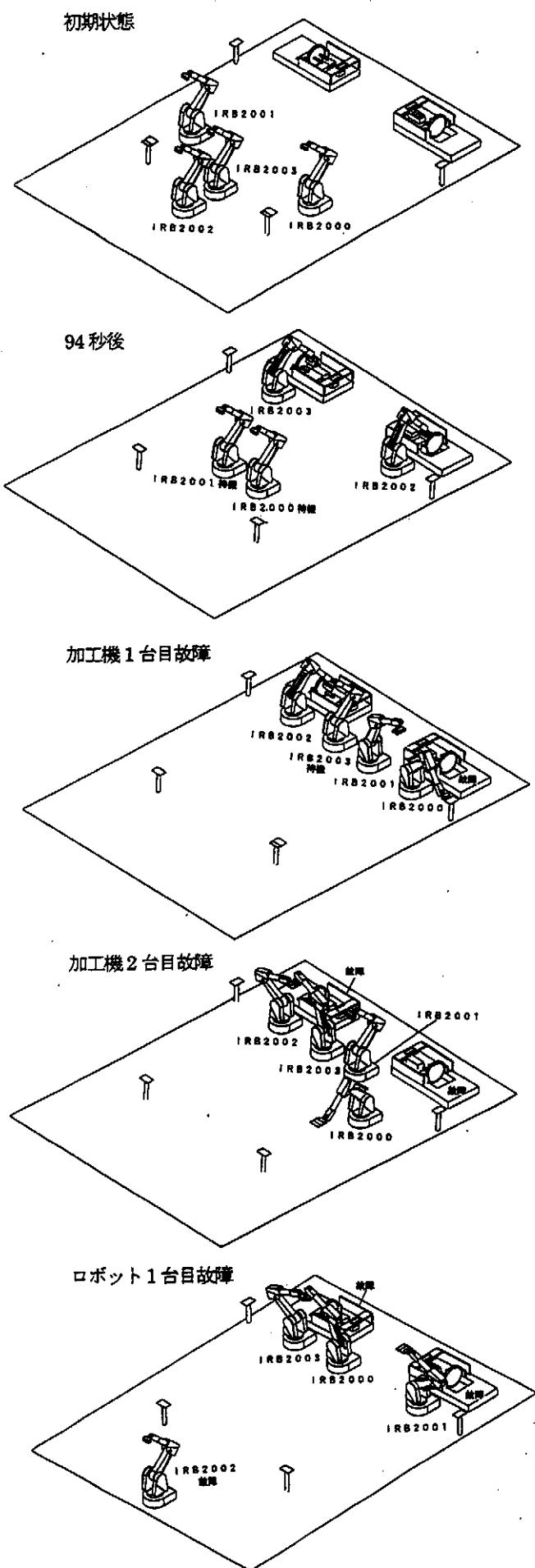


図 10 TYPE2 のシミュレーション結果

3. 2 結論と問題点

シミュレーションの結果、システムの一部が故障しても作業を続行し、目的を達成することが確かめられた。また以下のことが明らかになった。

- 高度に知能化されたロボットを用いなくても、各要素間の単純な信号のやりとりだけでシステムを構成できる。
- 比較的小規模なシステムならば、ろぼっとに特別な機構を持たせずに、移動を開始するタイミングと経路を決めてやるだけで衝突を回避できる。
- ワークに作業プログラムを持たせることができれば、ロボットや加工機のプログラムを変更する必要がなくなるのでさらなる柔軟性の向上につながる。

また、次のような問題点も明らかになった。

- より複雑なシステムになると本研究のような方法では完全に衝突を回避するのは難しい。
- 本研究では作業効率はあまり考えていない。
- 予測できない故障が起った時の対処の方法を考えていない。

しかし、このシステムにおいてはおよそ期待どおりの機能を持たせることができた。

参考文献

- 1) 長田 正：自律分散をめざすロボットシステム、(1995)、オーム社
- 2) 伊藤 正美：自律分散システムはいかにして構成されるか、(1990)、計測と制御、Vol. 29, No. 10, 877~881
- 3) 神余 浩夫、竹垣 盛一：協調分散制御システムアーキテクチャ：CODA の概念モデル、(1991)、計測自動制御学会論文集、Vol. 27, No. 4, 458~465,
- 4) 伊藤 高廣、進藤 章：二つのマニピュレータのためのスケジューリング、(1990)、計測自動制御学会論文集 Vol. 27, No. 4, 474~481