

人間との協同作業を特徴とする 遠隔作業ロボットシステム

佐藤知正* 松井俊浩* 平井成興*

A Telerobot System Featuring Man-robot Cooperative Task Execution

Tomomasa SATO Toshihiro MATSUI Shigeoki HIRAI

This paper proposes a new teleoperation system where man and robot cooperatively execute tasks. First, in the new system, the roles of man and robot in the generation of task procedures and task execution change dynamically. That is, while in the conventional supervisory system the role of man is limited to the generation of task procedures and that of the robot to task execution, in this new system man can participate in task execution and the robot can participate in the generation of task procedures.

Second, in the new system, the generation of task procedures and task execution are interspersed. That is, while in the supervisory system a task proceeds as batch process, in the newly proposed cooperative system it proceeds interactively.

MEISTER (Model Enhanced Intelligent and Skillful Teleoperational Robot) is an example of a cooperative task execution system which integrates a teleoperation oriented knowledge base, a cooperative maneuvering system, and a teleoperated-motion understanding system. A chemical experiment task demonstrates the effectiveness of man-robot cooperation.

Key Words : Intelligent Telerobot System, Man-Robot Cooperation, Multi-Level Man-Robot Interface, Task Knowledge-Base, Advanced Teleoperation.

1. はじめに

危険で劣悪な環境下での作業を人間に代わって行ってくれる遠隔作業ロボットシステムが求められている。これまでに提案されているシステムにおける人間とロボットとの関わり方はスーパーバイザリ方式をその基本概念としていた¹⁾。これは、ロボットの部分的な作業の自律実行機能を利用したもので、この手法に基づいた遠隔作業システムがいくつか報告されている^{1~6)}。このスーパーバイザリ方式は、本来は時間遅れがある状況での遠隔作業を念頭において提案されたものであり、そのような

状況において特に効果を発揮している^{1,4~6)}。しかしながら、この方式は、人間とロボットが持つそれぞれ得意な作業能力を十分に発揮させるという観点から眺めるに問題点を残している。

一方、実際に遠隔作業システムが必要とされる事例においては、例えば原子力プラント内作業のように、時間遅れが問題とならないものも多い。このような場合については、上記の問題点を解決しうる新しい遠隔作業形態が可能である。

本論文では遠隔作業における人間とロボットの望ましい関わり方として、協同作業という形態に着目し、この考えに基づいた遠隔作業ロボットシステム MEISTER (Model Enhanced Intelligent and Skillfull Teleope-

原稿受付 1991年3月29日

* 電子技術総合研究所

rational Robot) を示す。この MEISTER システムは、著者らがこれまでに発表してきた、遠隔作業ロボットの知識レベルの機能を支える教示を含んだ作業実行法⁷⁾、遠隔作業向き知識ベース⁸⁾、知識レベルの機能と動作レベルの介入の矛盾を解決する操縦行動理解システム⁹⁾等に、あらたに動作レベルで機能を支える協調操縦システムを加えて、人間とロボットとの協同作業という思想に基づいて統合したものである。本論文では、さらに構築された MEISTER システムによる一連の化学作業実験のなかで協同作業の模様を実際的に示す。

2. 協同作業とは

従来のスーパーバイザリ方式においては、以下のような単位を繰り返し行うことで作業を遂行するのが基本となっている。

人間は、作業の部分部分について遠隔のロボットが自動的に実行できるような作業手順を作成し、それをロボットに送る。ロボットはその手順に従って部分作業を自動的に実行し、実行終了後次の作業命令を待つ。

このように、スーパーバイザリ方式においては、人間は専ら作業手順の作成という知識のレベルだけ、ロボットはその実行という動作のレベルだけというように役割分担が階層的に固定されており、また、作業手順の作成と実施はバッチ処理のように切り離されて進行していた。

階層的に固定することの欠点は、動作のレベルでも人間の得意な能力があるし、知識のレベルでもロボットの得意な能力があるにもかかわらず、それそれをうまく利用できないことである。例えば、作業の実施状況にあわせて動作を細かく修正することを全てロボットに行わせるのは現状のロボット技術ではきわめて難しく、一方、作業計画に必要な環境データをもれなくチェックしたり、逐一管理することを人間が間違いなく行うことは難しい。前者については人間が行わざるを得ないであろうし、後者についてはロボットが行うのが適当であろう。

バッチ処理のように進行することの欠点は、作業手順作成時に実行環境情報が十分に反映しきれなかったり、作業の実行中にプログラム動作の細かな内容が変更できないことである。

一方、時間遅れが問題とならない場合については、新しい遠隔作業形態が可能である。具体的には、作業内容、環境、状況、進行状態等に応じて人間とロボットの役割層を動的に変更したり、場合によっては両者が同時に同じ役割層で協力し、さらに、作業手順作成時に実行環境情報を十分に反映し、作業の実行中にもプログラム動作の細かな内容を変更するなどのことが可能な作業形態で

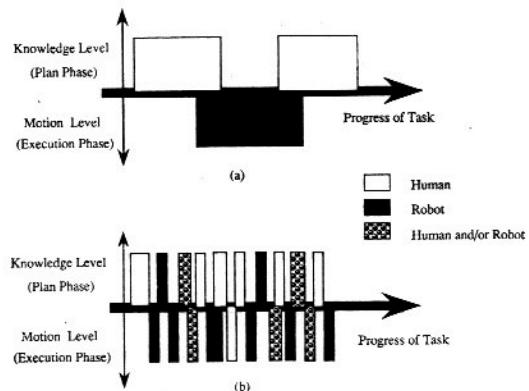


Fig. 1 Comparison of (a) supervisory and (b) cooperative task execution

ある。この作業形態の重要な特徴は、人間とロボットがそれぞれの得意な要素を臨機応変に出し合って作業を遂行できる点であり、このような作業形態を本論文では「協同作業」と呼ぶことにする。

従来の作業形態とこの新しい作業形態との違いのイメージを Fig. 1 に示す。従来のものでは図の (a) のように、人間とロボットの役割が階層的にはっきりと分かれ、かつ固定されており、作業の進行も手順生成のフェーズと実行のフェーズに大きくわかれバッチ処理的になっている。

これに対し本論文で提案する形態は図 (b) のように、人間とロボットの役割がどちらの層にもあること、作業手順の生成と実行のフェーズが細かく分割されている点が特徴である。この新しい形態によって上に述べた従来方式のいくつかの欠点が取り除かれる。

階層的に固定することの欠点を取り除くとは、具体的には、人間とロボットそれぞれ得意な能力を階層をまたいで出しあえることである。すなわち人間とロボットとが同じ層で場合によっては同時並行的に活動でき、これにより上で述べた意味での協同作業が可能になる。

バッチ処理的に作業を進行することの欠点を取り除くとは、作業手順の生成と実行のフェーズの分割を細かくすることによって作業結果に基づいたお互いの行動の密な修正が可能となり、協同作業に不可欠なインタラクティブな構造が確保されることである。

本論文では、1) 人間とロボットの相互アクセスを可能とした知識ベースとその内容を人間に分かりやすく伝えるマンマシンインターフェース、2) 人間とロボットの動作が重畠できる協調操縦システム、3) 両者の併用を支える操縦行動理解システム、によって上記の形態が実現可能となることを示す。

3. 協同作業の要素

本章では、遠隔作業において人間とロボットそれぞれが得意と考えられる要素の分析を行う。以下ではそれを知識のレベルと動作のレベルに分けて考察する。

3.1 知識のレベルでの協同の要素

作業手順の生成には、遠隔作業の対象となる物体の操作の仕方や、それら物体の位置データなどの知識が必要である。

データに関しては、作業に必要なデータの有無のチェック、データの獲得・蓄積などが遠隔作業ロボットには不可欠である。このうち、データを獲得するための物体認識は人間が得意で、獲得されたデータの蓄積・検索はロボットが得意である。

一方、作業手順の生成に関しては、作業目的の設定は人間にしかできない。それを保持しているデータにあわせて展開することはロボットが得意である。作業の実行に関しては、計画通りの動作を行うのはロボットが得意である。一方、実行された動作と実環境の整合性の確認は人間が得意である。

これらが遠隔作業の実施において効果的な知識レベルの協同の要素と考えられる。著者らは、このような協同の要素を取り入れたインタラクティブな遠隔作業実行を可能とする遠隔作業向き知識ベースを構築した。詳細は第4章で述べる。

3.2 動作のレベルでの協同の要素

作業はハンドの動作によって実現される。直線動作や姿勢を保つ動作など自由空間での簡単な幾何学的な拘束にならうのはロボットが得意である。これは、ロボットの動作生成が簡単な幾何学的形状をもとにした記述だからである。一方部分的な、あるいは、不完全な拘束条件

だけから動作を生成するようなことはロボットは不得意であり、むしろ人間の得意とするものである。例えば、人間はすばやく環境状況を把握できるので、環境状況にならった動作を容易に行える。これらが遠隔作業の実施において効果的な動作レベルの協同の要素と考えられる。

両者の動作を作業状況に合わせて組み合わせて利用できるようになると動作レベルでの協同が可能になる。このため、ロボットの動作と人間の操縦動作とを実時間で重畳できる協調操縦システムを実現した。詳細は5章でのべる。

3.3 作業進行に伴うロボットと人間の理解の一貫

このような協同ができるためには人間とロボットでの作業進行状況の理解が一致していることが不可欠である。

例えば、ロボットが行おうとしていること、行ったことを人間に理解しやすい形式で示す手段が求められる。そのようなものとしてモデルデータの内容を提示し、作業の進行を説明できる映像表示および音声出力装置を結合した。

また、人間が動作レベル、すなわち直接操縦で行った作業の内容をロボットのもつモデルデータに正しく反映させる手段も求められる。そのようなものとして操縦行動理解機能⁹⁾を実現した。詳細は第6章で述べる。

4. 遠隔作業向き知識ベース

MEISTERの知識ベースは、遠隔作業における信頼性の高い作業の実行法や上手な道具の使い方、簡易な物体の教示法などの知識を対象指向の考え方を利用して物体ごとに記述し、それらをFig. 2のような階層のもとで集積したものである⁸⁾。一般物体のモデル(General Handling Model)は、物体をpick-and-place操作の観点から抽象化したもので、その作業に関連したデータとして、物体基準座標(base-frame)とか接近位置(approach pose)など、関連した手続きとして、接近動作(approach)、把持動作(grip)、pick-and-place作業に必要なデータの教示手続き、などが格納されている。一方、アルコールランプなどの特殊物体のモデル(specific model)には、例えは芯の先端位置(wick-point)や点火手続き(light)、ランプのデータの教示手続き、などランプに固有な知識が格納されている。これらの物体モデルには3.1で述べたような協同の要素を取り入れたインタラクティブな協同作業を可能にするため、以下の3つの機能が組み込まれている。

- 1) 教示を含んだ作業実行法 作業の実行過程でモデルデータの有無のチェックが行なわれ、データがない場

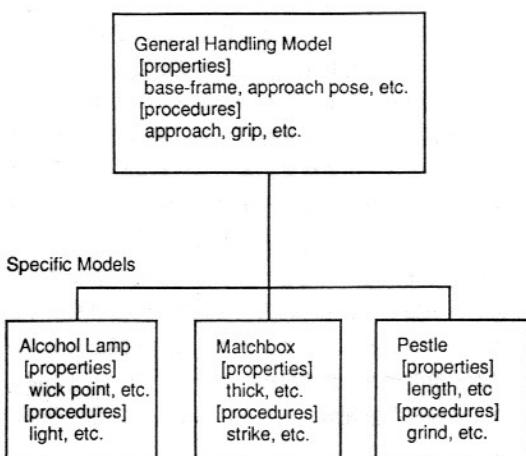


Fig. 2 Hierarchy of object models.

合にもエラーとせず、教示手続きが自動的に起動されるようにした。この手法の効果は以下の通りである。

a) 新しい物体を対象とする作業でもモデルデータの有無を気にせずに作業がかかる。そして、作業を通して教示されたデータを用いることで、次回からは自律的に作業が進められるようになる。

b) データ獲得に適した教示手順を利用できる。作業動作そのものが直接教示には適しないような場合には、その部分だけ専用の教示手順を物体モデルに定義しておけば、作業全体の流れをさほど阻害せずに作業の教示と実行が行える。

2) 動作単位毎の確認を含んだ作業実行法 ロボットが展開した作業動作を実行前に表示し、操作者に実環境との整合性を確認させ、承認を得た上でそれを実行するようにした。また、予定された動作とのずれが残った場合にも、直ちに協調操縦システムを通じてその補正ができるようにした。これにより、以下の効果が得られる。

a) 異常なデータのまま実行して環境を破壊するようなことが未然に防げる。

b) マニピュレータの動作の誤差、環境の不測の変化に対し、操作者が臨機応変な対応をとることが可能になり、作業実行全体の信頼性が高くなる。

3) 知識ベースの直観的な提示手段 上記のような協同作業を取り入れた教示法や実行法においては、ロボットが持っている知識の内容を人間の直観と一致するレベルで伝えることが重要である。具体的には、操作者への教示法の指示や動作実行の確認を円滑に行うための画像と音声出力を利用したインターフェース機能を組み込んだ¹⁰⁾。

a) 教示の過程では、モデルデータ獲得のために実現

すべきマニピュレータの適切な位置・姿勢が画像で示される。動作の確認に於いては、モデルのデータから次の動作の結果予測される状態がシミュレートして表示される。これらにより操作者は教示や動作の確認を直観的に行える。

b) 上記の画像による表示に加え、特に操作者の注意を喚起すべき作業ポイントにおいては、操作者への要求事項や作業動作の予告を音声でも伝えるようにした。利用する音声メッセージは、あらかじめ人間が読み上げ、音声サンプリング機能により蓄えておく。音声メッセージの利用により、操作者はコンソールディスプレイ上の文字を読む必要がなくなり、作業の監視に専念できる。作業に関する知識の具体的な形の参考として付録Ⅰに物体モデル記述の概略例と説明を示す。

4. 協調操縦システム

協調操縦システムは、ロボットの自動動作と人間の操縦動作の実時間的な重畠を可能とし、それにより素手では難しいような作業動作を作り出すことで、動作レベルでの協同を可能にするものである。

4.1 機能

Fig. 3に、機能構成を示す。本システムで利用可能な機能は、以下のものである。

1) 速度制御 ティーチペンドントのボタンを押すと、スレーブの手先が、設定されている座標系の指定された軸について一定の速度で動く。この機能は、図(a)のような直線動作に適している。

2) 増分型制御 スレーブマニピュレータの、 x 或は y, z 位置が所定のキーを押す毎に一定量ずつ増減する。回転角についても同様である。図の(b)のような精密

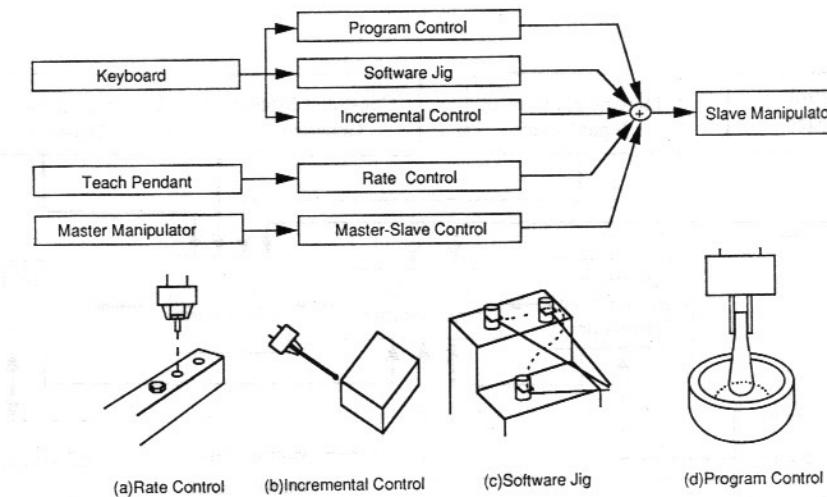


Fig. 3 Block diagram of the Cooperative Control System.

な位置あわせに適している。

3) ソフトウェア治具¹¹⁾ ソフトウェア治具は、従来のマスタスレーブ制御の上に重畠される空間的な動作拘束である。金物の治具のような効果があり、しかもソフトウェアのパッケージで記述できる。ソフトウェア治具を記述するための言語も開発されている。

図の(c)のように水の満たされたグラスを運ぶ作業を考えてみよう。操作者はグラスの姿勢を垂直に保つことに注意を注がなければならない。ソフトウェア治具は、このような拘束動作を自動的に実現し、操作者は、グラスの移動だけに専念できるようにするものである。

協調操縦システムの特徴は、比率制御、増分型制御、マスタスレーブ制御が重畠される点である。これにより、操作者が個々の機能を利用する場合に、モード切り換え操作の煩わしさがなく、いつでも自由に任意の機能を利用できる。

さらに、協調操縦システムでは次のような作業動作を作り出す機能も備えている。

4) プログラム重畠 スレーブがプログラムで制御されているときでも、操作者がマスタを動かせばスレーブが対応して変化するようにしたものである。これにより、プログラム動作中に生じた簡単な不具合ならば、臨時に修正して続行させることができる。たとえば、図の(d)のようにロボットが自動動作でりこぎ作業をしているとき、大きな塊が見つかった場合操作者が臨時にマスタを操作してそれを碎かせることができる。

5) 偏差自動解消 マスタスレーブ制御が起動される時点での、両者の相対的な対応付けが行われ両者の座標の偏差がマスタ/スレーブの機械的な運動を行うことなく、自動的に解消される。これにより、以下の効果が得られる。

a) マスタとスレーブの不整合から生じる急激な動きが防止される。すなわち、この機能があると、マスタとスレーブはいつでも結合・切り離しが可能であり、人間は必要なときにすぐにマスタスレーブ制御ができるし、不要なときにはいつでもマスタから手を離して別の操作を行うことができる。

b) マスタとスレーブそれぞれ作業に適した原点を独立に設定できる。これは例えば、次のように利用される。スレーブを長い距離移動させるばあい、マスタや人間の可動範囲の制約などから、対応する距離マスタを一気に動かすのは難しいことがある。本機能を利用すると、これは次のようにして行える。マスタスレーブ制御を起動して適当な距離マスタを動かしたところで、一旦マスタとスレーブを切り放し、マスタだけを操作しやすい位置に戻す。この間、スレーブはマスタと切り離された位置で静止している。そこから再びマスタスレーブ制御を起動してマスタを動かすとその分、スレーブはさらに目標に向かって移動する。これを繰り返すことで、人間は自分にとって楽な大きさの動作でスレーブを長距離移動させることができる。

4.2 メカニズム

本システムの構成を Fig. 4 に示す。関節角→制御座標変換 (Joint Angle/Control Frame Conversion) はマスタの各運動自由度の値から仮想の制御座標系におけるマスタの手先の位置、姿勢を生成する。生成された結果はマスタとスレーブの偏差を解消するための相対変換 (Master-Slave Transformation) を施した後、制御重畠部 (Superposing Mechanism) に送られる。制御重畠部は各指令入力装置から送られてくる信号を重畠してスレーブが実現すべき状態を生成する。この信号は、制御座標→関節角変換 (Control Frame/Joint Angle Con-

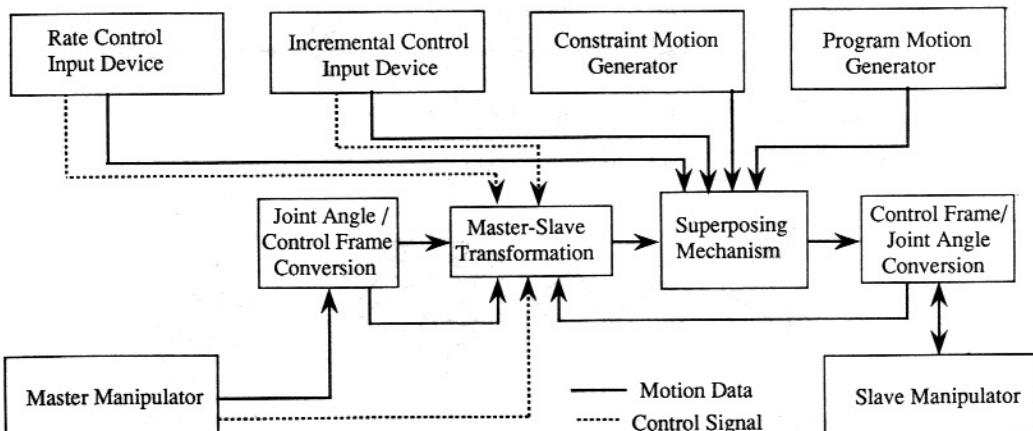


Fig. 4 Mechanism of Cooperative Maneuvering System

version) へと送られ、その結果、スレーブの手先は所定の運動を行い、所定の位置と姿勢あるいはトルクを生成する。4.1 で説明した個々の機能は以下のようにして実現されている。

1) 速度制御の重量 速度制御指令装置 (Rate Control Input Device) の各スイッチからの入力信号が続く間、制御座標の対応する自由度の値を一定の速度で増減しつづける。また、これと同時にマスタとスレーブの相対変換を更新する (図の破線の指令)。

2) 増分型制御の重量 増分型制御指令装置 (Incremental Control Input Device) の各自由度の増減を指示するスイッチの一回の操作について、制御座標の対応する自由度の値を一定の量ずつ増減させる。また、これと同時にマスタとスレーブの相対変換を更新する (図の破線の指令)。

3) ソフトウェア治具 指定された拘束動作に応じて、拘束動作生成部 (Constraint Motion Generator) が拘束すべき座標値を生成し、制御座標の対応する要素をそ

れで置き換える。

4) プログラム重畠 プログラム制御方策が使用されているときは、スレーブはプログラム動作生成部 (Program Motion Generator) で生成された軌道点に沿って動くように制御される。その制御サイクルの中で、マスターの位置の変化もチェックされる。もし、操作者がマスターを動かして、その前回の制御サイクルからの変化量が設定された値を越えた場合、自動的にマスタスレーブ制御に入り、スレーブは軌道から外れてマスターの動きに追従するようになる。操作者がマスターの操作をやめてマスターの動きが設定値未満になると、スレーブはその現在位置から軌道上の一一番近い点に復帰し、そこから再びプログラム動作を続行する。

5) 相対対応付 マスタスレーブ制御を開始する時、まずマスターとスレーブそれぞれの位置・姿勢を取り込み、それから両者の間の座標変換データを生成する (図の破線の指令)。以降、マスタスレーブ制御が続けられている間は、マスターからの信号はこの座標変換を施した上でスレーブに伝える。

以上の各制御方策の優先順位は、ソフトウェア治具、速度制御と増分型制御、マスタスレーブ制御の順になってしまっており、プログラム重畠はこれらとは独立に実現した。

5. モルデータの自動管理

協同作業においては、知識のレベルと動作のレベルの双方に於て人間の介入が行われる。一方、ロボットの自律作業が正しく働くためには、モデルに格納されているデータが実環境と一致している必要がある。人間が知識のレベルから指令を与え対象物体が移動する場合は、それに対応してモデルデータを更新する手順が物体モデルには備えられており、問題は生じない。ところが、人間

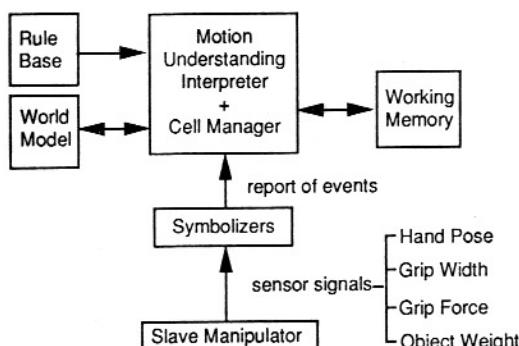


Fig. 5 Block diagram of the Motion Understanding System.

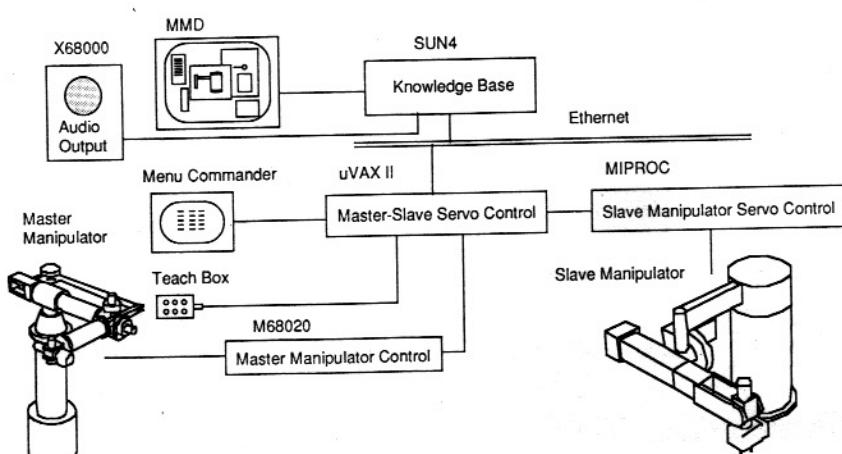


Fig. 6 Structure of Experimental System.

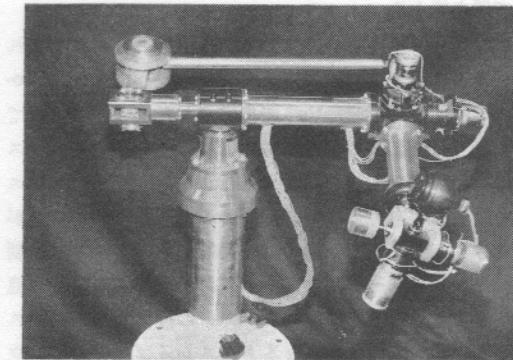


Fig. 7 Direct drive master manipulator.

が直接操縦で対象物体を動かすなど、動作レベルから介入を行ったときには、それにより生じた環境の変化はモデルのデータには反映されない。したがって、動作と知識、両レベルでの協同作業を矛盾なく両立させて利用するためには、このような動作レベルの介入時にモデルデータの更新を行う機能が必要である。これは、動作レベルと知識レベルの協同作業を統合するための重要なインフラストラクチャの一つである。このようなものとして、MEISTER では人間の介入操縦動作の内容を知識ベース内の物体モデルのデータの管理をするレベルで理解して、対応する管理を自動的に行なう操縦行動理解システムを備えている⁹⁾。

操縦行動理解システムの構成は Fig. 5 のようになっている。遠隔のロボットからは手先の位置、指の開閉、指に加わる力の変化を事象として送られている。インタプリタはその事象とモデルの状態を行動理解ルールに照らし合わせて、どの物体に対しどのような操作がなされたのかを解釈し、その結果に応じてモデルデータの変更を行なう。これにより、人間が動作レベルで直接介入して対象物体の位置の変化を引き起こした後でも、ロボットの自律作業実行が速やかに利用できるようになる。行動理解ルールのいくつかの例と説明を付録 II に示す。

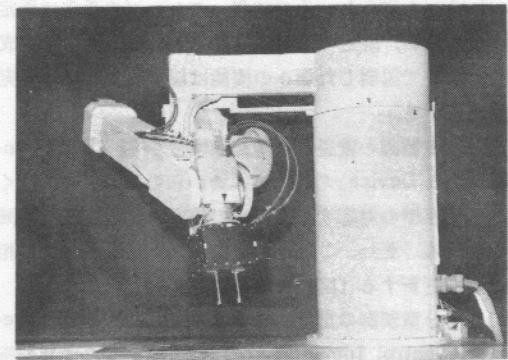


Fig. 8 ETA-II manipulator (slave manipulator).

6. 実験システム

実験システムの構成を Fig. 6 に示す。マスタマニピュレータは 6 軸のダイレクトドライブマスタマニピュレータ (Fig. 7)¹²⁾ である。メニュー命令¹¹⁾は CRT 上に表示されたメニューをキーボードで選択することにより、増分型制御の指令入力やソフトウェア治具の設定を行う。ティーチボックスは速度制御指令の入力をを行う。スレーブマニピュレータには 6 軸のダイレクトドライブマニピュレータ ETA-II (Fig. 8) を用いている。これらの装置がマイクロ VAX-II に結合されて協調操縦システムが構成されている。

MMD (マルチメディアディスプレイ) は、実画像、ソリッドモデル、ワイヤーフレームモデル、テキストなどのマルチメディアの情報を複数のウィンドウで高速に表示することのできる装置である¹³⁾。また、実画像とワイヤーフレーム像を重ねたり、ステレオ視もできるようになっている。

X6800 は、音声メッセージ用の音声サンプリングと音声出力を行う装置である¹⁴⁾。

物体モデルを始めとする上位のソフトウェアは、幾何モデルを備えたオブジェクト指向型 Lisp である Eus Lisp¹⁵⁾ で記述した。これらは Sun 3 および Sun 4 ワークステーションで走り、マイクロ VAX II とは Ethernet により交信を行なっている。

7. 作業実験

実験作業として原子力プラントにあるホットセルでの作業を想定し、炎色反応実験を取り上げた。作業環境を Fig. 9 に示す。基本的な作業手順は、乳鉢と乳棒を用いて試料を粉砕し、マッチを擦り、アルコールランプに点火し、そして試料の炎色反応を見るというものである。Fig. 10 は実験の模様のスナップである。以下、部分作業を動作レベルの協同作業と知識レベルの協同作業

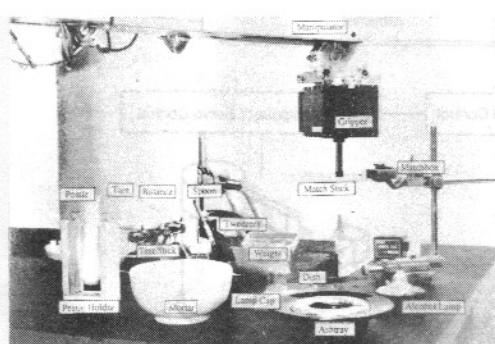


Fig. 9 Setup for the flame reaction test experiment.

に分けて説明する。

7.1 動作レベルでの協同作業

1) すりこぎ作業 この作業は具体的にはFig. 10(a)のようだ、乳鉢の中で乳棒を円周軌道にそって動かすことを行なう。このような繰り返し作業は、プログラムによる作業実施が向いている。しかしながらプログラム動作は、同じことを繰り返すだけで、臨機応変な動作の変更はできない。例えば、大きな塊が発見された場合、從

来のプログラム制御では、臨時にそれを集中的に粉碎しようとしてもできない。協調操縦システムにおいては、プログラム重畳という機能によりこれが可能となっている。すなわち、操作者がマスターを動かすとスレーブはプログラム動作を一時停止し、操縦動作に従うのでスレーブにその塊を直ちに碎かせることができる。

2) 炎色反応の観察 この作業では、ロボットは、試料の付着したテスト棒の先端をランプの真上、炎がある

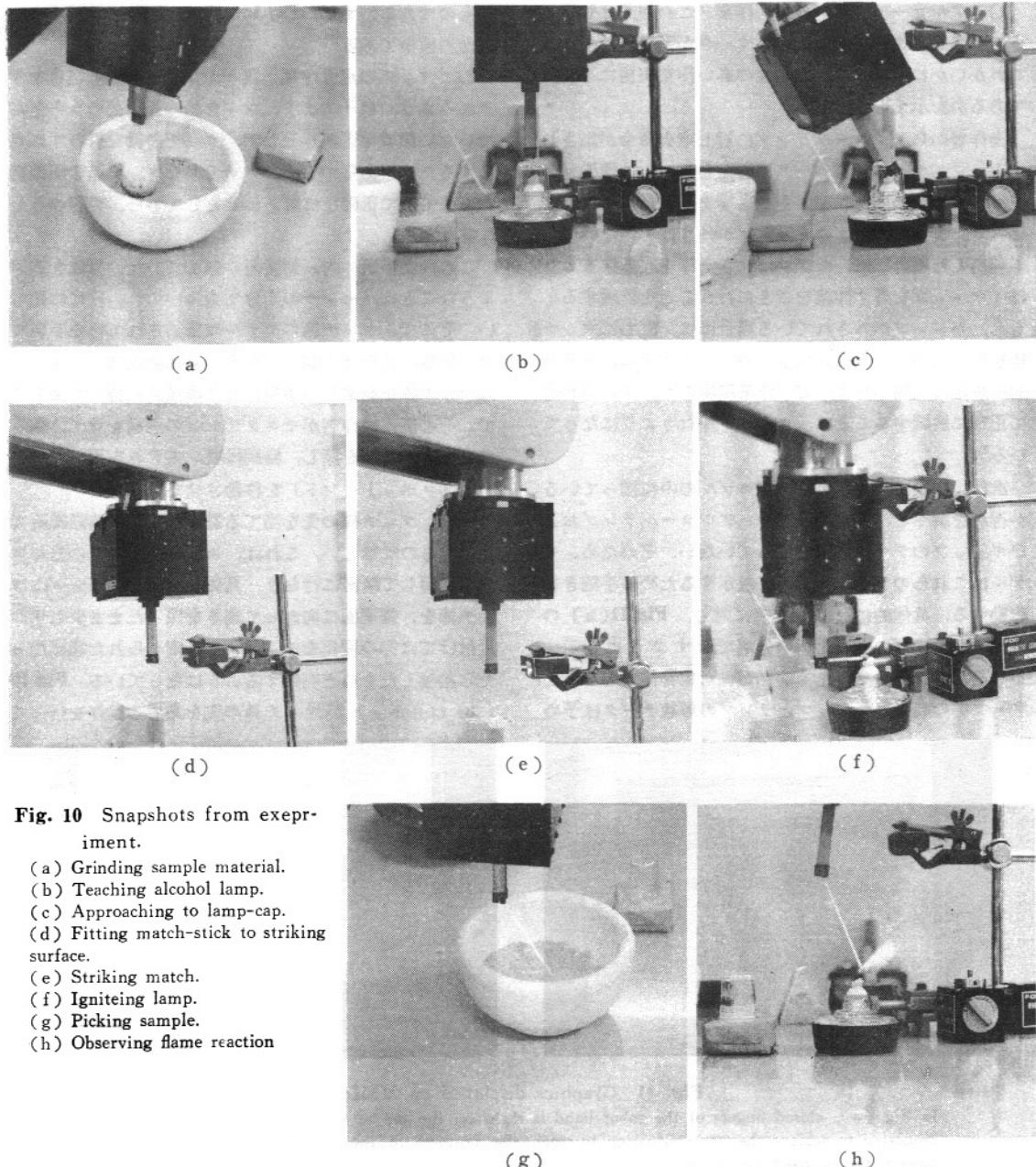


Fig. 10 Snapshots from experiment.

- (a) Grinding sample material.
- (b) Teaching alcohol lamp.
- (c) Approaching to lamp-cap.
- (d) Fitting match-stick to striking surface.
- (e) Striking match.
- (f) Igniting lamp.
- (g) Picking sample.
- (h) Observing flame reaction

と予想される位置に持つて来る。しかしながら、実際に風のため炎は揺らいでおり、テスト棒の先端の試料が炎に入らないことが多い。このような時に、操作者はマスタマニピュレータを操作して棒の先端を炎内に誘導し、反応を十分に観察できる。Fig.10(h)にその模様を示す。

これらは、協調操縦システムによって実現された人間とロボットの動作のレベルでの協同作業の実例である。

7.2 知識レベルでの協同作業

1) アルコールランプの点火作業 この作業は知識ベースにさまざまな作業実施のノウハウが蓄積されており、これがいかに即応的で、信頼性の高い作業実施に不可欠であるかを示すものである。

操作者からアルコールランプに対し点火命令が出されると、システムは知識ベースに蓄えられている情報(ランプのモデルの内部状態)をチェックしに行く。これによって、例えばランプのキャップが外されていないことが判明した場合には、モデルはマッチ棒を取る作業よりは、キャップを外す作業をさきに行うことを計画する。もし、キャップが外されている場合には、直ちにマッチ棒を取りにくくよう計画される。このように、作業目的の設定は人が行う一方、作業環境のデータにあわせて正確に展開することは、ロボットが行うようになっていいる⁸⁾。

次に、作業に必要なデータがモデルの中に揃っているか否かを調べる。実験例の場合、アルコールランプおよびキャップのデータはまだ揃っていない。そのため、モデルはこれらのデータを容易に教示するための手続きを起動する。具体的には、操作者に対し、Fig.11(a)のようにキャップの中心を指の先端で示すように指示する。これは、ランプとキャップが軸対象物体であること、テーブルの高さ、キャップとランプの形状データは予め

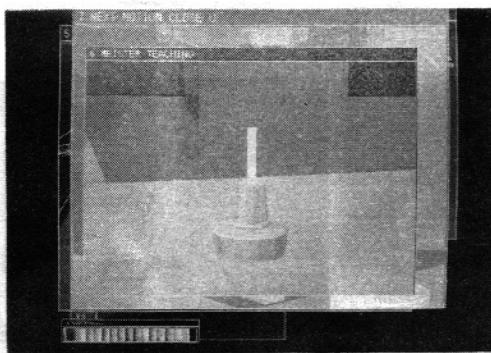
モデルに起述されていることをを利用して教示を容易なものとしている。Fig.10(b)は実際にロボットの指を指示された位置に持ってきたところである。このようにしてデータの教示が完了すると、モデルは獲得したデータを利用してさっそくキャップ外しの作業にかかる。

今度はデータが揃っている場合になるので、ロボットは動作の実行に先立ち、生成した動作のシミュレーションをMMD上に示し、操作者へ確認を求める。Fig.11(b)はキャップにアプローチするための動作を表示したものである。また、Fig.10(c)は実際のロボットの動作の模様である。

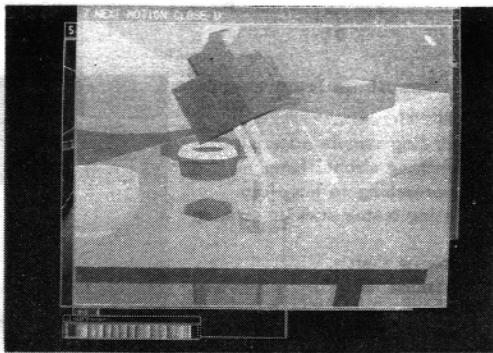
2) マッチの点火作業 我々がマッチを擦る場合を考えてみよう。我々はまず、マッチ箱の側薬面のうち傷んでいない領域を選ぶ。そして、マッチの軸を折らずに点火するため、あらかじめマッチの頭をマッチ箱の側薬面にあててみて正確な位置あわせをしてから、実際の点火作業に移る。

この作業手順の内、側薬面の傷んでいない領域を選ぶことは、センサの十分でないロボットには難しい。そこで、マッチ箱のモデルに格納されている手順では、領域の選択は人が行なう。具体的には、ロボットはマッチ棒を側薬面の中央付近に機械的に持つて来るの、その後、操作者がモニタテレビの画面を見て、傷みの少ないところを探し、協調操縦システムを用いてマッチ棒(ロボットハンド)を移動させてやる。

一方、マッチ棒の先を当てる動作を人が遠隔操縦で実現するのは難しい。これは、ロボットが指先の力センサを利用して精密に行なう。具体的には、ロボットはマッチ棒を、側薬面に向かって高さを保ったまま少しづつ近付けては力の変化をチェックし、定められた値になつたら接触したものと判断するようになっている。Fig.10(d)はロボットがマッチ棒の先をあてる動作を行って



(a)



(b)

Fig. 11 Graphics displayed on MMD.

In Fig. (a), closed fingers of the robot hand is right on the top of the lamp-cap. The operator is requested to achieve the same situation to teach the position of the cap. Fig. (b) shows expected motion for approaching to cap.

いるところ、(e)は、獲得したデータを利用してマッチを擦ったところである。

これらが、人間とロボットの知識のレベルでの協同作業の実例である。以上の作業を通じて、MMDによって、教示すべき対象のどの点をポイントしたら良いのか等がグラフィクスで表示され、指示文が音声で出力される。これにより、ロボットから操作者へのメッセージ伝達でも直観性の向上が実現され、知識レベルの協同を支えている。

8. 結 論

本論文では遠隔作業における人間とロボットの望ましい関わり方として協同作業という形態を提案した。

この協同作業方式においては、作業手順の生成とその実行に関し、人間とロボットの役割が動的に変化するのが第1の特徴である。すなわち従来は、人間が作業手順を作成しロボットがそれを実行するという固定的形態をとっていたが、協同作業では人間も作業実行に参画し、またロボットも作業手順作成に関与できる。作業手順の作成と実行が有機的に結合されているのが第2の特徴である。すなわち、従来のスーパーバイザリ方式においてはバッチ処理であったのが、本論文の協同作業ではインターラクティブ処理的になっている。

このような人間とロボットとの協同作業を可能にするための作業システムとして、遠隔作業向き知識ベース、協調操縦システム、操縦行動理解システムを統合した知的遠隔作業ロボット MEISTER を構築した。そして、一連の化学作業実験のなかで協同作業の効果を実際的に示した。

一方、時間遅れのある系にも本手法がどの程度まで対応しうるか検討することは将来の課題である。また、プログラム重畠については、本論文ではその基本概念を示し、限定された作業を扱うアルゴリズムを示した。これをより一般的な作業に適用できるように拡張することは、実用的な見地からも重要性が高い。

謝 辞

本研究を行うにあたり、ご支援いただきました元知能システム部長の弓場敏嗣氏（現、情報アーキテクチャ部長）、元自律システム研究室長の柿倉正義氏（現、東京電機大学）に感謝いたします。また、貴重な御意見をお寄せ頂きました元行動知能研究室長の高瀬国克氏および、同研究室の末広尚士氏をはじめ、日頃御討論頂く知能システム部の皆様に感謝致します。

付 錄 I 教示の実行法の記述例

1) 一般物体での例：approach の内容概略

```
(defmethod cobject
  (:approach (& key put as)
    (if (send c-hand :gripping-obj)           ①
        (send self :approach 1 put as)          ②
        (send self :approach 0)                 ③
      )
    )
  ① 手の中に物体があるかを検査する。
  ② あれば、さらに approach 1 を呼び出す。
  ③ なければ、さらに approach 0(下記)を呼び出す。
(defmethod cobject
  (:approach 0 ( )
    (cond ((and (setq curpt (send self :app))           ①
                (or ($ex-check-av.....)                      ②
                    (setq app.trn nil)))                     ③
               (send c-hand :mvc-dir curpt))              ④
              (t (setq curpt (teach-go.....)))           ⑤
                  (send self :moved curpt '.app)          ⑥
                  (send c-hand :psmvc-dir curpt)))
    curpt)
  ① モデルにアプローチ点が格納されているかを探す。
  ② もしあれば操作者に予定動作を提示し、同意を得る。
  ③ 同意がえられなければ格納されていたデータを消去する。
  ④ 同意が得られたときは動作を実行する。
  ⑤ アプローチ点が格納されていないので教示手続きを起動する。
  ⑥ 教示されたデータからアプローチ点を計算して格納する。
2) 個別物体での例：ランプの点火手順でのキャップのチェック
(defmethod lamp
  (:light (.....))
  .....
  (setq cap (find-such 1.....))           ②
  (cond (cap (send cap :remove)))         ③
  .....
  ); light end
)
① 点火手続きである。
② 接続されている物体からキャップを探す。
③ キャップが被っているときは外す作業 (:remove)を起動する。
3) 個別物体の例：ランプキャップ外し作業と教示手順
```

の内容概略

```
(defmethod lamp-cap
```

```
  (:remove ( ) ..... ①
    (or (send self :frm) (send self :teach)) ②
    (send self :pick) (send self :place ch 1))) ③
)
```

- ① キャップを外す作業である.
- ② 自分の位置データが確定しているか検査し、必要な
ら教示手順(下記)を起動する.
- ③ 外す手順そのものは一般物体のものが利用される.

```
(defmethod lamp-cap
```

```
  (:teach ( ) ..... ①
    ..... ②
    (jig-reset-vertical) ..... ③
    (setq cx
      (send self :teach-point-av
        'cap-top.....)) ..... ④
    (jig-reset-vertical) ..... ⑤
    (setq cap-top
      (list... (+*gtableheight*....))) ..... ⑥
    (send self :frm (wbse...cap-top) 'fill-in)
    .....))
)
```

- ① ランプキャップ固有の教示手順である.
- ② 教示しやすいように指の姿勢を垂直に拘束する.
- ③ キャップの頂点の位置を教示するよう要求する. こ
の手続きは一般物体のものである.
- ④ 教示終了後、姿勢拘束を解除する.
- ⑤ ロボットの指先位置のデータからキャップの頂点の
位置を計算する. テーブルの高さがデフォルトとな
っている.
- ⑥ キャップの位置を記入する. 再帰的にランプの位置
も計算、記入される.

付録 II 行動解釈ルールの例

```
(defrule approach-detect
```

```
! (in-cell ~object *x)
!(hand-empty)
(hand-frm ~position
  {*y ?neq nil :within-approach *x})
->
(renew approach-done ~object *x)
(send* *x :moved *y 'app))
)
```

ルールの意味

注目している領域の中に物体 *x がある、

手が空である、

手の現在位置 *y が物体 *x のアプローチ領域内であ
る、

ならば

“物体 *x への接近がなされた”という事実を更新せよ、
物体 *x へ手が *y の位置にアプローチをしたと知らせ
よ.

```
(defrule retract-approach
```

```
!(approach-done ~object *x)
(hand-frm ~position
  {*x ?neq nil :outof-apdroach *x})
->
```

(p-remove 1)
)

ルールの意味

物体 *x へのアプローチしている、

手の位置 *y が物体 *x のアプローチ領域のそとにでた、
ならば、

1番目の事実を抹消せよ.

```
(defrule move-to-grip
```

```
!(in-cell ~object *x)
!(hand-empty)
(hand.frn ~position
  {*y ?neq nil :within-grip-pnt *x})
->
```

(renew move-to-grip-done ~object *x)

(send* *x :moved *y 'm-t-g)
)

ルールの意味

注目している領域の中に物体 *x がある、

手が空である、

ならば

“手が *x の把持位置にある”という事実を更新せよ、

物体 *x へ手が *y の位置で把持に来たと知らせよ.

```
(defrule grip-detect
```

```
(hand-empty)
!(move-to-grip-done ~object *x)
(finger-wid ~direction closed
  -width {*y :within-grip-width *x})
(finger-force ~af-fx {*z>0} ~bf-fx=+z)
->
(renew grip-done ~object *x)
(send* c-hand :frm (?hand-pos))
(send* *x :assoc 'gripped-position (?hand-pos))
(send* *x :moved c-hand :frm) '^. grip)
(send* c-hand :grip *x)
)
```

(retract hand-empty)

)

ルールの意味

手が空である,
物体 *x の把持位置に手がある,
指の幅 *y が閉じて物体 *x の把持幅になった,
指の横方向に力 *z が発生した,
ならば
“物体 *x の把持がなされた”という事実を更新せよ,
手の現在位置を記録せよ,
物体 *x に把持位置を記録せよ,
物体 *x に現在の位置で把持がなされたと知らせよ,
手に対し物体 *object を持ったと知らせよ.
“手が空である”という事実を抹消せよ.

(defrule put-on

(in-cell ~object *h)
!(move-up-done ~object {*o ?neq *h})
(hand-frm ~position
 {*y ?neq nil :within-put *h})
(hand-weiht ~fz<0)
->
(renew put-on-done ~holder *h ~object *o)
(send *o :move *y 'put)
(p-remove 2)

)

ルールの意味

注目している領域に物体 *h がある,
*h と違う物体 *o に対し move-up-done している,
手の位置 *y が *h への設置位置にある,
手にかかる物体の重さが負の向きである,
ならば
“物体 *o を *h の上に置いた”という事実を更新せよ,
物体 *o に位置 *y で設置されたと知らせよ,
事実 2 を抹消せよ.

参考文献

- 1) W. R. Ferrel and T. B. Sheridan, "Supervisory control of remote manipulator", IEEE Spectrum, pp. 81-88, October 1967.
- 2) R. Fournier, P. Gravez and C. Mangeot, "High-Level Hierarchical Control in Computer Aided Teleoperation (CAT)", "Proc. 87 ICAR", pp. 411-420, 1987.
- 3) T. Hasegawa, et al, "An Integrated Tele Robotics System With a Geometric Environment Model and Manipulation Skills," Proc. IROS 90', pp. 335-341, 1990.
- 4) 渡辺, 内山, "宇宙実験用テレオペレーションシステム" 日本ロボット学会誌, Vol.7, No.6, pp. 759-764, 1989.
- 5) L. Conway, R. A. Volz and M. W. Walker, "Teleautonomous Systems : Projecting and Coordinateing Intelligent Action at Distance, IEEE Trans. Robotics and Automation, Vol.6, No.2, pp. 146-158, 1990.
- 6) E. P. Kan, "Supervised Robotics for Space Servicing," Proc. IROS '90, pp. 437-443, 1990.
- 7) 平井, 佐藤, LARTS/Tを用いた言語主導型マスタスレーブマニピュレーション法, 日本ロボット学会誌, Vol.2, No.6, pp. 526-535, 1984.
- 8) 佐藤, 平井, "ワールドモデルを利用したテレオペレータシステムの構成法", 日本ロボット学会誌, Vol.4, No.4, pp. 15-25, 1986.
- 9) 平井, 佐藤, "テレロボットにおけるワールドモデル管理のための操縦行動の理解機能", 日本ロボット学会誌, Vol.7, No.6, pp. 714-724, 1989.
- 10) 平井, 佐藤, 脊間, "知的遠隔作業ロボットシステム—グラフィックスと聴覚による操作者支援-", SAIRAS '89-講演集, pp. 260-263, 1989.
- 11) 平井, 佐藤, "言語介在型マスタスレーブマニピュレータシステム, 計測自動制御学会論文集, Vol.20, No.1, pp. 78-84, 1984.
- 12) 平井, 佐藤, ダイレクト・ドライブマスター・マニピュレータ: その制作と性能測定, 日本ロボット学会誌, Vol.5, No.1, pp. 14-18, 1987.
- 13) 松井, 塚本, "マルチメディアディスプレイを用いた統合型遠隔ロボット制御法", 日本ロボット学会誌, Vol.6, No.4, 301-310, 1988.
- 14) ×68000 取扱説明書, SHARP.
- 15) T. Matsui and M. Inaba, "EusLisp : An Object-Oriented Implementation of Lisp," Journal of Information Processing, Vol.13, No.3, 1990.