

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2013-163247  
(P2013-163247A)

(43) 公開日 平成25年8月22日(2013.8.22)

(51) Int.Cl. F I テーマコード(参考)  
**B 2 5 J 13/00 (2006.01)** B 2 5 J 13/00 Z 3 C 7 0 7

審査請求 未請求 請求項の数 7 O L (全 12 頁)

(21) 出願番号 特願2012-28096 (P2012-28096)  
 (22) 出願日 平成24年2月13日(2012.2.13)

(71) 出願人 000002369  
 セイコーエプソン株式会社  
 東京都新宿区西新宿2丁目4番1号  
 (74) 代理人 100095728  
 弁理士 上柳 雅誉  
 (74) 代理人 100107261  
 弁理士 須澤 修  
 (74) 代理人 100127661  
 弁理士 宮坂 一彦  
 (72) 発明者 大内 真  
 長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内  
 Fターム(参考) 3C707 LS08 LS20 LW08

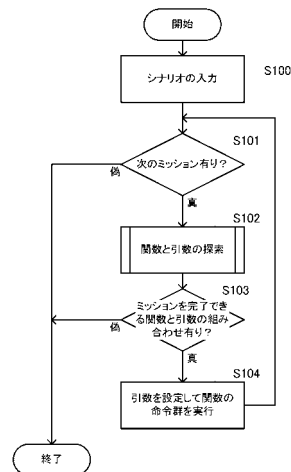
(54) 【発明の名称】 ロボットシステム、ロボット、ロボット制御装置およびロボット制御方法

(57) 【要約】

【課題】 ロボットティーチングを容易にする。

【解決手段】 複数のロボットコンポーネントと前記複数のロボットコンポーネントを制御する制御部とを備えるロボットを制御するロボット制御方法であって、前記ロボットコンポーネントを制御するための複数の関数を記憶し、ロボット外対象物の状態変化を含むミッションの入力を受け付け、前記変化に対応する前記関数を選択し、選択された前記関数に任意の引数を設定し、前記引数が設定された前記関数の実行結果をシミュレートすることにより、前記変化をもたらす前記関数と前記引数とを特定し、特定された前記関数と前記引数に対応する命令群を前記制御部へ出力する、ことを含むロボット制御方法。

【選択図】 図2



**【特許請求の範囲】****【請求項 1】**

複数のロボットコンポーネントと、  
前記ロボットコンポーネントを制御するための複数の関数を記憶するデータベースと、  
ロボット外対象物の状態を変化させることを含むミッションの入力を受け付ける入力受付部と、

前記変化に対応する前記関数を選択する選択部と、

選択された前記関数に任意の引数を設定し、前記引数が設定された前記関数の実行結果をシミュレートすることにより、前記変化をもたらす前記関数と前記引数の組み合わせを特定し、特定された前記関数と前記引数の組み合わせに対応する命令群を出力する出力部と、

10

前記命令群を実行することにより前記ロボットコンポーネントを制御する制御部と、  
を備えるロボットシステム。

**【請求項 2】**

前記出力部は、少なくとも 1 つの前記ロボットコンポーネントの識別子を前記引数として設定する、

請求項 1 に記載のロボットシステム。

**【請求項 3】**

前記ミッションは、評価指標を含み、

前記出力部は、前記変化をもたらす前記関数と前記引数の複数の組み合わせのうち、前記評価指標に基づいて前記変化を評価した場合に最も評価が高くなる 1 組に対応する命令群を出力する、

20

請求項 1 または 2 に記載のロボットシステム。

**【請求項 4】**

前記入力受付部は、前記ミッションを複数含むシナリオの入力を受け付け、

前記選択部、前記出力部および前記制御部は、前記シナリオに含まれる複数の前記ミッションについて連続して複数の前記関数を処理する、

請求項 1 から 3 のいずれか一項に記載のロボットシステム。

**【請求項 5】**

複数のロボットコンポーネントと前記複数のロボットコンポーネントを制御する制御部とを備えるロボットを制御するロボット制御装置であって、

30

前記ロボットコンポーネントを制御するための複数の関数を記憶するデータベースと、  
ロボット外対象物の状態を変化させることを含むミッションの入力を受け付ける入力受付部と、

前記変化に対応する前記関数を選択する選択部と、

選択された前記関数に任意の引数を設定し、前記引数が設定された前記関数の実行結果をシミュレートすることにより、前記変化をもたらす前記関数と前記引数の組み合わせを特定し、特定された前記関数と前記引数の組み合わせに対応する命令群を前記制御部に出力する出力部と、

を備えるロボット制御装置。

40

**【請求項 6】**

複数のロボットコンポーネントと前記複数のロボットコンポーネントを制御する制御部とを備えるロボットを制御するロボット制御方法であって、

前記ロボットコンポーネントを制御するための複数の関数を記憶し、

ロボット外対象物の状態を変化させることを含むミッションの入力を受け付け、

前記変化に対応する前記関数を選択し、

選択された前記関数に任意の引数を設定し、前記引数が設定された前記関数の実行結果をシミュレートすることにより、前記変化をもたらす前記関数と前記引数の組み合わせを特定し、特定された前記関数と前記引数の組み合わせに対応する命令群を前記制御部に出力する、

50

ことを含むロボット制御方法。

【請求項 7】

複数のロボットコンポーネントと、  
前記ロボットコンポーネントを制御するための複数の関数を記憶するデータベースと、  
ロボット外対象物の状態を変化させることを含むミッションの入力を受け付ける入力受付部と、

前記変化に対応する前記関数を選択する選択部と、

選択された前記関数に任意の引数を設定し、前記引数が設定された前記関数の実行結果をシミュレートすることにより、前記変化をもたらす前記関数と前記引数の組み合わせを特定し、特定された前記関数と前記引数の組み合わせに対応する命令群を出力する出力部と、

前記命令群を実行することにより前記ロボットコンポーネントを制御する制御部と、  
を備えるロボット。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ロボットシステム、ロボット、ロボット制御装置およびロボット制御方法に関する。

【背景技術】

【0002】

従来、オペレーターが入力したコマンドをロボットに実行させる前にそのコマンドの実行結果をシミュレートし、コマンドが実行可能であることが確認されてからそのコマンドを実行させることにより、ティーチングを容易にする方法が知られている。例えば特許文献 1 には、コマンドをオペレーターが入力してロボットをある位置から別の位置に移動させる際、実際にロボットが置かれている状況をセンサーセンサーを用いて特定し、その状況において許容される移動経路をシミュレーションによって特定した後に、ロボットに実行させるコマンドを決定する技術が記載されている。

【先行技術文献】

【特許文献】

【0003】

【特許文献 1】特開 2010 - 23227 号公報

【発明の概要】

【発明が解決しようとする課題】

【0004】

ところで、ロボットに与えられるミッションの本質は、ロボットを用いてロボット外対象物に何らかの変化をもたらすことであって、どのロボットコンポーネントをどのように動作させるかはそのミッションに従属する事柄である。例えば、複数のアームやハンドを備えた産業用ロボットを用いて第一のワークを第二のワークに取り付けるというミッションにおいて、どのアームやハンドを用いてそのミッションを達成することが最適であるかはロボットの仕様によって異なるが、どのアームやハンドが用いられようとも第一のワークが第二のワークに取り付けられたという結果が得られれば、ミッションは達成されたことになる。

【0005】

しかし、ロボットを構成するロボットコンポーネントが複雑化しているにもかかわらず、従来の技術によると、ロボットを構成する複数のロボットコンポーネントのそれぞれに与えられている機能をオペレーターが熟知していないことには、オペレーターはロボットを操作することができないという問題がある。例えば特許文献 1 に記載された技術を用いて特定のロボットコンポーネントにそれが置かれている状況に対して最適な動作をさせられるとしても、オペレーターはどのロボットコンポーネントを用いて目的を達成すべきかを知っていなければならない。すなわち、従来の技術では、ロボットコンポーネントとい

10

20

30

40

50

うミッションの達成手段を逐一指定しなければ、ロボットを操作することができないという問題があった。

【0006】

本発明は、このような問題を解決するために創作されたものであって、ロボットティーチングを容易にすることを目的の1つとする。

【課題を解決するための手段】

【0007】

(1) 上記目的を達成するためのロボットシステムは、複数のロボットコンポーネントと、前記ロボットコンポーネントを制御するための複数の関数を記憶するデータベースと、ロボット外対象物の状態を変化させることを含むミッションの入力を受け付ける入力受付部と、前記変化に対応する前記関数を選択する選択部と、選択された前記関数に任意の引数を設定し、前記引数が設定された前記関数の実行結果をシミュレートすることにより、前記変化をもたらす前記関数と前記引数の組み合わせを特定し、特定された前記関数と前記引数の組み合わせに対応する命令群を出力する出力部と、前記命令群を実行することにより前記ロボットコンポーネントを制御する制御部と、を備える。

10

【0008】

ここでロボット外対象物の状態を変化させることを含むミッションは、ロボットコンポーネントの動作によってワークにどのような状態変化が生じさせるかを特定する情報であって、例えばロボットコンポーネントが動作する前後のワークの位置、姿勢、形状、色、温度等を表す情報である。本発明によると、ミッションが入力されると、ミッションとして入力されたロボット外対象物の状態変化に応じて、その状態変化をもたらすロボットコンポーネントを制御するための関数が選択され、シミュレーションによってその状態変化をもたらす関数と引数の組み合わせが特定される。したがって、ロボットの仕様に応じて設計されている関数を選択し状態変化をもたらす引数を特定して設定するオペレーターの手間が省かれる。すなわち本発明によると、ロボットコンポーネントの動作を特定しなくてもロボットに対するティーチングが可能であるため、ロボットティーチングが容易になる。

20

【0009】

(2) 上記目的を達成するためのロボットシステムにおいて、前記出力部は、少なくとも1つの前記ロボットコンポーネントの識別子を前記引数として設定してもよい。

30

引数として適するロボットコンポーネントの識別子がロボットシステムによって設定される場合、ミッションを達成するのに適したロボットコンポーネントを選択するオペレーターの手間が省かれる。

【0010】

(3) 上記目的を達成するためのロボットにおいて、前記ミッションは、評価指標を含み、前記出力部は、前記変化をもたらす前記関数と前記引数の複数の組み合わせのうち、前記評価指標に基づいて前記変化を評価した場合に最も評価が高くなる1組に対応する命令群を出力してもよい。

ここで評価指標とは、シミュレートされた状態変化を評価するための指標であって、例えば所要時間、成功確率、安全度、消費電力量、最大負荷などである。評価指標を含むミッションが入力されるとともに、その評価指標に基づいて最も評価が高くなる関数と引数とがロボットシステムによって設定される場合、最適な状態変化を得るためのティーチングが容易になる。

40

【0011】

(4) 上記目的を達成するためのロボットにおいて、前記入力受付部は、前記ミッションを複数含むシナリオの入力を受け付け、前記選択部、前記出力部および前記制御部は、前記シナリオに含まれる複数の前記ミッションについて連続して複数の前記関数を処理してもよい。

複数のミッションを含むシナリオの入力に応じて複数のミッションが連続して達成される場合、ミッション毎にティーチングする必要がなくなるため、ロボットティーチングが

50

さらに容易になる。

【0012】

なお、本発明はロボット制御装置としてもロボットとしてもロボット制御方法としてもロボット制御プログラムとしてもロボット制御プログラムの記録媒体としても成立する。むしろ、そのコンピュータプログラムの記録媒体は、磁気記録媒体であってもよいし光磁気記録媒体であってもよいし、今後開発されるいかなる記録媒体であってもよい。

【図面の簡単な説明】

【0013】

【図1】本発明の実施形態にかかるブロック図。

【図2】本発明の実施形態にかかるフローチャート。

10

【図3】本発明の実施形態にかかるフローチャート。

【図4】図4A、図4Bおよび図4Cは本発明の実施形態にかかる模式図。

【図5】本発明の実施形態にかかるブロック図。

【発明を実施するための形態】

【0014】

以下、本発明の実施の形態を添付図面を参照しながら説明する。尚、各図において対応する構成要素には同一の符号が付され、重複する説明は省略される。

1. ロボットシステムの構成

図1は、本発明にかかるロボットシステム100の構成を示すブロック図である。ロボットシステム100は、PC(Personal Computer)10とロボット20とデータベース30とを備えている。PC10とデータベース30とは、LANによって接続されており、ロボット制御装置を構成している。PC10およびデータベース30のそれぞれは、CPU、RAM、ROM、入出力機構、ハードディスク装置等を備える。

20

【0015】

PC10は、キーボード12、マウス13、ディスプレイ11等の操作装置に接続されている。PC10は、ハードディスク装置に記憶されたロボット制御プログラムをRAMにロードしてCPUで実行することにより、次に述べる入力受付部10a、選択部10b、出力部10cとして機能する。

【0016】

入力受付部10aは、ロボット外対象物としてワークの状態変化と評価指標とをそれぞれ含む複数のミッションからなるシナリオの入力を受け付ける。本実施例における状態変化は動作前後のワークの位置を表す情報であるとする。また本実施例における評価指標は所要時間、成功確率および安全度であるとする。また本実施例におけるミッションとは、状態変化と評価指標からなるティーチングのためのコマンドである。入力受付部10aは、このようなミッションを対話形式でオペレーターに連続的にシナリオとして入力させるインターフェースを提供する。

30

【0017】

例えば、状態Bまで最も短い時間でワークXを動かすというミッションは、次の形式で入力される。ここで"time"は"所要時間"という評価指標を示すコードであるものとする。

Move From To(X, B, time)

40

【0018】

このミッションは、ロボット外対象物であるワークXの状態変化のみを規定し、そのような状態変化をもたらす手段を何ら規定していない。なお、ワークXの初期位置や初期姿勢や形状、ワークの状態Bにおける位置や姿勢や形状といった情報は、例えばミッションとは別に初期設定としてデータベース30に登録しておくこともできるし、各ミッションに含めることもできる。

【0019】

選択部10bは、状態変化に対応する1つ以上の関数をデータベース30から探索して選択する。探索される関数は、ロボット20の制御部21によって実行される命令群であって、引数を受け取って予め決められた処理を実行する命令群である。ここでいう予め決

50

められた処理とは、ロボット20に備えられたアクチュエーターを駆動したり、ロボット20に備えられたセンサーからデータを取得したりすることである。

【0020】

例えば、"1つのアームでワークXを位置Bに動かす"という関数は次の形式である。

Move From To(X, B, )

この関数は、次の命令群からなる。

arm :Move to X

arm :Grasp X

arm :Move to B

arm :Release X

10

【0021】

すなわち、アームがワークXをつかめる位置まで移動し、アームがワークXをつかみ、アームがBの位置まで移動し、アームがワークXを離す、という4つの処理からなる一連の動きを関数"Move From To(X, B, )"が規定している。

【0022】

また例えば、"2つのアーム、でワークXをBに動かす"という関数は次の形式である。

Move From To(X, B, 、 )

この関数は、次の命令群からなる。

arm :Move to X

arm :Grasp X

arm :Move to C

arm :Move to C

arm :Grasp X

arm :Release X

arm :Move to B

arm :Release X

20

【0023】

すなわち、アームがある状態のワークXをつかめる位置まで移動し、アームがワークXをつかみ、アームがCの位置まで移動し、アームがCの位置にあるワークXをつかめる位置まで移動し、アームがワークXを離し、アームがBまで移動し、アームがワークXを離す、という8つの処理からなる一連の動きを関数"Move From To(X, B, 、 )"が規定している。

30

【0024】

そしてワークXの状態は命令にともなう実際の操作によって変化するため、命令の実行にともなう状態情報がXに対応付けて記憶され都度更新される。このような状態情報は、ロボット制御装置としてのPC10が更新しても良いし、ロボット20の制御部21が更新しても良い。

【0025】

出力部10cは、選択された関数に任意の引数を設定し、引数が設定された関数の実行結果をシミュレートすることにより、状態変化をもたらす関数と引数の組み合わせを特定し、特定された関数と引数の組み合わせに対応する命令群を出力する。状態変化をもたらす関数と引数の組み合わせが複数存在する場合、出力部10cは、これらの組み合わせのうち、シミュレートされた状態変化を評価指標に基づいて評価した場合に最も評価が高くなる1組に対応する命令群を出力する。状態変化をシミュレートするに際しては、ロボット20を構成するアームなどのロボットコンポーネントの移動速度、移動経路、移動範囲、把持できる対象物の最大サイズと最大質量、把持力、消費電力量、振動特性、エラー発生確率といった特性情報が必要になる。本実施例では、データベース30に記憶されているこれらの特性情報を参照して出力部10cがシミュレーションを実行する。

40

【0026】

50

データベース 30 には、複数の関数、引数の型や取り得る値を規定する引数の宣言、ワークの形状、位置情報、ロボットコンポーネントの特性情報等が予め登録されている。データベース 30 には、ロボットシステム 100 に備わるロボットコンポーネントに対応した関数のみならず、ロボットシステム 100 に追加され得るロボットコンポーネントに対応した関数をも登録しておいてもよい。この場合、ロボットシステム 100 のセットアップ時に有効な関数と無効な関数とをフラグなどで区別しておけばよい。また引数が取り得る値をロボットシステム 100 の構成に応じて異らせる場合、ロボットシステム 100 のセットアップ時に引数が取り得る値の範囲を設定すればよい。

#### 【0027】

ロボット 20 は、第一のロボットコンポーネントである第一アーム 22、第二のロボットコンポーネントである第二アーム 23 および制御部 21 を備えている。第一アーム 22 および第二アーム 23 は、それぞれが機構部、複数のアクチュエーターアクチュエーター、複数のセンサーセンサー、およびそれらの駆動回路を備える。制御部 21 は、CPU、RAM、ROM、入出力機構等を備えるコンピューターである。制御部 21 は、出力部 10c から出力された命令群を実行することにより、これらのアクチュエーターアクチュエーターとセンサーセンサーを制御し、第一アーム 22 および第二アーム 23 に、"ワークをつかむ"、"ワークを動かす"、"ワークをはなす"といった操作を行わせる。

#### 【0028】

##### 2. ロボット制御方法

以下、ロボットシステム 100 の制御方法を図 2 および図 3 に示すフローチャートに基づいて説明する。

はじめに PC 10 に対するシナリオの入力によってロボットシステム 100 に対するティーチングが行われる (S100)。具体的には、入力受付部 10a がディスプレイ 11 に表示する GUI に従ってオペレーターがキーボード 12 およびマウス 13 を操作することによって複数のミッションが時系列に統合されたシナリオとして連続的に入力される。例えば、ワーク X を第一の位置に動かすという第一のミッションと、ワーク Y を第一の位置に動かすという第二のミッションと、ワーク X とワーク Y とをねじ結合するという第三のミッションとが 1 つのシナリオとして対話的に入力される。

#### 【0029】

任意のオペレーターがシナリオを入力する以前の段階で、ロボットシステム 100 の構成とワークの構成に応じた様々な初期情報を運用管理者がデータベース 30 に登録しておくことでティーチングが容易になる。すなわち、ロボットシステム 100 のセットアップとティーチングを分けることにより、ティーチングが容易になる。なお、シナリオの入力を受け付けた結果をファイルとして保存し、シナリオが保存されているファイルを PC 10 に入力することでシナリオの入力を受け付けても良い。

#### 【0030】

入力受付部 10a がシナリオの入力を受け付けると、選択部 10b は、シナリオに含まれる全てのミッションに順次注目し、注目したミッションについて関数と引数の探索処理を実行する (S101、S102)。

#### 【0031】

図 3 は関数と引数の探索処理の詳細を示すフローチャートである。

関数と引数の探索処理において、選択部 10b は、まず、現在注目しているミッションに対応する関数をデータベースから 1 つずつ探索する (S1020)。ミッションと関数の対応関係は、引数部分を除いた命令コード文字列の完全一致または部分一致によって規定される。具体的には例えば、ミッションが前述した "Move From To(X、B、time)" であって、データベースには、"Move From To(X、B、 )" と、"Move From To(X、B、 )" とが有効な関数として登録されているとする。この場合、"Move From To" の完全一致により、ミッション "Move From To(X、B、time)" に対応する関数として "Move From To(X、B、 )" と、"Move From To(X、B、 )" とが順に探索される。なお、ミッションと関数の対応関係をデータベース 30 に別途登録しておいてもよい。

10

20

30

40

50

## 【 0 0 3 2 】

次に、出力部 1 0 c は、ミッションに対応する関数が新たに 1 つ探索されたか否かを判定し ( S 1 0 2 1 )、関数が探索された場合には、関数に対応する引数を探索する ( S 1 0 2 2 )。すなわち、探索された関数の引数が取り得る値をデータベース 3 0 から 1 つずつ探索する。例えば、関数 "Move From To(X、B、 )" が探索されているとする。ここでワークを意味する X と状態または位置を意味する B は、シナリオの入力前またはシナリオの入力時に設定されている定数である。したがって、探索対象となる引数はアームを意味する である。ロボットシステム 1 0 0 において が取り得る値は、第一アーム 2 2 の識別子 "arm1" と、第二アーム 2 3 の識別子 "arm2" である。データベース 3 0 には、 が取り得る値として "arm1" と "arm2" とが登録されているため、関数 "Move From To(X、B、 )" に対応する引数として "arm1" と "arm2" とが 1 つずつ探索される。なお、ここでは単純に引数の取り得る値としてデータベース 3 0 に登録されている値が探索されているため、具体的なワーク X を具体的な状態又は位置である B に第一アーム 2 2 または第二アーム 2 3 によって動かすことができるかは、問題とされていない。

10

## 【 0 0 3 3 】

次に出力部 1 0 c は、探索された関数に対応する引数が新たに 1 つ探索されたか否かを判定し ( S 1 0 2 3 )、引数が探索された場合には、探索された関数と引数の組み合わせをシミュレーターで実行する ( S 1 0 2 4 )。ここでシミュレーターによって、データベース 3 0 に登録されているワークの形状、位置情報、ロボットコンポーネントの特性情報等が参照され、探索された関数とその引数の組み合わせで関数に規定されている命令群を実行した場合の処理予測が出力される。処理予測としては、成否、所要時間、成功確率、安全度が出力される。

20

## 【 0 0 3 4 】

シミュレーターによって処理予測が出力されると、出力部 1 0 c は、探索された関数と引数の組み合わせでミッションを完遂することができるか否かを処理予測に基づいて判定する ( S 1 0 2 5 )。具体的には、処理予測が成功であれば、探索された関数と引数の組み合わせでミッションを完遂できると判定し、処理予測が失敗であれば、探索された関数と引数の組み合わせでミッションを完遂できないと判定する。

## 【 0 0 3 5 】

探索された関数と引数の組み合わせでミッションを完遂できると判定した場合、出力部 1 0 c は、その関数と引数の組み合わせと処理予測とを保存する ( S 1 0 2 6 )。探索された関数と引数の組み合わせでミッションを完遂できないと判定した場合、出力部 1 0 c は、再び関数に対応する引数を探索し、探索した関数と引数の組み合わせについて処理予測を保存する処理 ( S 1 0 2 2 ~ S 1 0 2 6 ) を、探索した関数に対応する引数が無くなるまで ( S 1 0 2 3 で偽判定されるまで ) 繰り返す。

30

## 【 0 0 3 6 】

ステップ S 1 0 2 0 で探索した 1 つの関数について対応する全ての引数の探索とシミュレーションが終了すると、注目しているミッションについて対応する次の関数を再びステップ S 1 0 2 0 で探索し、新たに探索された 1 つの関数についてステップ S 1 0 2 2 ~ S 1 0 2 6 の処理を繰り返す。

40

## 【 0 0 3 7 】

注目しているミッションについて対応する次の関数が無くなると ( ステップ S 1 0 2 1 で偽判定となる )、出力部 1 0 c は、保存された関数と引数の処理予測をミッションの評価指標に基づいて評価する ( S 1 0 2 8 )。

## 【 0 0 3 8 】

そして、出力部 1 0 c は評価指標に基づいて評価した結果、最も評価が高かった組み合わせを選択する ( S 1 0 2 9 )。

## 【 0 0 3 9 】

ここで例えば、"Move From To(X、B、 )" と "Move From To(X、B、 、 )" とがミッションに対応する関数として探索され、"Move From To(X、B、 )" に対応する として引数

50

"arm1"と"arm2"が探索されたとする。そして、"Move From To(X、B、 )"に対応するとして引数"arm1"と"arm2"が探索され、"Move From To(X、B、 arm1、 )"に対応するとして引数"arm1"と"arm2"が探索され、"Move From To(X、B、 arm2、 )"に対応するとして引数"arm1"と"arm2"が探索されたと仮定する。

【 0 0 4 0 】

図 4 A に示すように第一アーム 2 2 で位置 A にあるワーク X を位置 B まで動かすことができるのに対して、図 4 B に示すように第二アーム 2 3 ではそれができない。したがって、関数"Move From To(X、B、 )"と の引数"arm1"との組み合わせの実行結果は成功と予測される。そして関数"Move From To(X、B、 )"と の引数"arm2"との組み合わせの実行結果は失敗と予測される。また、図 4 C に示すように第一アーム 2 2 から第二アーム 2 3 にワーク X を受け渡して位置 A にあるワーク X を位置 B まで動かすことができるため、関数"Move From To(X、B、 )"と の引数"arm1"と の引数"arm2"との組み合わせの実行結果は成功と予測される。しかし、第一アーム 2 2 も第二アーム 2 3 も自分自身でワーク X を受け渡すことはできないため、関数"Move From To(X、B、 )"と、 の引数"arm1"と の引数"arm1"との組み合わせも、 の引数"arm2"と の引数"arm2"との組み合わせも、実行結果は失敗と予測される。また、第二アーム 2 3 は図 4 B に示すように位置 A にあるワーク X をつかめる位置まで動けないため、関数"Move From To(X、B、 )"と の引数"arm2"と の引数"arm1"との組み合わせの実行結果も失敗と予測される。したがって、この場合、保存される関数と引数の組み合わせは、"Move From To(X、B、 )"と の引数"arm1"との組み合わせと、"Move From To(X、B、 )"と の引数"arm1"と の引数"arm2"との組み合わせのみとなる。

10

20

【 0 0 4 1 】

そして、評価指標が所要時間であれば、2つの成功する組み合わせのうち最も所要時間が短い組み合わせが選択され、評価指標が安全度であれば、2つの成功する組み合わせのうち最も安全度が高い組み合わせが選択され、評価指標が成功確率であれば、2つの成功する組み合わせのうち最も成功確率が高い組み合わせが選択される。例えば所要時間および成功確率の評価指標に対しては、ワークの受け渡しが必要な"Move From To(X、B、 )"と の引数"arm1"との組み合わせが選択される。また例えば安全度の評価指標に対しては、アームの運動範囲が狭い"Move From To(X、B、 )"と の引数"arm1"と の引数"arm2"との組み合わせが選択される。

30

【 0 0 4 2 】

図 2 においてステップ S 1 0 2 として示された関数と引数の探索処理を以上述べたとおりに実行すると、出力部 1 0 c は、図 2 に示すステップ S 1 0 3 において、ミッションを完了できる関数と引数の組み合わせがあったか否か、すなわち、関数と引数の組み合わせが保存されているか否かを判定する。関数と引数の組み合わせが保存されていない場合、シナリオは実行できないものとして処理を終了する。

【 0 0 4 3 】

関数と引数の組み合わせが保存されている場合、出力部 1 0 c は、保存された引数を保存された関数に設定し、関数の命令群を出力する。その結果、出力部 1 0 c から出力された命令群が制御部 2 1 によって実行される。制御部 2 1 は命令群の実行に伴って第一アーム 2 2 および第二アーム 2 3 を制御する ( S 1 0 4 ) 。

40

【 0 0 4 4 】

1つの関数の命令群が実行されると、選択部 1 0 b および出力部 1 0 c は次のミッションが無くなるまでシナリオに含まれる全てのミッションについて順次上述の処理を繰り返し、次のミッションが無くなると ( S 1 0 1 で偽判定の場合 )、シナリオの処理を終了する。すなわち、全てのミッションについてシミュレーションをしてから命令群を現実に実行するのではなく、1つのミッションについて命令群を現実に実行してから、次の1つのミッションについてシミュレーションを実行する。このため、例えば、1つのミッションについて現実に命令群を実行した結果を次のミッションのシミュレーションにフィードバックすることもできる。

50

## 【 0 0 4 5 】

以上説明した実施例によると、複数のミッションを含むシナリオが入力されると、ミッションとして入力されたロボット外対象物であるワークの状態変化に応じて、その状態変化をもたらすロボットコンポーネントを制御するための関数が選択され、シミュレーションによってその状態変化をもたらす関数と引数の組み合わせが特定される。したがって、ロボット20の仕様に応じて設計されている関数を選択し状態変化をもたらす引数を特定して設定するオペレーターの手間が省かれる。具体的には、引数として適するアームの識別子がPC10によって自動設定されるため、ミッションを達成するのに適したアームを選択するオペレーターの手間が省かれる。さらに、評価指標に基づいて最も評価が高くなる関数と引数とがPC10によって自動設定されるため、最適な状態変化を得るためのティーチングが容易になる。また、複数のミッションを含むシナリオの入力に応じて複数のミッションが連続して達成されるため、ミッション毎にティーチングする必要がなくなるため、ロボットティーチングがさらに容易になる。

10

## 【 0 0 4 6 】

## 3. 他の実施形態

尚、本発明の技術的範囲は、上述した実施例に限定されるものではなく、本発明の要旨を逸脱しない範囲内において種々変更を加え得ることは勿論である。例えば、図5に示すようにロボット制御装置をロボット20とを一体化しても良い。またデータベースを複数のロボット制御装置が共用するサーバーコンピュータとして構成し、インターネット等の通信回線を介して選択部および出力部と通信させるようにしてもよい。また入力受付部、選択部および出力部として機能するコンピュータとその不揮発性記憶媒体とをデータベースとして機能させても良い。

20

## 【 0 0 4 7 】

また、シミュレーターで関数と引数の組みあわせの実行結果を予測する際に、ミッションに設定されていない評価指標に対応する予測については予測しなくてもよい。これにより、シミュレーションに要する時間を短縮することができる。

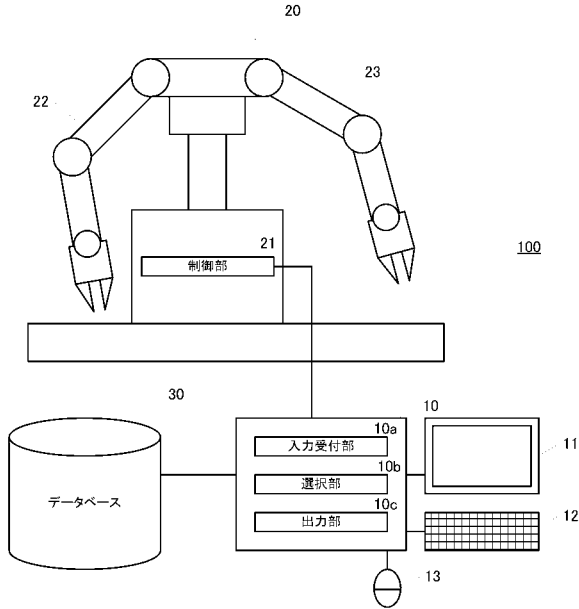
## 【 符号の説明 】

## 【 0 0 4 8 】

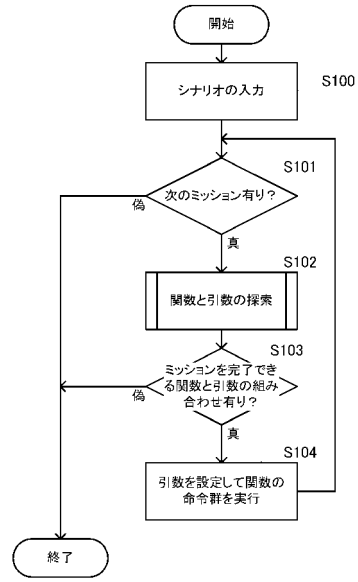
10 a ... 入力受付部、10 b ... 選択部、10 c ... 出力部、11 ... ディスプレイ、12 ... キーボード、13 ... マウス、20 ... ロボット、21 ... 制御部、22 ... 第一アーム、23 ... 第二アーム、30 ... データベース、100 ... ロボットシステム

30

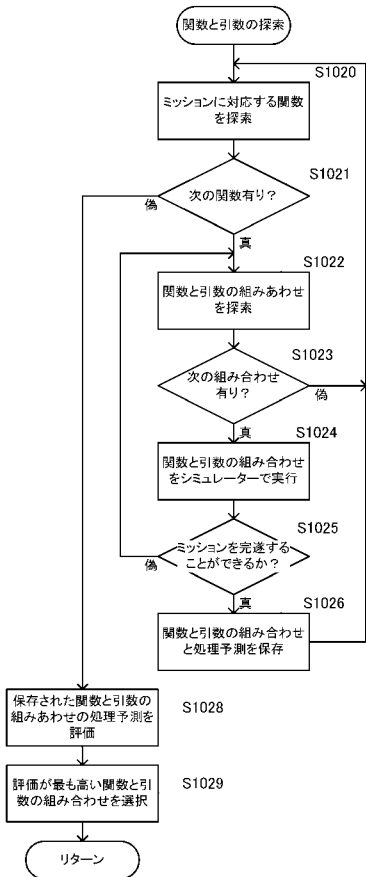
【 図 1 】



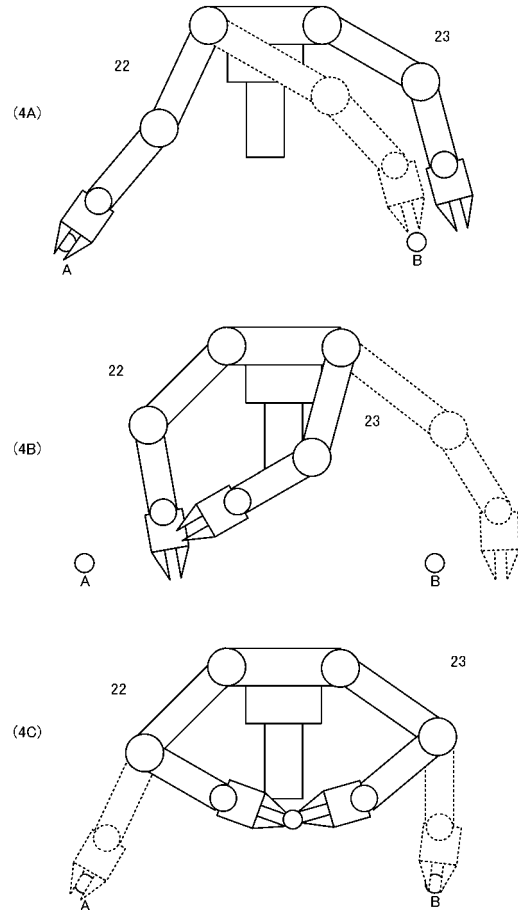
【 図 2 】



【 図 3 】



【 図 4 】



【 図 5 】

