

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6668629号
(P6668629)

(45) 発行日 令和2年3月18日(2020.3.18)

(24) 登録日 令和2年3月2日(2020.3.2)

(51) Int.Cl. F 1
B 2 5 J 13/08 (2006.01) B 2 5 J 13/08 Z

請求項の数 2 (全 12 頁)

| | |
|--|--|
| <p>(21) 出願番号 特願2015-150428 (P2015-150428) (22) 出願日 平成27年7月30日 (2015.7.30) (65) 公開番号 特開2017-30068 (P2017-30068A) (43) 公開日 平成29年2月9日 (2017.2.9) 審査請求日 平成30年7月23日 (2018.7.23)</p> | <p>(73) 特許権者 000002369 セイコーエプソン株式会社 東京都新宿区新宿四丁目1番6号 (74) 代理人 100116665 弁理士 渡辺 和昭 (74) 代理人 100194102 弁理士 磯部 光宏 (74) 代理人 100179475 弁理士 仲井 智至 (74) 代理人 100216253 弁理士 松岡 宏紀 (72) 発明者 下平 泰裕 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</p> |
|--|--|

最終頁に続く

(54) 【発明の名称】 ロボット制御装置およびロボットシステム

(57) 【特許請求の範囲】

【請求項1】

マニピュレーターと、
 前記マニピュレーターに作用する力を検出する力検出器と、
 を備えるロボットを制御するロボット制御装置において、
 前記マニピュレーターの動作を制御する制御部は、前記マニピュレーターを動作させる
 コマンドを実行するとき、前記力検出器の出力に基づいて前記マニピュレーターの動作を
 制御するか、前記力検出器の出力に基づかずに前記マニピュレーターの動作を制御するか
 を前記コマンドのパラメータに応じて切り換え、
前記制御部は、前記コマンドを実行し、前記力検出器の出力に基づいて前記マニピュレ
ーターの動作を制御するとき、前記力検出器をリセットするとともに、前記力検出器のリ
セットから所定時間経過後に、前記力検出器の出力に基づく前記マニピュレーターの動作
制御を開始する、
 ロボット制御装置。

【請求項2】

マニピュレーターと、前記マニピュレーターに作用する力を検出する力検出器と、を備
 えるロボットと、
 前記マニピュレーターの動作を制御する制御部を備えるロボット制御装置と、を備え、
前記制御部は、
 前記マニピュレーターを動作させるコマンドを実行するとき、前記力検出器の出力に基

10

20

づいて前記マニピュレーターの動作を制御するか、前記力検出器の出力に基づかずに前記マニピュレーターの動作を制御するかを前記コマンドのパラメータに応じて切り換え、

前記コマンドを実行し、前記力検出器の出力に基づいて前記マニピュレーターの動作を制御するとき、前記力検出器をリセットするとともに、前記力検出器のリセットから所定時間経過後に、前記力検出器の出力に基づく前記マニピュレーターの動作制御を開始する

を備えるロボットシステム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ロボット制御装置およびロボットシステムに関する。

【背景技術】

【0002】

ロボットの分野においては、マニピュレーターに加わる力に応じてマニピュレーターを制御する力制御が用いられている。たとえば、特許文献1の図2には、「Activate Force Control;」コマンドと、「Deactivate ForceControl;」コマンドとによって、マニピュレーターに加わる力に応じてマニピュレーターを制御する力制御モードと、マニピュレーターに加わる力と無関係にマニピュレーターを制御する位置制御モードとの切換が行われるプログラムが記載されている。

【先行技術文献】

【特許文献】

【0003】

【特許文献1】米国特許7340323号公報

【発明の概要】

【発明が解決しようとする課題】

【0004】

しかし、特許文献1に記載されているようにコマンドによって位置制御モードから力制御モードに切り換えた後に、加わる力に応じてマニピュレーターを動作させるコマンドを実行する場合、マニピュレーターの動作結果の再現性が低くなるという問題がある。すなわち、水晶の圧電効果を用いている場合など、力検出器によっては出力が時間に依存するため、力検出器のどの時間の出力に基づいてマニピュレーターを制御するかによって、動作結果が異なることになる。具体的には、位置制御モードから力制御モードに切り換わった時点における力検出器の出力を基準としてマニピュレーターに加わる力を検出すると、力制御モードに切り換わってからマニピュレーターを実際に動かすコマンドが実行されるまでの時間の違いにより、マニピュレーターの動作結果が異なることになる。

【0005】

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、ロボットに対する力制御の再現性を高める技術を提供することを目的とする。

【課題を解決するための手段】

【0006】

前記目的を達成するためのロボット制御装置は、マニピュレーターと、前記マニピュレーターに作用する力を検出する力検出器と、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、前記マニピュレーターの動作を制御する制御部は、前記マニピュレーターを動作させるコマンドを実行するとき、前記力検出器の出力に基づいて前記マニピュレーターの動作を制御するか、前記力検出器の出力に基づかずに前記マニピュレーターの動作を制御するかを当該コマンドのパラメータに応じて切り換える。

【0007】

ここで「マニピュレーターを動作させるコマンド」は、それ自体でマニピュレーターを動作させるコマンドを意味し、設定、モードの切換といった、それ自体ではマニピュレーターを動作させることがないコマンドを含まない。すなわち例えば、本発明では、マニピ

10

20

30

40

50

ュレーターを動作させるコマンドの引数やコマンドの本体自体によって、力検出器の出力に基づいてマニピュレーターの動作を制御するか否かが決定される。したがって本発明によると、力検出器の出力に基づいてマニピュレーターの動作を制御する場合には、マニピュレーターを実際に動かすコマンドが実行される時点を出力の基準とする力検出器の出力に基づいてマニピュレーターの動作を制御することができる。このため本発明によるとロボットに対する力制御の再現性を高めることができる。

【0008】

なお請求項に記載された各手段の機能は、構成自体で機能が特定されるハードウェア資源、プログラムにより機能が特定されるハードウェア資源、又はそれらの組み合わせにより実現される。また、これら各手段の機能は、各々が物理的に互いに独立したハードウェア資源で実現されるものに限定されない。

10

【図面の簡単な説明】

【0009】

【図1】ロボットシステムの斜視図である。

【図2】ロボットシステムのブロック図である。

【図3】制御装置で実行される動作制御コマンドを示す表である。

【図4】力制御対応コマンドが実行される場合の処理順序を示すフローチャートである。

【図5】力覚センサーの出力を示す折れ線グラフである。

【図6】引数オブジェクトの階層構造を説明するための図である。

【図7】引数オブジェクトの設定画面を示す画面構成図である。

20

【図8】プログラムコードの実施例と比較例である。

【発明を実施するための形態】

【0010】

以下、本発明の実施形態について添付図面を参照しながら説明する。なお、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。

【0011】

(1) ロボットシステムの構成

本発明の一実施例としてのロボットシステムは、図1に示すように、ロボット1と、エンドエフェクター2と、制御装置3と、教示装置4(ティーチングペンダント)と、を備えている。制御装置3は、本発明のロボット制御装置の構成例である。制御装置3は図示しないケーブルによりロボット1と通信可能に接続される。なお、制御装置3の構成要素がロボット1に備えられていても良い。制御装置3は、ロボット1に駆動電力を供給する電源部31とロボット1を制御するための制御部32とを備える。制御装置3と教示装置4とはケーブルで、または無線通信可能に接続される。教示装置4は、専用のコンピューターであってもよいし、ロボット1を教示するためのプログラムがインストールされた汎用のコンピューターであってもよい。例えばロボット1を教示するための専用装置であるティーチングペンダント5を教示装置4の代わりに用いても良い。さらに、制御装置3と教示装置4とは、図1に示すように別々の筐体を備えていてもよいし、一体に構成されていてもよい。

30

【0012】

ロボット1は、アームAに各種のエンドエフェクター2を装着して使用される単腕ロボットである。アームAは6つの関節J1~J6を備える。関節J1~J6によって6個のアーム部材A1~A6が連結される。関節J2、J3、J5は曲げ関節であり、関節J1、J4、J6はねじり関節である。関節J6には、ワークに対して把持や加工等を行うための各種のエンドエフェクター2が装着される。先端の関節J6の回転軸上の所定位置をツールセンターポイント(TCP)と表す。TCPの位置は各種のエンドエフェクター2の位置の基準となる。アームAとエンドエフェクター2とは、本発明のマニピュレーターの構成例である。

40

【0013】

関節J6には力覚センサーFSが備えられている。力覚センサーFSは、6軸の力検出

50

器である。力覚センサーFSは、固有の座標系であるセンサー座標系において互いに直交する3個の検出軸と平行な力の大きさと、当該3個の検出軸まわりのトルクの大きさとを検出する。なお、力覚センサーFSは本発明の力検出器の構成例であるが、関節J6以外の関節J1～J5のいずれか1つ以上に力検出器としての力覚センサーを備えても良い。

【0014】

ロボット1が設置された空間を規定する座標系をロボット座標系というとき、ロボット座標系は、水平面上において互いに直交するX軸とY軸と、鉛直上向きを正方向とするZ軸とによって規定される3次元の直交座標系である。Z軸における負の方向は概ね重力方向と一致する。またX軸周りの回転角をRXで表し、Y軸周りの回転角をRYで表し、Z軸周りの回転角をRZで表す。X、Y、Z方向の位置により3次元空間における任意の位置を表現でき、RX、RY、RZ方向の回転角により3次元空間における任意の姿勢を表現できる。以下、位置と表記した場合、姿勢も意味し得ることとする。また、力と表記した場合、トルクも意味し得ることとする。制御装置3は、アームAを駆動することによって、ロボット座標系においてTCPの位置を制御する。

10

【0015】

図2は、ロボットシステムのブロック図である。制御部32はロボット1の制御を行うための制御プログラムがインストールされたコンピューターである。制御部32は、プロセッサやRAMやROMを備え、これらのハードウェア資源がプログラムと協働することによりロボット1を制御する。

【0016】

ロボット1は、図1に示した構成のほかに、アクチュエーターとしてのモーターM1～M6と、センサーとしてのエンコーダーE1～E6とを備える。アームAを制御することはモーターM1～M6を制御することを意味する。モーターM1～M6とエンコーダーE1～E6とは、関節J1～J6のそれぞれに対応して備えられており、エンコーダーE1～E6はモーターM1～M6の回転角度を検出する。制御装置3は、モーターM1～M6の回転角度の組み合わせと、ロボット座標系におけるTCPの位置との対応関係U1を記憶している。また、制御装置3は、ロボット1が行う作業の工程ごとに目標位置 S_t と目標力 f_{s_t} の少なくともいずれかをコマンドに基づいて記憶する。目標位置 S_t と目標力 f_{s_t} を引数(パラメーター)とするコマンドはロボット1が行う作業の工程ごとに設定される。

20

30

【0017】

制御装置3は、設定された目標位置と目標力とがTCPにて実現されるように、コマンドに基づいてアームAを制御する。目標力とは、アームAの動作に応じて力覚センサーFSが検出すべき力である。ここでSの文字は、ロボット座標系を規定する軸の方向(X、Y、Z、RX、RY、RZ)のなかのいずれか1個の方向を表すこととする。また、Sは、S方向の位置も表すこととする。例えば、 $S = X$ の場合、ロボット座標系にて設定された目標位置のX方向成分が $S_t = X_t$ と表記され、目標力のX方向成分が $f_{s_t} = f_{x_t}$ と表記される。

【0018】

制御装置3は、モーターM1～M6の回転角度 D_a を取得すると、対応関係U1に基づいて、当該回転角度 D_a をロボット座標系におけるTCPの位置 $S(X, Y, Z, RX, RY, RZ)$ に変換する。また制御装置3は、TCPの位置Sと、力覚センサーFSの検出値とに基づいて、力覚センサーFSに現実に作用している作用力 f_s をロボット座標系において特定する。作用力 f_s の作用点は、TCPとは別に原点Oとして定義される。原点Oは、力覚センサーFSが力を検出している点に対応する。なお制御装置3は、ロボット座標系におけるTCPの位置Sごとに、力覚センサーFSのセンサー座標系における検出軸の方向を規定した対応関係U2を記憶している。従って、制御装置3は、ロボット座標系におけるTCPの位置Sと対応関係U2とに基づいて、ロボット座標系における作用力 f_s を特定できる。また、ロボットに作用するトルクは、作用力 f_s と、ツール接触点(エンドエフェクター2とワークの接触点)から力覚センサーFSまでの距離とから算出す

40

50

ることができ、図示されないfsトルク成分として特定される。

【0019】

制御装置3は、作用力 f_s に対して重力補償を行う。重力補償とは、作用力 f_s から重力に起因する力やトルクの成分を除去することである。重力補償を行った作用力 f_s は、エンドエフェクター2に作用している重力以外の力と見なすことができる。

【0020】

本実施例のインピーダンス制御は、仮定の機械的インピーダンスをモーターM1～M6によって実現する能動インピーダンス制御である。制御装置3は、このようなインピーダンス制御を、ワークの嵌合作業、研磨作業など、エンドエフェクター2が対象物(ワーク)から力を受ける接触状態の工程で適用する。インピーダンス制御では、目標力を後述する運動方程式に代入してモーターM1～M6の回転角度を導出する。制御装置3がモーターM1～M6を制御する信号は、PWM(Pulse Width Modulation)変調された信号である。運動方程式に基づいて目標力から回転角度を導出してモーターM1～M6を制御するモードを力制御モードというものとする。また制御装置3は、エンドエフェクター2がワークから力を受けない非接触状態の工程では、目標位置から線形演算で導出する回転角度でモーターM1～M6を制御する。目標位置から線形演算で導出する回転角度でモーターM1～M6を制御するモードを位置制御モードというものとする。また制御装置3は、目標位置から線形演算で導出する回転角度と目標力を運動方程式に代入して導出する回転角度とを例えば線型結合によって統合し、統合した回転角度でモーターM1～M6を制御するハイブリッドモードでもロボット1を制御する。制御装置3は、位置制御モードと力制御モードとハイブリッドモードを力覚センサーFSまたはエンコーダーE1～R6の検出値に基づいて自律的に切り替えることもできるし、コマンドに応じて位置制御モードと力制御モードとハイブリッドモードを切り替えることもできる。以上の構成より制御装置3は、エンドエフェクター2が目標の位置において目標の姿勢となり、かつ、エンドエフェクター2に目標の力とモーメントとが作用するようにアームAを駆動することができる。

【0021】

制御装置3は、目標力 f_{st} と作用力 f_s とをインピーダンス制御の運動方程式に代入することにより、力由来補正量 S を特定する。力由来補正量 S とは、TCPが機械的インピーダンスを受けた場合に、目標力 f_{st} との力偏差 $f_s(t)$ を解消するために、TCPが移動すべき位置 S の大きさを意味する。下記の(1)式は、インピーダンス制御の運動方程式である。

【数1】

$$m\Delta\ddot{S}(t) + d\Delta\dot{S}(t) + k\Delta S(t) = \Delta f_s(t) \quad \dots (1)$$

【0022】

(1)式の左辺は、TCPの位置 S の2階微分値に仮想慣性係数 m を乗算した第1項と、TCPの位置 S の微分値に仮想粘性係数 d を乗算した第2項と、TCPの位置 S に仮想弾性係数 k を乗算した第3項とによって構成される。(1)式の右辺は、目標力 f_{st} から現実の力 f を減算した力偏差 $f_s(t)$ によって構成される。(1)式における微分とは、時間による微分を意味する。ロボット1が行う工程において、目標力 f_{st} として一定値が設定される場合もあるし、目標力 f_{st} として時間の関数が設定される場合もある。

【0023】

仮想慣性係数 m はTCPが仮想的に有する質量を意味し、仮想粘性係数 d はTCPが仮想的に受ける粘性抵抗を意味し、仮想弾性係数 k はTCPが仮想的に受ける弾性力のバネ定数を意味する。各係数 m 、 d 、 k は方向ごとに異なる値に設定されてもよいし、方向に拘わらず共通の値に設定されてもよい。

【0024】

そして、制御装置3は、対応関係U1に基づいて、ロボット座標系を規定する各軸の方向の動作位置を、各モーターM1～M6の目標の回転角度である目標角度 D_t に変換する。そして、制御装置3は、目標角度 D_t からモーターM1～M6の現実の回転角度であるエンコーダーE1～E6の出力 D_a を減算することにより、駆動位置偏差 $D_e (= D_t - D_a$

10

20

30

40

50

)を算出する。そして、制御装置3は、駆動位置偏差 D_e に位置制御ゲイン K_p を乗算した値と、現実の回転角度 D_a の時間微分値である駆動速度との差である駆動速度偏差に、速度制御ゲイン K_v を乗算した値とを加算することにより、制御量 D_c を導出する。なお、位置制御ゲイン K_p および速度制御ゲイン K_v は、比例成分だけでなく微分成分や積分成分にかかる制御ゲインを含んでもよい。制御量 D_c は、モーターM1～M6のそれぞれについて特定される。以上説明した構成により、制御装置3は、目標力 f_{st} とに基づいてアームAを力制御モードで制御することができる。ハイブリッドモードでは、制御装置3は、目標位置 S_t に、力由来補正量 S を加算することにより動作位置($S_t + S$)を特定する。

【0025】

教示装置4には、制御装置3に目標位置 S_t と目標力 f_{st} とを引数とする実行プログラムを生成して制御装置3にロードするための教示プログラムがインストールされている。教示装置4は、ディスプレイ43やプロセッサやRAMやROMを備え、これらのハードウェア資源が教示プログラムと協働して実行プログラムを生成する。

【0026】

(2) 実行プログラム

実行プログラムは予め定められたプログラム言語によって記述され、翻訳プログラムにより中間言語を経て機械語プログラムに変換される。制御部32のCPUはクロックサイクルで機械語プログラムを実行する。翻訳プログラムは教示装置4で実行しても良いし、制御装置3で実行しても良い。実行プログラムのコマンドは本体と引数とから構成される。コマンドには、アームAとエンドエフェクター2を動作させる動作制御コマンド、エンコーダやセンサーの検出値を読み出すモニターコマンド、各種の変数を設定する設定コマンド等が含まれる。なお、本明細書において、コマンドの実行は、当該コマンドが翻訳された機械語プログラムの実行と同義である。

【0027】

図3に動作制御コマンド(本体)の一例を示す。図3に示すように動作制御コマンドは、力制御モードでアームAを動作させることができる力制御対応コマンドと、力制御モードでアームAを動作させることができない位置制御コマンドとを含む。力制御対応コマンドでは、引数により力制御モードのオンを指定できる。当該引数により力制御モードのオンが指定されない場合には、位置制御モードで当該力制御対応コマンドが実行され、当該引数により力制御モードのオンが指定される場合には、当該力制御対応コマンドは力制御モードで実行される。また力制御対応コマンドは力制御モードにおいて実行可能であり、位置制御コマンドは力制御モードでは実行不能である。力制御モードにおいて位置制御コマンドが実行されることがないように、翻訳プログラムによる構文エラーチェックが実行される。さらに、力制御対応コマンドでは、引数により力制御モードの継続を指定できる。力制御モードで実行される力制御対応コマンドにおいて当該引数により力制御モードの継続が指定された場合、力制御モードは継続され、当該引数により力制御モードの継続が指定されない場合、当該力制御対応コマンドの実行完了までに力制御モードは終了する。すなわち力制御対応コマンドが力制御モードで実行されるとしても、引数により明示的に指定されない限り、力制御モードは当該力制御対応コマンドに応じて自律的に終了し、当該力制御対応コマンドの実行終了後においてまで力制御モードが継続することはない。なお図3において「CP」は移動方向を指定できるコマンドの分類、「PTP」は目標位置を指定できるコマンドの分類、「CP+PTP」は移動方向と目標位置を指定できるコマンドの分類である。

【0028】

ここで、引数により力制御モードのオンが指定され、かつ、力制御モードの継続が指定されていない「Move」コマンドが力制御モードでない状態において実行開始される場合を例にして、力制御対応コマンドの実行手順、すなわち制御部32の動作順序について図4を参照しながら説明する。

【0029】

10

20

30

40

50

まずはじめに制御部 3 2 は、力制御モードのオンを指定する引数があるか無いかを判定する (S 1 0 2) 。

力制御モードのオンを指定する引数がある場合、制御部 3 2 は、力覚センサー F S をリセットする (S 1 1 2) 。力覚センサー F S のリセットは、力覚センサー F S の出力が、このリセットタイミングでゼロとして検出されるように力覚センサー F S 自体または制御部 3 2 が記憶するオフセット値を設定することである。

【 0 0 3 0 】

続いて制御部 3 2 は、力制御モードに遷移してインピーダンス制御を実行し、アーム A を動作させる (S 1 1 4) 。すなわち制御部 3 2 は、力覚センサー F S の出力に基づいてモーター M 1 ~ M 6 の回転角度を制御する。ここで目標力は、力制御対応コマンドの引数として指定されていてもよいし、力制御対応コマンドより先に実行される設定コマンドで指定されていてもよい。位置制御モードから力制御モードへの切替は、例えば図 2 に示す S と S t とを足し合わせる算術要素において S の入力をオンにして S t の入力をオフ (ゼロ) にする操作によって実施される。

【 0 0 3 1 】

続いて制御部 3 2 は、力制御モードを終了する (S 1 1 6) 。力制御モードから位置制御モードへの切替は、例えば図 2 に示す S と S t とを足し合わせる算術要素において S の入力をオフ (ゼロ) にして S t の入力をオンにする操作によって実施される。

【 0 0 3 2 】

力制御モードのオンを指定する引数がない場合、制御部 3 2 は位置制御モードにおいて位置制御を実行し、アーム A を動作させる (S 1 2 0) 。すなわち制御部 3 2 は、目標位置から線形演算で導出する回転角度でモーター M 1 ~ M 6 の回転角度を制御する。

【 0 0 3 3 】

ところで水晶を圧電素子として用いる力覚センサーでは、力覚センサーに加わる力に変化が無くても、図 5 A に示すように、時間の経過に伴って出力が大きくなる傾向にある。したがって、例えば図 5 B に示す時間 t_0 で力制御モードがオンになるとすると、同じ力が加わっているにもかかわらず、 t_0 から t_1 が経過した時点で検出される力 f_1 と、 t_0 から t_2 が経過した時点で検出される力 f_2 とが異なる結果となる。本実施例では、上述したように力制御対応コマンドの実行時点で力制御モードに遷移して力覚センサー F S がリセットされるため、力制御対応コマンドが実行される時点を出力の基準とする力覚センサー F S の出力に基づいてアーム A の動作を制御することができる。このためアーム A に対する力制御の再現性を高めることができる。例えば、動作制御コマンドに対して力覚センサー F S のリセットが同期しない場合、すなわち動作制御コマンドの実行に先行して力覚センサー F S がリセットされる場合、動作制御コマンドの実行が 3 m s e c 程度 ($t_2 - t_1$) 遅れることになる。力制御が 1 0 m s e c 間隔で実行されるとすると、この 3 m s e c は力制御の間隔に対し大きな実行の遅れであり再現性を悪化させる。

【 0 0 3 4 】

実行プログラムのコマンドの引数は階層化されたオブジェクト構造となっている。力制御コマンドに関連する引数のクラスを図 6 に示す。「Force Coordinate System Object」は、座標系の設定に用いられるクラスであって、F C S x (x は任意の整数) で定義することができる。「Force Control Object」は、力制御の対象となる座標軸等の設定に用いられるクラスであって、F C x (x は任意の整数) で定義することができる。「Force Trigger Object」は、分岐条件、停止条件、割り込みの設定に用いられるクラスであって、F T x (x は任意の整数) で定義することができる。「Force Monitor Object」は、データの取得、ログ記録等の設定に用いられるクラスであって、F M x (x は任意の整数) で定義することができる。「Force Control Object」、「Force Trigger Object」および「Force Monitor Object」で指定される座標系は「Force Coordinate System Object」で定義された座標系となる。すなわち、「Force Control Object」、「Force Trigger Object」および「Force Monitor Object」は、上位層クラスの「Force Coordinate System Object」に従属する下位層クラスである。「Force Sensor Object」は、力覚センサー F S

10

20

30

40

50

の制御と力覚センサー F S からの情報取得の設定に用いるクラスであって、F S x (x は任意の整数) で定義することができる。「Mass Properties Object」は、重心位置の設定に用いるクラスであって、M P x (x は任意の整数) で定義することができる。「Robot Object」は、力制御と関連したロボット 1 の情報を取得するための設定に用いるクラスであって、R o b o t x (x は任意の整数) で定義することができる。

【 0 0 3 5 】

次に、このように階層化された引数の設定方法について説明する。図 7 A は「Force Coordinate System Object」クラスの引数オブジェクトを設定する画面の構成例である。図 7 A は、「ハンド面 3 先端」というラベルが付された「力覚座標 4」(プログラムコードでは F C S 4 と記述される引数オブジェクト)の原点の T C P を基準とするロボット座標系における相対的な位置と、T C P の姿勢 (T C P に固定された座標系) を基準とする相対的な姿勢とを設定した状態を示している。ここで制御部 3 2 は、モーター M 1 ~ M 6 の回転角度の組み合わせと、ロボット座標系における T C P の位置との対応関係 U 1 を記憶しているため、「Force Coordinate System Object」クラスの引数オブジェクトにおいて T C P に対する相対的な位置と姿勢を設定することにより、基準座標系に対して任意の座標系を定義することができる。

【 0 0 3 6 】

図 7 B は、「Force Control Object」クラスの引数オブジェクトを設定する画面の構成例である。図 7 B は、「姿勢做い」というラベルが付された「力制御 2」(プログラムコードでは F C 2 と記述される引数オブジェクト)の座標系と、力制御の有効軸と、インピーダンス制御の運動方程式の係数 (仮想バネ係数、仮想粘性係数、仮想質量係数) とを設定した状態を示している。ここで座標系として「4」が選択されているため、図 7 A の画面で設定された「力覚座標 4」(F C S 4) の座標系の各軸について、「有効」の項目において力制御のオン (True) とオフ (False) が設定される。すなわち、力制御が有効になる方向は、上位の「Force Coordinate System Object」クラスのオブジェクトの設定に依存する。図 7 C は「Force Trigger Object」クラスの引数オブジェクトを設定する画面の構成例である。図 7 C においても、「接触検知」というラベルが付された「カトリガー 1」(プログラムコードでは F T 1 と記述される引数オブジェクト)の座標系として「4」が選択されているため、図 7 A の画面で設定された「力覚座標 4」(F C S 4) の座標系の各軸について処理条件が定められる。すなわち、停止条件の各変数が示す値やベクトルの方向は、上位の「Force Coordinate System Object」クラスのオブジェクトの設定に依存する。このように本実施例のコマンドの引数は階層化されているため、「Force Coordinate System Object」クラスのオブジェクトとして 1 つの座標系を設定すれば、その座標系を様々なコマンドに適用することができ、設定作業が容易になる。

【 0 0 3 7 】

力制御対応コマンドを用いたプログラムコードの実施例を比較例とともに図 8 に示す。図 8 に示す実施例は、「F C 1」という引数オブジェクトが 6 つの設定コマンド「F s e t」で定義された後に、力制御対応コマンド「M o v e」が実行され、その後に「F C 1」オブジェクトの変数の 1 つである「Fz_TargetForce」が設定コマンド「F s e t」で変更されるプログラムコードである。

【 0 0 3 8 】

図 8 に示す比較例では、コマンドの引数が階層化されておらず、力制御モードで用いられる全ての設定項目が「ForceControlSetting」コマンドにより一度に設定される。そして、カンマで区切られたそれぞれの引数の定義は、コマンド本体と引数の間に記述するカンマの数で決まる。したがって比較例のプログラムコードの意味を即座に理解することは困難である。これに対し、本発明の実施例では、力制御対応コマンドに関連する引数は、階層化されたオブジェクト毎に「Fset」コマンドにより設定されるため、プログラムコードの意味を解釈することも、プログラムコードを記述することもはるかに容易である。

【 0 0 3 9 】

本実施例のコマンド「Move P1 FC1 CF」では、「Fset」コマンドにより設定された「F

10

20

30

40

50

「C1」オブジェクトの他に、図示しない設定コマンドにより設定された「P1」オブジェクトと、「CF」が引数として設定されている。ここで「P1」は、接触位置を指定する引数である。「CF」は、コマンドの実行終了後も力制御モードの継続を指定する引数である。コマンド「Move P1 FC1 CF」は、「Force Control Object」クラスの引数オブジェクト「FC1」が指定されており、目標位置が引数「P1」で指定されているため、コマンド「Move P1 FC1 CF」が実行されると、力覚センサーFSがリセットされ、力覚センサーFSの出力に基づくハイブリッドモードでアームAが制御される。また引数「CF」が指定されているため、コマンドの実行終了後も力制御モードが継続する。

【0040】

比較例では、「ForceControl On」コマンドの実行により、力覚センサーFSがリセットされ、力覚センサーFSの出力に基づいてアームAを制御可能な状態となる。続いて「Move P1」コマンドの実行により、力覚センサーFSの出力に基づいてアームAが制御される。これに対して、本実施例のコマンド「Move P1 FC1 CF」の実行が開始されてから力覚センサーFSがリセットされるまでの実時間は、対応する機械語プログラムが常に同じになるため、制御部32のクロック数によって予め定まる一定の時間となる。したがって上述したように、本実施例では力制御の再現性が保証される。

10

【0041】

(3) 他の実施形態

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限りにおいて変更を加え得ることは勿論である。例えば、上記実施例ではコマンドの引数で力制御モードのオンを指定できたが、コマンドの本体で力制御モードのオンを指定できてもよい。具体的にはコマンドの本体自体が、力制御モードをオンするものと、力制御モードをオンしないものとに分かれているプログラム言語をロボットシステムに実装しても良い。

20

【0042】

また力制御対応コマンドにより力覚センサーをリセットしてから力制御によりアームを動作させるまでの時間をタイマーで制御しても良い。すなわち、力制御対応コマンドの実行手順にタイマーのセットを組み込んでも良い。

また力検出器の出力に基づいてマニピュレーターの動作を制御する例としてインピーダンス制御を取り上げたが、運動方程式を用いずに、例えば力検出器の出力に対して線形な制御量でマニピュレーターの動作を制御しても良い。

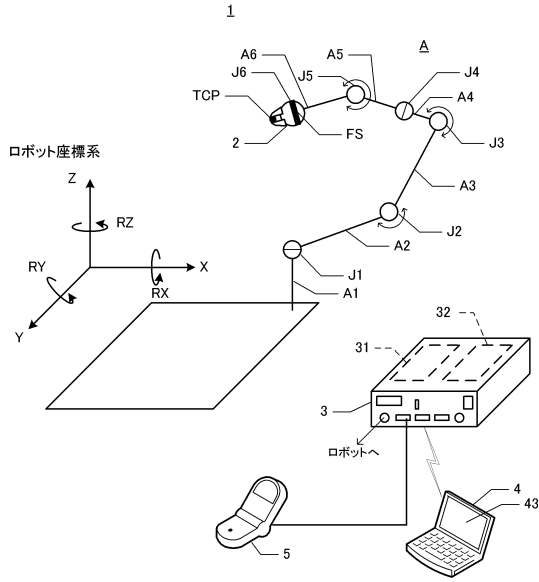
30

【符号の説明】

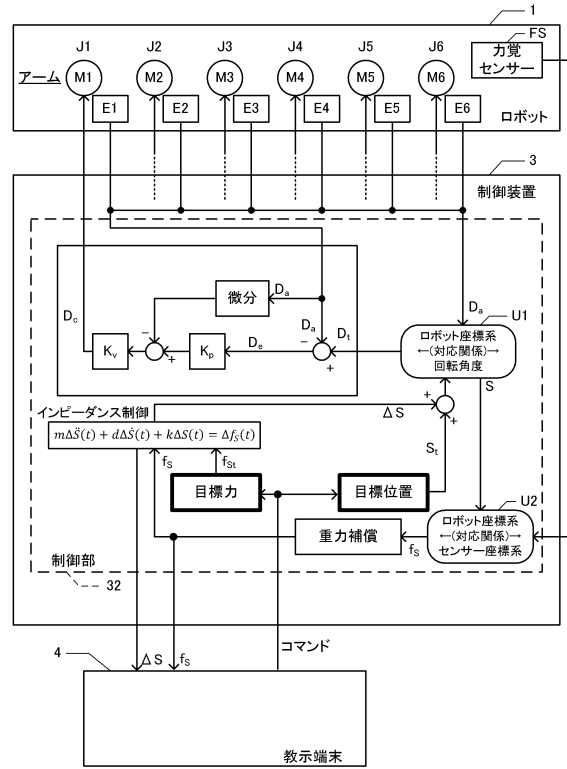
【0043】

1 ... ロボット、 2 ... エンドエフェクター、 3 ... 制御装置、 4 ... 教示装置、 5 ... ティーチングペンダント、 31 ... 電源部、 32 ... 制御部、 43 ... ディスプレイ、 A ... アーム、 A1 - A6 ... アーム部材、 E1 - E6 ... エンコーダー、 FS ... 力覚センサー、 J1 - J6 ... 関節、 M1 - M6 ... モーター

【図1】



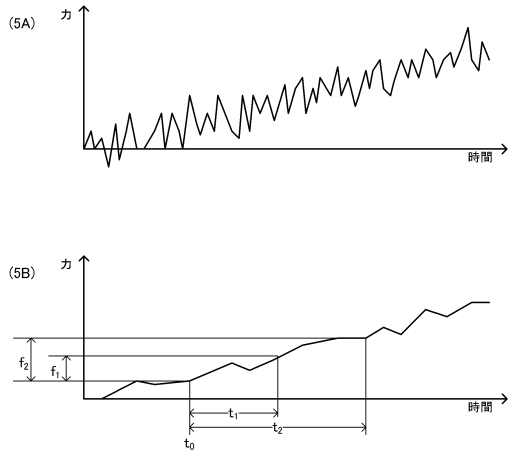
【図2】



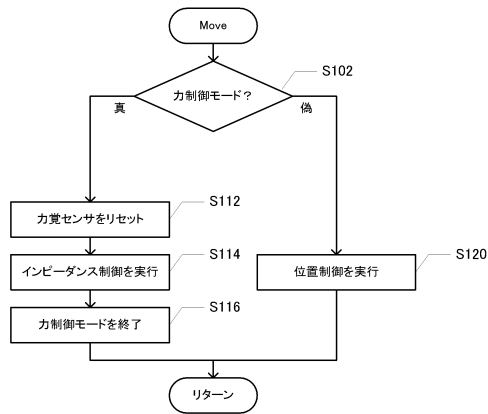
【図3】

| | 力制御対応コマンド | 位置制御コマンド |
|--------|---|---|
| CP | Move, TMove, BMove, CVMove, Arc, Arc3, FCSSMove | Move, TMove, BMove, CVMove, Arc, Arc3, FCSSMove |
| PTP | | Pass, Pulse, Go, TGo, BGo, JTran, PTran |
| CP+PTP | | Jump, Jump3 |
| その他 | FCKeep | Jump3CP |

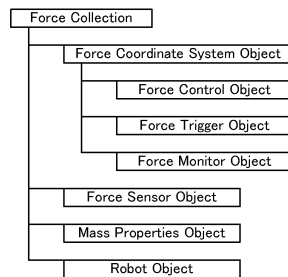
【図5】



【図4】



【図6】



【 図 7 】

【 図 8 】

力覚座標系の設定

力覚座標4 設定項目

| 番号 | ラベル | TCPからの相対位置 | [ミリ] |
|----|------------|------------|-------|
| 1 | ハンド面1先端 | x | 0 |
| 2 | ハンド面2ワーク側面 | y | 30 |
| 3 | | z | 20 |
| 4 | ハンド面3先端 | TCPとの相対姿勢 | |
| | | モード | TCP一致 |

力制御設定

力制御2 設定項目

| 番号 | ラベル | 力座標系選択 | 4 |
|----|--------|--------|------------------------------|
| 1 | 接触まで | 有効 | True,True,True, False, False |
| 2 | 押し付け状態 | 仮想バネ | 0.0,0.0,0.0 |
| 3 | | 仮想粘性 | 3.3,3.3,3.3 |
| 4 | 姿勢微い | 仮想質量 | 1.1,1.1,1.1 |

カトリガー設定

カトリガー1 設定項目

| 番号 | ラベル | 力座標系選択 | 4 |
|----|--------|----------|--|
| 1 | 接触検知 | 外側がTrue? | True,True,True, False, False |
| 2 | 緊急停止用 | 有効 | True, True, True, False, False, True, True |
| 3 | | 並進方向 上限 | 80,80,80 |
| 4 | 微い開始判定 | 並進方向 下限 | -80,-80,-80 |
| | | 回転方向 上限 | 2000,2000,2000 |
| | | 回転方向 下限 | -2000,-2000,-2000 |
| | | カトルク | 80,2000 |

```

<比較例>
ForceControlSetting0.1,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
0.3,3.3,1.1,1.3,3.3,1.1,1

ForceControl On
Move P1
ForceControlSetting0.1,0.0,0.0,-
0.1,0.0,0.0,0.0,0.3,3.3,1.1,1.3,3.3,1.1,1

<実施例>
Fset FC1.Fz,Enabled, True
Fset FC1.Fz,targetForce, 0
Fset FC1.Fz,Spring, 0
Fset FC1.Fz,Damper, 3
Fset FC1.Fz,Mass, 3

Move P1 FC1 OF
Fset FC1.Fz,TargetForce,-0.1 * RStdGravity
  
```

```

<処理内容>
Fz方向=有効
Fz方向の伸付力=0
Fz仮想バネ=0
Fz仮想粘性=3
Fz仮想質量=3
接触位置まで力制御で移動

Fz方向の伸付力を-0.1kgfに変更
  
```

フロントページの続き

- (72)発明者 元 吉 正樹
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
- (72)発明者 竹内 馨
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
- (72)発明者 五十嵐 克司
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

審査官 貞光 大樹

- (56)参考文献 特開平5 - 150826 (JP, A)
特開平6 - 315881 (JP, A)
特開2000 - 308985 (JP, A)
植之原 道宏, 位置不確定性存在下でのオフライン教示システムの一構成法, 日本ロボット学会誌, 日本, 社団法人日本ロボット学会, 1996年10月15日, 第14巻 第7号, 第139 - 147ページ

- (58)調査した分野(Int.Cl., DB名)
B25J 1/00 - 21/02