

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6651707号  
(P6651707)

(45) 発行日 令和2年2月19日 (2020.2.19)

(24) 登録日 令和2年1月27日 (2020.1.27)

(51) Int.Cl. F I  
**B 2 5 J** 9/10 (2006.01) B 2 5 J 9/10 A

請求項の数 9 (全 31 頁)

(21) 出願番号	特願2015-80904 (P2015-80904)	(73) 特許権者	000002369
(22) 出願日	平成27年4月10日 (2015.4.10)		セイコーエプソン株式会社
(65) 公開番号	特開2016-198855 (P2016-198855A)		東京都新宿区新宿四丁目1番6号
(43) 公開日	平成28年12月1日 (2016.12.1)	(73) 特許権者	504157024
審査請求日	平成30年1月16日 (2018.1.16)		国立大学法人東北大学
			宮城県仙台市青葉区片平二丁目1番1号
		(74) 代理人	100146835
			弁理士 佐伯 義文
		(74) 代理人	100064908
			弁理士 志賀 正武
		(74) 代理人	100140774
			弁理士 大浪 一徳
		(72) 発明者	南本 高志
			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

最終頁に続く

(54) 【発明の名称】 ロボット、制御装置、及び制御方法

(57) 【特許請求の範囲】

【請求項1】

アームと、

前記アームを動作させる制御部と、を含み、

前記制御部は、第2物体と、前記アームとともに動く第1物体との相対位置姿勢に基づいて、前記アームにより作業を行うことが可能な空間内の各点において規定される量のうち、予め決められた並進方向へ他の物体と干渉せずに並進可能な距離と、予め決められた回転方向へ他の物体と干渉せずに回転可能な回転角との少なくとも一方によって表される量を、前記第1物体の動かし難さ又は動かし易さの指標となる量として算出し、算出された前記指標となる量と前記第1物体の第1位置とに基づいて経路を生成する、

ロボット。

【請求項2】

前記制御部は、前記第1位置と前記第1物体の第2位置とを用いて前記経路を生成する、

請求項1に記載のロボット。

【請求項3】

前記制御部は、更に前記第1位置に対応する第1姿勢と、前記第2位置に対応する第2姿勢とを用いて前記経路を生成する、

請求項2に記載のロボット。

【請求項4】

10

20

前記制御部は、前記相対位置姿勢と前記第 1 位置に基づいて前記アームを移動させる速さを決定する、

請求項 1 から 3 のうちいずれか一項に記載のロボット。

【請求項 5】

前記制御部は、前記相対位置姿勢に基づく前記距離と前記回転角との少なくとも一方によって表される量を用いて前記速さを決定する、

請求項 4 に記載のロボット。

【請求項 6】

前記制御部は、前記第 1 物体の第 3 位置と前記第 1 物体の第 4 位置とを用いて前記速さを決定する、

請求項 5 に記載のロボット。

【請求項 7】

前記制御部は、更に前記第 3 位置に対応する第 3 姿勢と、前記第 4 位置に対応する第 4 姿勢とを用いて前記速さを決定する、

請求項 6 に記載のロボット。

【請求項 8】

第 2 物体と、ロボットが備えるアームとともに動く第 1 物体との相対位置姿勢に基づいて、前記アームにより作業を行うことが可能な空間内の各点において規定される量のうち、予め決められた並進方向へ他の物体と干渉せずに並進可能な距離と、予め決められた回転方向へ他の物体と干渉せずに回転可能な回転角との少なくとも一方によって表される量を、前記第 1 物体の動かし難さ又は動かし易さの指標となる量として算出し、算出された前記指標となる量と前記第 1 物体の第 1 位置とに基づいて経路を生成する、

制御装置。

【請求項 9】

第 2 物体と、ロボットが備えるアームとともに動く第 1 物体との相対位置姿勢に基づいて、前記アームにより作業を行うことが可能な空間内の各点において規定される量のうち、予め決められた並進方向へ他の物体と干渉せずに並進可能な距離と、予め決められた回転方向へ他の物体と干渉せずに回転可能な回転角との少なくとも一方によって表される量を、前記第 1 物体の動かし難さ又は動かし易さの指標となる量として算出し、算出された前記指標となる量と前記第 1 物体の第 1 位置とに基づいて経路を生成する、

制御方法。

【発明の詳細な説明】

【技術分野】

【0001】

この発明は、ロボット、制御装置、及び制御方法に関する。

【背景技術】

【0002】

ロボットに物体同士の組み付けを行わせる技術の研究や開発が行われている。

【0003】

これに関し、部品把持手段を備える組立機構を用いて複数の部品を順次組みつけて所定の組立対象物を構成する自動組立システムが知られている（特許文献 1 参照）。

【先行技術文献】

【特許文献】

【0004】

【特許文献 1】特開 2006 - 350620 号公報

【発明の概要】

【発明が解決しようとする課題】

【0005】

しかしながら、この自動組立システムは、部品を移動させる速さを部品同士の相対的な位置及び姿勢に基づいて変更することができず、組み立て時間の短縮を図ることが困難で

10

20

30

40

50

あった。また、この結果により、部品同士を組み立てる動作を含む作業に要する時間が長くなってしまう場合があった。

【課題を解決するための手段】

【0006】

上記課題の少なくとも一つを解決するために本発明の一態様は、アームと、前記アームを動作させる制御部と、を含み、前記制御部は、第2物体と、前記アームとともに動く第1物体との相対位置姿勢、及び前記第1物体の第1位置に基づいて経路を生成する、ロボットである。

この構成により、ロボットは、第2物体と、アームとともに動く第1物体との相対位置姿勢、及び第1物体の第1位置に基づいて経路を生成する。これにより、ロボットは、第1物体と第2物体との相対位置姿勢、及び第1物体の第1位置に基づく経路に沿って第1物体を移動させることができる。その結果、ロボットは、例えば、他の物体との干渉の抑制及び第1物体を移動させる時間の短縮を図ることができる。

【0007】

また、本発明の他の態様は、ロボットにおいて、前記制御部は、前記相対位置姿勢に基づく前記第1物体の移動可能な範囲を用いて前記経路を生成する、構成が用いられてもよい。

この構成により、ロボットは、第1物体と第2物体との相対位置姿勢に基づく第1物体の移動可能な範囲を用いて経路を生成する。これにより、ロボットは、第1物体の移動可能な範囲を用いて生成した経路に沿って第1物体を移動させることができる。

【0008】

また、本発明の他の態様は、ロボットにおいて、前記制御部は、前記移動可能な範囲に基づいて、前記相対位置姿勢に応じた前記第1物体の動かし難さ又は動かし易さの指標となる量を算出し、算出された前記指標となる量を用いて前記経路を生成する、構成が用いられてもよい。

この構成により、ロボットは、第1物体の移動可能な範囲に基づいて、第1物体と第2物体との相対位置姿勢に応じた第1物体の動かし難さ又は動かし易さの指標となる量を算出し、算出された指標となる量を用いて経路を生成する。これにより、ロボットは、第1物体の動かし難さ又は動かし易さの指標となる量を用いて生成した経路に沿って第1物体を移動させることができる。

【0009】

また、本発明の他の態様は、ロボットにおいて、前記制御部は、前記第1位置と前記第1物体の第2位置とを用いて前記経路を生成する、構成が用いられてもよい。

この構成により、ロボットは、第1物体の第1位置と第1物体の第2位置とを用いて経路を生成する。これにより、ロボットは、第1物体の第1位置と第1物体の第2位置とを用いて生成した経路に沿って第1物体を移動させることができる。

【0010】

また、本発明の他の態様は、ロボットにおいて、前記制御部は、更に前記第1位置に対応する第1姿勢と、前記第2位置に対応する第2姿勢とを用いて前記経路を生成する、構成が用いられてもよい。

この構成により、ロボットは、更に第1物体の第1位置に対応する第1姿勢と、第1物体の第2位置に対応する第2姿勢とを用いて経路を生成する。これにより、ロボットは、第1物体の第1位置に対応する第1姿勢と、第1物体の第2位置に対応する第2姿勢とを用いて生成した経路に沿って第1物体を移動させることができる。

【0011】

また、本発明の他の態様は、ロボットにおいて、前記制御部は、前記相対位置姿勢と前記所定位置に基づいて前記アームを移動させる速さを決定する、構成が用いられてもよい。

この構成により、ロボットは、第1物体と第2物体との相対位置姿勢と第1物体の第1位置に基づいてアームを移動させる速さを決定する。これにより、ロボットは、第1物体

10

20

30

40

50

と第2物体との相対位置姿勢と第1物体の第1位置に基づいて決定した速さによって第1物体を移動させることができる。

【0012】

また、本発明の他の態様は、ロボットにおいて、前記制御部は、前記相対位置姿勢に基づく前記第1物体の移動可能な範囲を用いて前記速さを決定する、構成が用いられてもよい。

この構成により、ロボットは、第1物体と第2物体との相対位置姿勢に基づく第1物体の移動可能な範囲を用いてアームを移動させる速さを決定する。これにより、ロボットは、第1物体の移動可能な範囲を用いて決定した速さによって第1物体を移動させることができる。

10

【0013】

また、本発明の他の態様は、ロボットにおいて、前記制御部は、前記相対位置姿勢に応じた前記第1物体の動かし難さ又は動かし易さの指標となる量を算出し、算出された前記指標となる量を用いて前記速さを決定する、構成が用いられてもよい。

この構成により、ロボットは、第1物体と第2物体との相対位置姿勢に応じた第1物体の動かし難さ又は動かし易さの指標となる量を算出し、算出された指標となる量を用いてアームを移動させる速さを決定する。これにより、ロボットは、第1物体の動かし難さ又は動かし易さの指標となる量を用いて決定した速さによって第1物体を移動させることができる。

【0014】

20

また、本発明の他の態様は、ロボットにおいて、前記制御部は、前記第1物体の第3位置と前記第1物体の第4位置とを用いて前記速さを決定する、構成が用いられてもよい。

この構成により、ロボットは、第1物体の第3位置と第1物体の第4位置とを用いてアームを移動させる速さを決定する。これにより、ロボットは、第1物体の第3位置と第1物体の第4位置とを用いて決定した速さによって第1物体を移動させることができる。

【0015】

また、本発明の他の態様は、ロボットにおいて、前記制御部は、更に前記第3位置に対応する第3姿勢と、前記第4位置に対応する第4姿勢とを用いて前記速さを決定する、構成が用いられてもよい。

この構成により、ロボットは、更に第1物体の第3位置に対応する第3姿勢と、第1物体の第4位置に対応する第4姿勢とを用いてアームを移動させる速さを決定する。これにより、ロボットは、第1物体の第3位置に対応する第3姿勢と、第1物体の第4位置に対応する第4姿勢とを用いて決定した速さによって第1物体を移動させることができる。

30

【0016】

また、本発明の他の態様は、第2物体と、ロボットが備えるアームとともに動く第1物体との相対位置姿勢、及び前記第1物体の第1位置に基づいて経路を生成する、制御装置である。

この構成により、制御装置は、第2物体と、ロボットが備えるアームとともに動く第1物体との相対位置姿勢、及び第1物体の第1位置に基づいて経路を生成する。これにより、制御装置は、第1物体と第2物体との相対位置姿勢、及び第1物体の第1位置に基づく経路に沿って第1物体を移動させることができる。その結果、制御装置は、例えば、他の物体との干渉の抑制及び第1物体を移動させる時間の短縮を図ることができる。

40

【0017】

また、本発明の他の態様は、第2物体と、ロボットが備えるアームとともに動く第1物体との相対位置姿勢、及び前記第1物体の第1位置に基づいて経路を生成する、制御方法である。

この構成により、制御方法は、第2物体と、ロボットが備えるアームとともに動く第1物体との相対位置姿勢、及び第1物体の第1位置に基づいて経路を生成する。これにより、制御方法は、第1物体と第2物体との相対位置姿勢、及び第1物体の第1位置に基づく経路に沿って第1物体を移動させることができる。その結果、制御方法は、例えば、他の

50

物体との干渉の抑制及び第 1 物体を移動させる時間の短縮を図ることができる。

【 0 0 1 8 】

以上により、ロボット、制御装置、及び制御方法は、第 2 物体と、アームとともに動く第 1 物体との相対位置姿勢、及び第 1 物体の第 1 位置に基づいて経路を生成する。これにより、ロボット、制御装置、及び制御方法は、第 1 物体と第 2 物体との相対位置姿勢、及び第 1 物体の第 1 位置に基づく経路に沿って第 1 物体を移動させることができる。その結果、ロボット、制御装置、及び制御方法は、例えば、他の物体との干渉の抑制及び第 1 物体を移動させる時間の短縮を図ることができる。

【図面の簡単な説明】

【 0 0 1 9 】

10

【図 1】第 1 実施形態に係るロボットシステム 1 の一例を示す構成図である。

【図 2】本実施形態における作業領域に沿ってロボット 20 が操作対象 N を組付対象 O に組み付ける様子の一例を示す図である。

【図 3】制御装置 30 のハードウェア構成の一例を示す図である。

【図 4】制御装置 30 の機能構成の一例を示す図である。

【図 5】制御装置 30 が操作対象 N の経路ポテンシャルを算出し、所定の作業を行う処理の流れの一例を示すフローチャートである。

【図 6】図 5 に示したステップ S 100 の可動領域算出処理における並進可動領域算出処理の流れの一例を示すフローチャートである。

【図 7】図 5 に示したステップ S 100 の可動領域算出処理における回転可動領域算出処理の流れの一例を示すフローチャートである。

20

【図 8】可動領域算出部 37 がステップ S 100 において選択した座標が示す仮想空間の点に配置された操作対象 N の一例を示す図である。

【図 9】図 5 に示したステップ S 110 の可動ポテンシャル算出処理の流れの一例を示すフローチャートである。

【図 10】可動ポテンシャル、並進可動ポテンシャル、及び回転可動ポテンシャルそれぞれの一例を表すグラフを示す図である。

【図 11】図 5 に示したステップ S 120 の距離ポテンシャル算出処理の流れの一例を示すフローチャートである。

【図 12】操作対象 N の組付位置を示す点と、ポテンシャル算出部 38 がステップ S 510 において選択した座標が示す仮想空間の点とにそれぞれ配置された操作対象 N の一例を示す図である。

30

【図 13】このようにして算出された並進距離ポテンシャルの一例を表すグラフを示す図である。

【図 14】ステップ S 130 において可動ポテンシャルと並進距離ポテンシャルを合算することによりポテンシャル算出部 38 が算出した経路ポテンシャルの一例を表すグラフを示す図である。

【図 15】平滑化処理部 39 が生成した滑らかな経路ポテンシャルの一例を表すグラフを示す図である。

【図 16】可動領域算出部 37 a による並進可動領域算出処理の流れの一例を示すフローチャートである。

40

【図 17】可動領域算出部 37 a による回転可動領域算出処理の流れの一例を示すフローチャートである。

【図 18】ポテンシャル算出部 38 a による距離ポテンシャル算出処理の流れの一例を示すフローチャートである。

【発明を実施するための形態】

【 0 0 2 0 】

< 第 1 実施形態 >

以下、本発明の第 1 実施形態について、図面を参照して説明する。図 1 は、第 1 実施形態に係るロボットシステム 1 の一例を示す構成図である。ロボットシステム 1 は、ロボッ

50

ト 20 と、制御装置 30 を備える。なお、ロボットシステム 1 は、ロボット 20 とは別体の撮像部を備える構成であってもよい。

【0021】

まず、ロボット 20 及び制御装置 30 の構成について説明する。

ロボット 20 は、アームと、力センサー 25 と、制御装置 30 を備えた単腕ロボットである。

単腕ロボットは、この一例における 1 本のアーム（腕）を備えるロボットである。なお、ロボット 20 は、単腕ロボットに代えて、双腕ロボットであってもよい。双腕ロボットは、2 本のアームを備えるロボットである。また、ロボット 20 は、撮像部を備える構成であってもよい。

10

【0022】

アームは、物体を把持可能な爪部を備えるエンドエフェクター E と、マニピュレーター M と、図示しない複数のアクチュエーターによって構成される。アームは、7 軸垂直多関節型のアームである。具体的には、アームは、支持台と、エンドエフェクター E と、マニピュレーター M とがアクチュエーターによる連携した動作によって 7 軸の自由度の動作を行う。

【0023】

アームが 7 軸の自由度で動作する場合、アームは、6 軸以下の自由度で動作する場合と比較して取り得る姿勢が増える。これによりアームは、例えば、動作が滑らかになり、更にアームの周辺に存在する物体との干渉を容易に回避することができるようになる。また、アームが 7 軸の自由度で動作する場合、アームの制御は、アームが 8 軸以上の自由度で動作する場合と比較して計算量が少なく容易である。このような理由から、アームは、7 軸の自由度で動作することが望ましい。それ故、この一例では、アームが 7 軸の自由度で動作する場合について説明する。なお、アームは、6 軸以下の自由度で動作する構成であってもよく、8 軸以上の自由度で動作する構成であってもよい。

20

【0024】

各アクチュエーターは、ケーブルによって制御装置 30 と通信可能に接続されている。これにより、アクチュエーターは、制御装置 30 から取得される制御信号に基づいて、エンドエフェクター E とマニピュレーター M を動作させることができる。なお、ケーブルを介した有線通信は、例えば、イーサネット（登録商標）や USB（Universal Serial Bus）等の規格によって行われる。また、アクチュエーターのうちの一部又は全部は、Wi-Fi（登録商標）等の通信規格により行われる無線通信によって制御装置 30 と接続される構成であってもよい。

30

【0025】

力センサー 25 は、エンドエフェクター E とマニピュレーター M の間に備えられている。力センサー 25 は、エンドエフェクター E に作用した力やモーメントの大きさを示す値を検出する。なお、力センサー 25 は、エンドエフェクター E に加わる力やモーメントの大きさを示す値を検出するトルクセンサー等の他のセンサーであってもよい。力センサー 25 は、通信により力センサー情報を制御装置 30 へ出力する。

【0026】

力センサー情報は、力センサー 25 が検出した力やモーメントの大きさを示す値を、力センサー 25 の出力値として含む情報である。また、力センサー情報は、制御装置 30 によるロボット 20 の力センサー情報に基づく制御に用いられる。力センサー情報に基づく制御とは、例えば、インピーダンス制御等のコンプライアンス制御を示す。

40

【0027】

力センサー 25 は、ケーブルによって制御装置 30 と通信可能に接続されている。ケーブルを介した有線通信は、例えば、イーサネット（登録商標）や USB 等の規格によって行われる。なお、力センサー 25 は、Wi-Fi（登録商標）等の通信規格により行われる無線通信によって制御装置 30 と接続されてもよい。

【0028】

50

上記で説明したロボット 20 が備えるこれらの各機能部は、この一例において、ロボット 20 に内蔵された制御装置 30 から制御信号を取得する。そして、これらの各機能部は、取得した制御信号に基づいた動作を行う。なお、ロボット 20 は、外部に設置された制御装置 30 により制御される構成に代えて、制御装置 30 を内蔵する構成であってもよい。

#### 【0029】

制御装置 30 は、ロボット 20 が備える各機能部に制御信号を送信することにより、ロボット 20 を動作させる。また、制御装置 30 は、ロボット 20 に所定の作業を行わせる。この一例において、制御装置 30 は、所定の作業として、ロボット 20 を動作させることにより第 1 物体を他の第 2 物体に組み付ける作業を行う。

10

#### 【0030】

以下、第 1 物体及び第 2 物体と、ロボット 20 が行う所定の作業とについて説明する。

第 1 物体は、例えば、図 1 に示したようにロボット 20 がエンドエフェクター E により把持している操作対象 N のことである。なお、第 1 物体は、操作対象 N に代えて、ロボット 20 のエンドエフェクター E の所定部位やマニピュレーター M の所定部位等であってもよい。操作対象 N は、この一例において、エンドエフェクター E の爪部により把持可能な程度の高さを有する六角柱形状の物体である。なお、操作対象 N は、これに代えて、他の形状や大きさであってもよい。

#### 【0031】

第 2 物体は、ロボット 20 により第 1 物体を組み付けられる物体のことである。なお、第 2 物体は、これに代えて、操作対象 N を移動させる際に障害となる物体や、何らかの治具、壁面、作業台等の位置及び姿勢がロボット 20 の座標原点の位置及び姿勢に対して変化しないように設置された物体であれば如何なる物体であってもよい。

20

#### 【0032】

第 2 物体は、例えば、図 1 に示した組付対象 O である。すなわち、組付対象 O は、ロボット 20 により操作対象 N が組み付けられる物体である。図 1 に示したように、この一例では、組付対象 O は、治具 F に支持されている。なお、組付対象 O は、これに代えて、テーブルや何らかの台等の他の場所に設置される構成であってもよい。その場合、組付対象 O の位置及び姿勢は、ロボット 20 の座標原点の位置及び姿勢に対して変化しないように設置される。ロボット 20 の座標原点の位置及び姿勢は、例えば、ロボット 20 の支持台の重心の位置及び姿勢である。なお、ロボット 20 の座標原点の位置及び姿勢は、これに代えて、ロボット 20 に対応付けられた他の位置及び姿勢であってもよい。

30

#### 【0033】

制御装置 30 は、操作対象 N と組付対象 O との相対的な位置及び姿勢、及び操作対象 N の第 1 位置に基づいて、操作対象 N を組付対象 O に組み付けるまでに操作対象 N を作業領域内において移動させる経路を生成する。以下では、一例として操作対象 N の第 1 位置が、ロボット 20 が行う所定の作業において操作対象 N が組付対象 O に組み付けられた後の操作対象 N の位置である場合について説明する。なお、操作対象 N と組付対象 O との相対的な位置及び姿勢は、第 1 物体と第 2 物体との相対位置姿勢の一例である。

#### 【0034】

また、制御装置 30 は、経路を生成する際、操作対象 N が組付対象 O や他の物体との干渉が起こる可能性が低い経路であり、且つ操作対象 N を速く移動させることが可能な経路を生成する。

40

#### 【0035】

これにより、制御装置 30 は、操作対象 N と組付対象 O との相対的な位置及び姿勢、及び操作対象 N が組付対象 O に組み付けられた後の操作対象 N の位置に基づいて生成する経路に沿って操作対象 N を移動させることができる。その結果、制御装置 30 は、他の物体との干渉の抑制及び第 1 物体を移動させる時間の短縮を図ることができる。なお、操作対象 N の第 1 位置は、作業領域内における操作対象 N の他の位置であってもよい。また、以下では、説明の便宜上、ロボット 20 が行う所定の作業において操作対象 N が組付対象 O

50

に組み付けられた後の操作対象Nの位置を組付位置と称して説明する。また、以下では、組付位置に対応する操作対象Nの姿勢を組付姿勢と称して説明する。

【0036】

また、以下では、操作対象Nの位置及び姿勢が、操作対象Nの重心の位置及び姿勢である場合について説明する。なお、操作対象Nの位置及び姿勢は、これに代えて、操作対象Nとともに動く他の点の位置及び姿勢であってもよい。

また、以下では、組付対象Oの位置及び姿勢は、組付対象Oの重心の位置及び姿勢である場合について説明する。なお、組付対象Oの位置及び姿勢は、これに代えて、組付対象Oとともに固定される他の点の位置及び姿勢であってもよい。

【0037】

また、この一例における作業領域は、ロボット20が操作対象Nを組付対象Oに組み付けるために操作対象Nの重心を移動させることが可能な経路を含む平面又は空間の一部又は全部の領域のことである。また、作業領域には、組付対象Oが含まれている。以下では、説明を簡略化するため、一例として作業領域が図2に示した二次元( $x$   $y$ )平面である場合について説明する。

【0038】

図2は、本実施形態における作業領域に沿ってロボット20が操作対象Nを組付対象Oに組み付ける様子の一例を示す図である。図2に示した $x$ 軸及び $y$ 軸は、組付対象Oを含む二次元平面(すなわち、作業領域)上の座標軸である。ロボット20は、例えば、エンドエフェクターEにより把持した操作対象Nを、図2に示した矢印(前述の経路)に沿って組付対象Oに組み付ける。

【0039】

本実施形態では、この組み付けの際、ロボット20は、図2に示した座標軸により示される二次元平面に沿ってのみ操作対象Nを移動する。なお、ロボット20は、二次元平面である作業領域に沿って操作対象Nを移動する構成に代えて、三次元( $x$   $y$   $z$ )空間内に広がる経路に沿って操作対象Nを移動する構成であってもよい。

【0040】

また、制御装置30は、操作対象Nと組付対象Oとの相対的な位置及び姿勢、及び操作対象Nの組付位置に基づいて、制御装置30が生成した経路上の各位置においてロボット20が操作対象Nを移動させる速さを決定する。

【0041】

そして、制御装置30は、生成した経路と決定した速さとに基づいてロボット20を動作させることにより、操作対象Nを組付対象Oに組み付ける。このように操作対象Nを組付対象Oに組み付けることにより、ロボット20は、組付対象Oや他の物体との干渉の抑制及び操作対象Nを移動させる時間の短縮を図ることができる。なお、図1に示した点線VNは、組付対象Oに組み付けられた状態の操作対象Nの輪郭を示している。

【0042】

なお、制御装置30は、コンプライアンス制御によってロボット20を動作させることにより、ロボット20に操作対象Nを組付対象Oに組み付けさせる。これにより、制御装置30は、操作対象Nと組付対象Oや他の物体との干渉が起きた場合であっても、例えば、操作対象Nや組付対象O、他の物体を変形させることなくロボット20に操作対象Nを組付対象Oに組み付けさせることができる。

【0043】

次に、図3を参照し、制御装置30のハードウェア構成について説明する。図3は、制御装置30のハードウェア構成の一例を示す図である。制御装置30は、例えば、CPU(Central Processing Unit)31と、記憶部32と、入力受付部33と、通信部34と、表示部35を備える。また、制御装置30は、通信部34を介してロボット20と通信を行う。これらの構成要素は、バスBusを介して相互に通信可能に接続されている。

【0044】

CPU31は、記憶部32に格納された各種プログラムを実行する。



記憶部 32 は、例えば、HDD (Hard Disk Drive) や SSD (Solid State Drive)、EEPROM (Electrically Erasable Programmable Read - Only Memory)、ROM (Read - Only Memory)、RAM (Random Access Memory) などを含む。記憶部 32 は、制御装置 30 が処理する各種情報や画像、プログラム、組付対象 O の位置及び姿勢を示す情報、所定の作業を行う際の作業領域を示す情報等を格納する。なお、記憶部 32 は、制御装置 30 に内蔵されるものに代えて、USB 等のデジタル入出力ポート等によって接続された外付け型の記憶装置でもよい。

【0045】

入力受付部 33 は、例えば、キーボードやマウス、タッチパッド等を備えたティーチングペンダントや、その他の入力装置である。なお、入力受付部 33 は、タッチパネルとして表示部 35 と一体に構成されてもよい。

10

通信部 34 は、例えば、USB 等のデジタル入出力ポートやイーサネット（登録商標）ポート等を含んで構成される。

表示部 35 は、例えば、液晶ディスプレイパネル、あるいは、有機 EL (ElectroLuminescence) ディスプレイパネルである。

【0046】

次に、図 4 を参照し、制御装置 30 の機能構成について説明する。図 4 は、制御装置 30 の機能構成の一例を示す図である。制御装置 30 は、記憶部 32 と、制御部 36 を備える。

【0047】

20

制御部 36 は、制御装置 30 の全体を制御する。制御部 36 は、可動領域算出部 37 と、ポテンシャル算出部 38 と、平滑化処理部 39 と、経路生成部 40 と、速さ決定部 41 と、力センサー取得部 42 と、ロボット制御部 43 を備える。制御部 36 が備えるこれらの機能部のうち一部又は全部は、例えば、CPU 31 が、記憶部 32 に記憶された各種プログラムを実行することで実現される。また、これらの機能部のうち一部または全部は、LSI (Large Scale Integration) や ASIC (Application Specific Integrated Circuit) 等のハードウェア機能部であってもよい。

【0048】

可動領域算出部 37 は、可動領域を算出する。可動領域は、作業領域内の各点において定義される量である。また、可動領域は、作業領域内において操作対象 N が組付対象 O や他の物体と接触せずに動くことができる範囲を定量的に示す量である。

30

【0049】

ポテンシャル算出部 38 は、可動領域算出部 37 が算出した可動領域に基づいて可動ポテンシャルを算出する。可動ポテンシャルは、作業領域内の各点において定義される量である。また、可動ポテンシャルは、作業領域内の各点における操作対象 N の動かし難さ（又は、動かし易さ）の指標となる量である。

【0050】

可動ポテンシャルは、作業領域内の各点において算出されるため、作業領域内の各点を示す座標の関数として表される。なお、可動ポテンシャルは、ポテンシャルと称しているが、物理量ではない。可動ポテンシャルをポテンシャルと称する理由は、作業領域の各点における操作対象 N の剛性、すなわちエンドエフェクター E により把持された操作対象 N の動かし難さ（もしくは、動かし易さ）の指標となる量を、空間の各点において物体に働く保存力に係るポテンシャルエネルギーになぞらえたためである。操作対象 N を動かし難い状態とは、この場合、操作対象 N を動かすことにより、操作対象 N と組付対象 O や他の物体とが干渉してしまう可能性が高い状態のことである。

40

【0051】

ここで、作業領域内における操作対象 N の位置及び姿勢は、作業領域内において組付対象 O の位置及び姿勢が変化しないため、操作対象 N と組付対象 O との相対的な位置及び姿勢を表す。すなわち、ポテンシャル算出部 38 は、操作対象 N と組付対象 O との相対的な位置及び姿勢に基づいて操作対象 N の動かし難さの指標となる量を、作業領域内の各点に

50

おける可動ポテンシャルとして算出する。

【 0 0 5 2 】

また、ポテンシャル算出部 3 8 は、作業領域内における操作対象 N の組付位置に基づいて距離ポテンシャルを算出する。距離ポテンシャルは、作業領域内の各点において定義される量である。また、距離ポテンシャルは、作業領域内の各点において、ロボット 2 0 が操作対象 N を移動させることが可能な速さの指標となる量のことである。例えば、距離ポテンシャルが低いところでは、当該速さを速くすることができ、距離ポテンシャルが高いところでは、当該速さを距離ポテンシャルが低いところに比べて遅くしなければならない。

【 0 0 5 3 】

距離ポテンシャルは、作業領域内の各点において算出されるため、作業領域内の各点を示す座標（すなわち、位置）の関数として表される。なお、距離ポテンシャルは、ポテンシャルと称しているが、物理量ではない。距離ポテンシャルをポテンシャルと称する理由は、作業領域の各点においてエンドエフェクター E により操作対象 N を移動させることが可能な速さの指標となる量を、空間の各点において物体を加速させる保存力に係るポテンシャルエネルギーになぞらえたためである。

【 0 0 5 4 】

また、ポテンシャル算出部 3 8 は、算出した可動ポテンシャルと距離ポテンシャルを合算した経路ポテンシャルを算出する。経路ポテンシャルは、ロボット 2 0 が操作対象 N を組付対象 O に組み付ける際に、操作対象 N を動かし易く、そして速く移動させることができる位置の指標となる量である。経路ポテンシャルは、可動ポテンシャル及び距離ポテンシャルが作業領域内の各点を示す座標（位置）の関数であるため、作業領域内の各点を示す座標（位置）の関数として表される。

【 0 0 5 5 】

平滑化処理部 3 9 は、ポテンシャル算出部 3 8 によって算出された経路ポテンシャルを表す関数に対して、ローパスフィルターを適用する。これにより、平滑化処理部 3 9 は、経路ポテンシャルを表す関数を平滑化（すなわち、平均化）する。以下では、説明の便宜上、平滑化処理部 3 9 が経路ポテンシャルを表す関数をこのように変換することを、滑らかなポテンシャルを生成すると称して説明する。また、以下では、ローパスフィルターを適用する前の経路ポテンシャルを表す関数を、不連続に変化する経路ポテンシャルと称して説明する。不連続に変化する経路ポテンシャルに基づいてロボット 2 0 を制御しようとすると、例えば、アクチュエーター等に負担が掛かり、故障の原因となりかねない。そのため、平滑化処理部 3 9 は、算出された不連続に変化する経路ポテンシャルにローパスフィルターを適用することで、滑らかなポテンシャルを生成する。

【 0 0 5 6 】

経路生成部 4 0 は、平滑化処理部 3 9 が平滑化した経路ポテンシャルに基づいて、ロボット 2 0 が操作対象 N を組付対象 O に組み付けるまでに移動させる経路を生成する。経路生成部 4 0 が生成する当該経路は、操作対象 N が組付対象 O や他の物体との干渉が起こる可能性が低い経路であり、且つ操作対象 N を速く移動させることが可能な経路である。

速さ決定部 4 1 は、平滑化処理部 3 9 が平滑化した経路ポテンシャルに基づいて、経路生成部 4 0 が生成した経路上の各点において操作対象 N をロボット 2 0 が移動させる速さを決定する。

【 0 0 5 7 】

力センサー取得部 4 2 は、力センサー 2 5 により検出される力センサー情報を取得する。

ロボット制御部 4 3 は、経路生成部 4 0 が生成した経路と、速さ決定部 4 1 が決定した当該経路上の各点において操作対象 N をロボット 2 0 が移動させる速さとに基づいて、ロボット 2 0 を動作させ、操作対象 N を組付対象 O に組み付ける。この際、ロボット制御部 4 3 は、力センサー取得部 4 2 が取得した力センサー情報に基づいて、コンプライアンス制御によりロボット 2 0 を動作させる。

10

20

30

40

50

## 【 0 0 5 8 】

以下、図 5 を参照し、制御装置 3 0 が経路ポテンシャルを算出し、所定の作業を行う処理について説明する。図 5 は、制御装置 3 0 が操作対象 N の経路ポテンシャルを算出し、所定の作業を行う処理の流れの一例を示すフローチャートである。以下では、一例として、予めロボット 2 0 がエンドエフェクター E により操作対象 N を把持している場合について説明する。従って、制御装置 3 0 は、ロボット 2 0 に操作対象 N を把持させてからロボット 2 0 に所定の作業を行わせる構成であってもよい。

## 【 0 0 5 9 】

まず、可動領域算出部 3 7 は、操作対象 N の可動領域を算出するための可動領域算出処理を行う（ステップ S 1 0 0）。この一例において、操作対象 N の可動領域は、作業領域を表す二次元平面上のある点から当該二次元平面内において組付対象 O と接触せずに操作対象 N が x 方向及び y 方向に移動可能な距離の総和（以下、並進可動領域と称する）と、当該点において組付対象 O と接触せずに正方向（例えば、反時計回り）及び負方向（例えば、時計回り）に回転可能な回転角の総和（以下、回転可動領域と称する）との 2 つの量によって定義される量であり、当該二次元平面上の各点において定義される量である。そのため、可動領域算出処理は、並進可動領域算出処理と回転可動領域算出処理の 2 つの処理によって構成される。

## 【 0 0 6 0 】

並進可動領域算出処理は、作業領域内の各点における操作対象 N の並進可動領域を算出する処理である。回転可動領域算出処理は、作業領域内の各点における操作対象 N の回転可動領域を算出する処理である。可動領域算出部 3 7 は、可動領域算出処理において並進可動領域算出処理と回転可動領域算出処理のいずれか一方を先に行ってもよく、両方を同時に行ってもよい。なお、可動領域算出部 3 7 は、可動領域算出処理において、並進可動領域算出処理と回転可動領域のうちのいずれか一方のみを行う構成（すなわち、操作対象 N の可動領域は、並進可動領域と回転可動領域のうちのいずれか一方によって定義される構成）であってもよい。

## 【 0 0 6 1 】

なお、操作対象 N の可動領域は、作業領域内の各点における操作対象 N に係る他の量によって定義されてもよい。また、操作対象 N の可動領域は、作業領域内の各点において定義される構成に代えて、例えば、作業領域を複数の領域に分割した場合のそれぞれの領域毎に定義される構成であってもよい。

## 【 0 0 6 2 】

次に、ポテンシャル算出部 3 8 は、ステップ S 1 0 0 において可動領域算出部 3 7 が算出した操作対象 N の並進可動領域と回転可動領域のうちのいずれか一方又は両方に基づいて、作業領域内の各点における可動ポテンシャルを算出するための可動ポテンシャル算出処理を行う（ステップ S 1 1 0）。本実施形態では、ポテンシャル算出部 3 8 は、可動ポテンシャル算出処理において、操作対象 N の並進可動領域と回転可動領域の両方に基づいて、作業領域内の各点における可動ポテンシャルを算出する場合について説明する。

## 【 0 0 6 3 】

次に、ポテンシャル算出部 3 8 は、作業領域内の各点における距離ポテンシャルを算出するための距離ポテンシャル算出処理を行う（ステップ S 1 2 0）。なお、ポテンシャル算出部 3 8 が行うステップ S 1 1 0 の処理とステップ S 1 2 0 の処理は、順序が逆に行われてもよく、並列に行われてもよい。

次に、ポテンシャル算出部 3 8 は、ステップ S 1 1 0 において算出した可動ポテンシャルと、ステップ S 1 2 0 において算出した距離ポテンシャルを合算した経路ポテンシャルを算出する（ステップ S 1 3 0）。

## 【 0 0 6 4 】

次に、平滑化処理部 3 9 は、ステップ S 1 3 0 においてポテンシャル算出部 3 8 が算出した不連続に変化する経路ポテンシャルにローパスフィルターを適用することで、滑らかに変化する経路ポテンシャルを生成（平滑化）する（ステップ S 1 4 0）。

次に、経路生成部 40 は、ステップ S 140 において平滑化処理部 39 が平滑化した経路ポテンシャルに基づいて、ロボット 20 が操作対象 N を組付対象 O に組み付けるまでに移動させる経路を生成する経路生成処理を行う（ステップ S 150）。

【0065】

次に、速さ決定部 41 は、ステップ S 140 において平滑化処理部 39 が平滑化した経路ポテンシャルに基づいて、ステップ S 150 において経路生成部 40 が生成した経路上の各点において操作対象 N をロボット 20 が移動させる速さを決定する速さ決定処理を行う（ステップ S 160）。

【0066】

次に、ロボット制御部 43 は、ステップ S 150 において経路生成部 40 が生成した経路と、ステップ S 160 において速さ決定部 41 が決定した速さであって、経路上の各点において操作対象 N をロボット 20 が移動させる速さとに基づいて、ロボット 20 に所定の作業を行わせる（ステップ S 170）。なお、ロボット制御部 43 は、ステップ S 170 において、力センサー取得部 42 が取得した力センサー情報に基づいて、コンプライアンス制御によってロボット 20 を動作させる。

【0067】

以下、図 6 及び図 7 を参照し、図 5 に示したステップ S 100 の可動領域算出処理について説明する。図 6 は、図 5 に示したステップ S 100 の可動領域算出処理における並進可動領域算出処理の流れの一例を示すフローチャートである。

【0068】

まず、可動領域算出部 37 は、組付対象 O の大きさや形状を示す情報と、組付対象 O の位置及び姿勢を示す情報と、作業領域を示す情報を記憶部 32 から読み込み、読み込まれた組付対象 O の位置及び姿勢を示す情報と、作業領域を示す情報とに基づいて作業領域を表す仮想空間を生成する（ステップ S 190）。

【0069】

なお、この一例において、組付対象 O の大きさや形状を示す情報と、組付対象 O の位置及び姿勢を示す情報と、作業領域を示す情報は、予め記憶部 32 に記憶されている場合について説明する。なお、作業領域を示す情報は、これに代えて、ロボットシステム 1 が撮像部を備え、撮像部により撮像された撮像画像に基づいて検出される構成であってもよい。また、この一例において、作業領域を示す情報は、作業領域の形状が図 2 に示したように長形状の平面であるため、当該長方形の各頂点を示す座標の組み合わせである。また、組付対象 O の位置及び姿勢を示す情報は、これに代えて、ロボットシステム 1 が撮像部を備え、撮像部により撮像された撮像画像に基づいて検出される構成であってもよい。

【0070】

ステップ S 190 において可動領域算出部 37 が生成した仮想空間内には、仮想的に組付対象 O が配置されている。仮想的に組付対象 O が配置されている仮想空間内での組付対象 O の位置及び姿勢は、現実の作業領域内の組付対象 O の位置及び姿勢に対応している。

【0071】

次に、可動領域算出部 37 は、ステップ S 190 において生成した仮想空間の各点を示す座標を 1 つずつ選択する。そして、可動領域算出部 37 は、選択した座標毎に、ステップ S 210 からステップ S 270 までの処理を繰り返し行う（ステップ S 200）。

【0072】

ステップ S 200 において仮想空間の座標を 1 つ選択した後、可動領域算出部 37 は、選択した座標が示す仮想空間の点に、仮想的に操作対象 N を配置する。この際、可動領域算出部 37 は、仮想空間の点に操作対象 N の重心が一致するように操作対象 N を配置する。また、可動領域算出部 37 は、仮想空間の点に配置した操作対象 N の姿勢を、所定の姿勢に初期化する（ステップ S 210）。なお、所定の姿勢とは、例えば、予め決められた操作対象 N の重心の位置を通る中心線と、x 軸又は y 軸との成す角度がゼロ度となるような姿勢であるが、これに限られず、他の姿勢であってもよい。

【0073】

次に、可動領域算出部 37 は、数値を示す情報を格納可能な変数 D を生成し、変数 D の値をゼロに初期化する（ステップ S 220）。

次に、可動領域算出部 37 は、ステップ S 260 における操作対象 N の回転に係る回転角の累積値が 360° に達したか否かを判定する（ステップ S 230）。

【0074】

操作対象 N の回転に係る回転角の累積値が 360° に達していないと判定した場合（ステップ S 230 - No）、可動領域算出部 37 は、ステップ S 200 において選択した座標が示す仮想空間の点に配置された操作対象 N が x 軸の正方向、x 軸の負方向、y 軸の正方向、y 軸の負方向のそれぞれの方向（以下、4 つの方向と称する）への移動可能な距離の和を算出する（ステップ S 240）。

10

【0075】

なお、可動領域算出部 37 が算出するこれらの移動可能な距離の和は、4 つ方向それぞれへの移動可能な距離の和に代えて、4 つの方向のうちのいずれか 1 つへの移動距離であってもよく、4 つの方向それぞれへの移動可能な距離の一部の組み合わせの和であってもよく、1 以上の他の方向への移動可能な距離の和（他の方向が 1 つのみの場合、和ではなく、その方向への移動可能な距離）であってもよく、上記の 4 つの方向それぞれへの移動可能な距離と他の方向への移動可能な距離との和であってもよい。例えば、可動領域算出部 37 は、操作対象 N が移動可能な距離の和として、1 以上の他の方向（例えば、x 軸の延伸方向と y 軸の延伸方向との間の方向（斜め方向）等）への移動可能な距離の和を算出する場合、それらの方向への距離を三平方の定理等、距離空間において距離を算出する方法によって算出する。

20

【0076】

次に、可動領域算出部 37 は、ステップ S 240 において算出した 4 つの方向への移動可能な距離の和を、ステップ S 220 において生成した変数 D に格納されている値に加算する（ステップ S 250）。

次に、可動領域算出部 37 は、ステップ S 200 において選択した座標が示す仮想空間の点に配置した操作対象 N の姿勢を、所定角度だけ回転させる（ステップ S 260）。なお、所定角度は、その倍数に 360° を含む任意の角度であることが望ましく、この一例において、5° や 10° 等であるとするが、その倍数に 360° を含まない他の角度であってもよい。

30

【0077】

ステップ S 230 からステップ S 260 までの処理をまとめて換言すると、可動領域算出部 37 は、ステップ S 200 において選択した座標が示す仮想空間の点に配置した操作対象 N の姿勢を所定角度ずつ変化させ、それぞれの姿勢毎に前述の 4 つの方向への移動可能な距離の和を算出し、算出された姿勢毎の 4 つの方向への移動可能な距離の和を変数 D に加算する。

【0078】

ここで、図 8 を参照し、ステップ S 230 からステップ S 260 までの処理について説明する。図 8 は、可動領域算出部 37 がステップ S 100 において選択した座標が示す仮想空間の点に配置された操作対象 N の一例を示す図である。図 8 において、座標 (x, y) を、可動領域算出部 37 がステップ S 200 において選択した座標であるとして説明する。点 G (x, y) は、可動領域算出部 37 が座標 (x, y) に配置した操作対象 N の重心の位置を示す。また、枠 VS は、可動領域算出部 37 が生成した仮想空間であって作業領域を表す仮想空間を示す。以下では、この枠 VS によって囲まれた範囲を仮想空間 VS と称して説明する。なお、図 8 に示した座標軸は、この仮想空間 VS 上の位置及び姿勢を表す座標軸である。また、以下では、図 8 において示した操作対象 N の姿勢が、所定の姿勢である場合について説明する。

40

【0079】

可動領域算出部 37 は、座標 (x, y) に配置した操作対象 N を前述の 4 つの方向のそれぞれに移動させる。この時、可動領域算出部 37 は、操作対象 N の輪郭が、仮想空間 V

50

Sの枠や組付対象Oの輪郭と干渉（すなわち、接触）するまで操作対象Nを移動させた時の移動距離を、4つの方向のそれぞれの方向に移動可能な距離として算出する。図8において、距離 $D_{+x}$ は、x軸の正方向への移動可能な距離であり、距離 $D_{-x}$ は、x軸の負方向への移動可能な距離である。また、図8において、距離 $D_{+y}$ は、y軸の正方向への移動可能な距離であり、距離 $D_{-y}$ は、y軸の負方向への移動可能な距離である。可動領域算出部37は、これらの距離 $D_{+x}$ 、距離 $D_{-x}$ 、距離 $D_{+y}$ 、距離 $D_{-y}$ の和を変数Dに格納された値に加算する。

#### 【0080】

そして、可動領域算出部37は、操作対象Nの姿勢を所定角度ずつ360°まで変化させ、所定角度変化させる毎にその時の姿勢において、このような4つの方向への移動可能な距離の和を算出し、算出された4つの方向への移動可能な距離の和を変数Dに格納された値に加算していく。この4つの方向への移動可能な距離の和は、操作対象Nの位置及び姿勢に応じて異なる値を取る。従って、変数Dに加算されていく値は、操作対象Nの位置及び姿勢に応じた値となる。このようにして、可動領域算出部37は、ステップS230からステップS260までの処理を行う。

#### 【0081】

一方、ステップS230において操作対象Nの回転に係る回転角の累積値が360°に達していると判定した場合（ステップS230 - Yes）、可動領域算出部37は、変数Dに格納されている値を、ステップS100で選択した座標における並進可動領域 $F_t$ とし（ステップS270）、その後、ステップS200に戻って仮想空間の次の座標を選択する。以下では、仮想空間VS内での各点を示す座標を座標 $(x, y)$ とした場合、その座標における並進可動領域 $F_t$ の値を並進可動領域 $F_t(x, y)$ と表す。なお、この一例では、仮想空間の各点を示す座標と、作業領域の各点を示す座標とは、キャリブレーション等によって予め対応付けられている場合について説明する。そのため、仮想空間VS内の各点は、作業領域内の各点を示す。

#### 【0082】

図7は、図5に示したステップS100の可動領域算出処理における回転可動領域算出処理の流れの一例を示すフローチャートである。なお、図7に示したフローチャートにおいて、ステップS190からステップS210までの処理は、図6に示したフローチャートにおけるステップS190からステップS210までの処理と同様の処理であるため、説明を省略する。

#### 【0083】

ただし、可動領域算出部37は、図7に示したステップS200において選択した座標毎に、図7に示したフローチャートのステップS210からステップS360までの処理を繰り返し行う。また、可動領域算出部37により回転可動領域算出処理が図6に示した並進可動領域算出処理の後に（又は並進可動領域算出処理と並列に）行われる場合、すでに仮想空間が生成されているため、図7に示したステップS190において新たに仮想空間を生成しなくてもよい。

#### 【0084】

図7に示したステップS210の処理を実行した後、可動領域算出部37は、数値を示す情報を格納可能な変数Hを生成し、変数Hの値をゼロに初期化する（ステップS320）。

次に、可動領域算出部37は、ステップS300において選択した座標が示す仮想空間の点に配置した操作対象Nが、図8に示した仮想空間VSの枠や組付対象Oの輪郭に現時点で干渉（すなわち、接触）しているか否かを判定する（ステップS330）。

#### 【0085】

操作対象Nが仮想空間VSの枠や組付対象Oの輪郭に接触していないと判定した場合（ステップS330 - No）、可動領域算出部37は、ステップS200において選択した座標が示す仮想空間の点に配置した操作対象Nを、例えば、操作対象Nの重心を中心として正方向と負方向のそれぞれに、操作対象Nの輪郭が仮想空間VSの枠や組付対象Oの輪

10

20

30

40

50

郭に接触するまで回転させる。ただし、操作対象Nの輪郭が仮想空間VSの枠や組付対象Oの輪郭に接触しない場合、正方向と負方向のそれぞれの回転角は、 $360^\circ$ であるとする。可動領域算出部37は、正方向への回転角と、負方向への回転角との和を回転可能角度として算出し（ステップS340）、変数Hに格納されている値に加算する（ステップS350）。

#### 【0086】

一方、ステップS330において操作対象Nが仮想空間VSの枠や組付対象Oの輪郭に干渉（すなわち、接触）していると判定した場合（ステップS330 - Yes）、可動領域算出部37は、変数Hに格納されている値を、回転可動領域 $F_r$ とし（ステップS360）、その後、ステップS200に戻って次の仮想空間の座標を選択する。以下では、仮想空間VS内での各点を示す座標を座標 $(x, y)$ とした場合、その座標における回転可動領域 $F_r$ の値を回転可動領域 $F_r(x, y)$ と表す。

#### 【0087】

以下、図9を参照し、図5に示したステップS110の可動ポテンシャル算出処理について説明する。図9は、図5に示したステップS110の可動ポテンシャル算出処理の流れの一例を示すフローチャートである。まず、ポテンシャル算出部38は、ステップS100で可動領域算出部37により算出された仮想空間VS内の各点（すなわち、作業領域内の各点）における可動領域、すなわち並進可動領域 $F_t(x, y)$ 及び回転可動領域 $F_r(x, y)$ に基づいて、並進可動領域 $F_t(x, y)$ の全ての点の中での最大値 $F_{t\_MAX}$ と、回転可動領域 $F_r(x, y)$ の全ての点の中での最大値 $F_{r\_MAX}$ を検出する（ステップS400）。

#### 【0088】

次に、ポテンシャル算出部38は、ステップS400において検出した並進可動領域 $F_t(x, y)$ の最大値 $F_{t\_MAX}$ と、回転可動領域 $F_r(x, y)$ の最大値 $F_{r\_MAX}$ とに基づいて、作業領域内の各点におけるポテンシャルを算出する（ステップS410）。ここで、このポテンシャルを算出する処理について説明する。ポテンシャル算出部38は、以下に示す式（1）～式（3）に基づいて作業領域内の各点における可動ポテンシャルを算出する。

#### 【0089】

$$P_t(x, y) = F_{t\_MAX} - F_t(x, y) \quad \cdots (1)$$

#### 【0090】

$$P_r(x, y) = F_{r\_MAX} - F_r(x, y) \quad \cdots (2)$$

#### 【0091】

$$P_{t+r}(x, y) = P_t(x, y) + P_r(x, y) \quad \cdots (3)$$

#### 【0092】

ここで、 $P_t(x, y)$ は、作業領域内のある点を示す座標 $(x, y)$ における可動ポテンシャルであり、並進可動領域に基づいて算出される可動ポテンシャル（以下、並進可動ポテンシャルと称する）を表す。また、 $P_r(x, y)$ は、作業領域内のある点を示す座標 $(x, y)$ における可動ポテンシャルであり、回転可動領域に基づいて算出される可動ポテンシャル（以下、回転可動ポテンシャルと称する）を表す。また、 $P_{t+r}(x, y)$ は、作業領域内のある点を示す座標 $(x, y)$ における可動ポテンシャルであり、並進可動ポテンシャルと回転可動ポテンシャルを合算することにより算出される可動ポテンシャルである。以下の説明においては、単に可動ポテンシャルと称した場合、上記の式（3）によって算出される $P_{t+r}(x, y)$ を示す。

#### 【0093】

上記の式（1）は、並進可動領域の最大値 $F_{t\_MAX}$ から、作業領域内のある点を示す座標 $(x, y)$ における並進可動領域 $F_t(x, y)$ を減算した値を、座標 $(x, y)$ における並進可動ポテンシャルとして定義する式である。このように並進可動ポテンシャルを定義することで、作業領域内の点のうち並進可動ポテンシャルが大きな点では、エンドエフェクターEにより把持された操作対象Nの剛性が大きい、すなわち操作対象Nを動か

10

20

30

40

50

し難い（並進させ難い）ことを表すことができる。このように操作対象Nを動かし難いということは、換言すると、操作対象Nを並進させることによって組付対象Oや他の物体と干渉しやすいことを示している。

【0094】

また、上記の式（2）は、回転可動領域の最大値  $F_{r\text{MAX}}$  から、作業領域内のある点を示す座標（ $x, y$ ）における回転可動領域  $F_r(x, y)$  を減算した値を、座標（ $x, y$ ）における回転可動ポテンシャルとして定義する式である。このように回転可動ポテンシャルを定義することで、作業領域内の点のうち回転可動ポテンシャルが大きな点では、エンドエフェクターEにより把持された操作対象Nの剛性が大きい、すなわち操作対象Nを動かし難い（回転させ難い）ことを表すことができる。このように操作対象Nを動かし

10

【0095】

すなわち、上記の式（3）により算出される可動ポテンシャル  $P_{t+r}(x, y)$  が大きな点は、エンドエフェクターEにより把持された操作対象Nの剛性が大きい、すなわち操作対象Nを動かし難い（並進又は回転させ難い）ことを表す。このように操作対象Nを動かし難いということは、換言すると、操作対象Nを並進又は回転させることによって組付対象Oや他の物体と干渉しやすいことを示している。

【0096】

図10（A）は、このようにして算出された可動ポテンシャルの一例を表すグラフを示す図である。このグラフから、ポテンシャル算出部38が算出した可動ポテンシャルは、その大きさが不連続に変化することが分かる。これは、可動ポテンシャルを算出するために用いる並進可動ポテンシャル及び回転可動ポテンシャルの大きさがそれぞれ不連続に変化するためである。

20

【0097】

ここで、可動ポテンシャルに関連して、図10（A）の場合以外の例を示しておく。

図10（B）は並進可動ポテンシャルの一例を表すグラフを示す図であり、図10（C）は回転可動ポテンシャルの一例を表すグラフを示す図である。なお、図10（A）では並進可動ポテンシャルと回転可動ポテンシャルとが合成された結果である可動ポテンシャルの例を示したが、ここでは、他の例として、並進可動ポテンシャルと回転可動ポテンシャルとのそれぞれの例を別々に示してある。前記したように、この例は、図10（A）の場合とは異なる例であり、図10（B）及び図10（C）に示される並進可動ポテンシャル及び回転可動ポテンシャルが合成された結果は、図10（A）に示される可動ポテンシャルとは異なる可動ポテンシャルとなる。図10（B）及び図10（C）に示される例において、並進可動ポテンシャル及び回転可動ポテンシャルの大きさは不連続に変化している。

30

【0098】

以下、図11を参照し、図5に示したステップS120の距離ポテンシャル算出処理について説明する。図11は、図5に示したステップS120の距離ポテンシャル算出処理の流れの一例を示すフローチャートである。なお、図11に示したフローチャートにおいて、ステップS190及びステップS200の処理は、図6に示したフローチャートにおけるステップS190及びステップS200の処理と同様の処理であるため、説明を省略する。

40

【0099】

ただし、図11に示したステップS190及びステップS200の処理を行う機能部は、図6に示したステップS190及びステップS200の処理を行う可動領域算出部37ではなく、ポテンシャル算出部38である。また、ポテンシャル算出部38は、図11に示したステップS200において選択した座標毎に、図11に示したフローチャートのステップS510の処理を繰り返し行う。また、ポテンシャル算出部38は、図5に示したステップS100において既に仮想空間が生成済みであるため、ステップS190の処理

50



を省略する構成であってもよい。

#### 【 0 1 0 0 】

図 1 1 に示したステップ S 2 0 0 において仮想空間の座標を選択した後、ポテンシャル算出部 3 8 は、操作対象 N の組付位置を示す情報を記憶部 3 2 から読み込む。なお、以下では、操作対象 N の組付位置を示す情報は、予め記憶部 3 2 に記憶されている場合について説明する。ポテンシャル算出部 3 8 は、読み込んだ操作対象 N の組付位置を示す情報と、ステップ S 2 0 0 において選択した座標とに基づいて、当該組付位置から当該座標が示す位置までの間の距離を、並進距離として算出する（ステップ S 5 1 0 ）。

#### 【 0 1 0 1 】

ここで、図 1 2 を参照し、ステップ S 5 1 0 の処理について説明する。図 1 2 は、操作対象 N の組付位置を示す点と、ポテンシャル算出部 3 8 がステップ S 5 1 0 において選択した座標が示す仮想空間の点とにそれぞれ配置された操作対象 N の一例を示す図である。図 1 2 において、点線 NS は、組付位置を示す点  $G(x_s, y_s)$  に配置された操作対象 N の輪郭を表している。また、実線 NC は、ポテンシャル算出部 3 8 がステップ S 5 1 0 において選択した座標が示す仮想空間の点  $G(x_c, y_c)$  に配置された操作対象 N の輪郭を表している。

#### 【 0 1 0 2 】

図 1 2 に示したように、ポテンシャル算出部 3 8 は点  $G(x_s, y_s)$  と点  $G(x_c, y_c)$  の間の距離  $L$  を、前述の並進距離として算出する。なお、図 1 2 では、ステップ S 5 1 0 の処理を説明するために仮想空間 VS 内に操作対象 N を配置した様子を例にとって説明したが、ポテンシャル算出部 3 8 は、仮想空間 VS 内に操作対象 N を配置せずに、操作対象 N の組付位置を示す点の仮想空間内における座標と、ポテンシャル算出部 3 8 がステップ S 5 1 0 において選択した座標とに基づいて並進距離を算出する構成であってもよい。以下では、仮想空間 VS 内での各点を示す座標を座標  $(x, y)$  とした場合、その座標における並進距離  $S_t$  の値を並進距離  $S_t(x, y)$  と表す。

#### 【 0 1 0 3 】

このように、ポテンシャル算出部 3 8 は、図 1 1 に示したステップ S 2 0 0 からステップ S 5 1 0 までの処理を繰り返し行うことにより、操作対象 N の組付位置と、仮想空間内の全ての点それぞれとの間の距離を、並進距離  $S_t(x, y)$  として算出する。

#### 【 0 1 0 4 】

次に、ポテンシャル算出部 3 8 は、ステップ S 5 1 0 において算出した並進距離  $S_t(x, y)$  に基づいて、並進距離  $S_t(x, y)$  の全ての点の中での最大値  $S_{tMAX}$  を検出する（ステップ S 5 2 0 ）。

#### 【 0 1 0 5 】

次に、ポテンシャル算出部 3 8 は、ステップ S 5 2 0 において検出した並進距離  $S_t(x, y)$  の最大値  $S_{tMAX}$  に基づいて、作業領域内の各点における距離ポテンシャルを算出する（ステップ S 5 3 0 ）。ここで、この距離ポテンシャルを算出する処理について説明する。ポテンシャル算出部 3 8 は、以下に示す式（ 4 ）に基づいて作業領域内の各点における距離ポテンシャルを算出する。

#### 【 0 1 0 6 】

$$PT_t(x, y) = S_{tMAX} - S_t(x, y) \quad \cdots (4)$$

#### 【 0 1 0 7 】

ここで、 $PT_t(x, y)$  は、作業領域内のある点を示す座標  $(x, y)$  における距離ポテンシャルであり、並進距離に基づいて算出される距離ポテンシャル（以下、並進距離ポテンシャルと称する）を表す。上記の式（ 4 ）は、並進距離の最大値  $S_{tMAX}$  から、作業領域内のある点を示す座標  $(x, y)$  における並進距離  $S_t(x, y)$  を減算した値を、座標  $(x, y)$  における並進距離ポテンシャルとして定義する式である。このように並進距離ポテンシャルを定義することで、作業領域内の点のうち並進距離ポテンシャルが大きな点は、操作対象 N が組付位置に近い、すなわち操作対象 N が組付対象 O に近いことを表すことができる。

## 【0108】

ロボット20は、例えば、操作対象Nと組付対象Oの間の他の物体が無い場合、操作対象Nが組付対象Oから遠い位置では、操作対象Nが組付対象Oに接触しないために速く操作対象Nを移動させることができる。また、ロボット20は、操作対象Nが組付対象Oに近い位置では、操作対象Nが組付対象Oに接触してしまう可能性が高いために遅く操作対象Nを移動させた方が望ましい。このことから、座標 $(x, y)$ における並進距離ポテンシャル $PT_t(x, y)$ の値は、座標 $(x, y)$ においてロボット20が操作対象Nを移動させることが可能な速さの指標とすることができる。

## 【0109】

図13は、このようにして算出された並進距離ポテンシャルの一例を表すグラフを示す図である。図13には、並進距離ポテンシャルの一例を表すグラフを示す。図13に示したグラフから、ポテンシャル算出部38が算出した並進距離ポテンシャルは、上に凸の形状のグラフとなり、操作対象Nの組付位置から遠ざかるほどに値が小さくなることが分かる。なお、図13に示したグラフにおける最大値は、操作対象Nの組付位置における並進距離ポテンシャルの値である。

## 【0110】

このように、ポテンシャル算出部38は、図11に示したステップS200からステップS530までの処理を実行することにより、並進距離ポテンシャルを算出することができる。

## 【0111】

図14は、ステップS130において可動ポテンシャルと並進距離ポテンシャルを合算することによりポテンシャル算出部38が算出した経路ポテンシャルの一例を表すグラフを示す図である。図10に示したように可動ポテンシャルの大きさが不連続に変化しているため、図14に示したようにポテンシャル算出部38が算出した経路ポテンシャルは、その大きさが不連続に変化する。

## 【0112】

ここで、制御装置30が備える制御部36の平滑化処理部39は、図5に示したステップS140でこの不連続に変化する経路ポテンシャルにローパスフィルターを適用することで、図15に示すような滑らかな経路ポテンシャルを生成する。図15は、平滑化処理部39が生成した滑らかな経路ポテンシャルの一例を表すグラフを示す図である。平滑化処理部39が図15に示すような滑らかな経路ポテンシャルを生成するため、ロボット制御部43は、滑らかな経路ポテンシャルに基づいて、ロボット20に所定の作業を行わせることができる。その結果、マニピュレータMNPのアクチュエーターに掛かる負荷を抑制することができる。

## 【0113】

ここで、図15に示した経路ポテンシャルを参照し、図5に示したステップS150の経路生成処理について詳しく説明する。図15に示した経路ポテンシャルの値が他の位置に比べて相対的に低い位置は、操作対象Nが組付対象Oや他の物体と干渉する可能性が低い位置であり、且つ組付対象Oから遠いために速く移動させることができる位置である。また、経路ポテンシャルの値が他の位置に比べて相対的に高い位置は、操作対象Nが組付対象Oや他の物体と干渉する可能性が高い位置であり、且つ組付対象Oに近いために遅く移動させなければならない位置である。

## 【0114】

そのため、ロボット20が操作対象Nを組付対象Oに組み付けるために移動させる際、ロボット20は、経路ポテンシャルの値が低い位置を経由して移動させることによって、操作対象Nと他の物体との干渉の抑制及び操作対象Nを移動させる時間の短縮を図ることができる。

## 【0115】

これを実現するための方法として、ステップS150における経路生成部40は、経路ポテンシャルの値を参照し、操作対象Nの組付位置に隣接する位置のうち、経路ポテンシ

10

20

30

40

50

ャルの値が最も小さな位置を経由点として特定する。経由点は、ロボット20が操作対象Nを初期の位置から組付位置まで移動させる際に操作対象Nが経由する点のことである。以下では、説明の便宜上、操作対象Nの初期の位置を、初期位置と称して説明する。

【0116】

ここで、ある位置Xに隣接する位置とは、作業領域内の位置を示す座標が隣接する位置のことである。この一例において、作業領域内の位置は、図2に示した座標軸によって表される二次元平面上の位置であり、当該座標軸の座標値によって示される。この座標値を変化させる場合、その変化量は、数学的には無限に小さな値にすることができる。しかし、実際の制御においては、当該変化量は、無限に小さな値にすることができず、ある有限の最小値にしなければならない。そのため、この一例における作業領域での位置は、図2に示した座標軸の座標値によって示される複数の格子点によって表される。すなわち、ある位置Xに隣接する位置とは、位置Xを表す格子点に隣接する他の格子点が表す位置のことである。以下では、説明の便宜上、ある位置Xに隣接する位置を、位置Xの隣接位置と称して説明する。

10

【0117】

経路生成部40は、組付位置に隣接する経由点を特定した後、当該経由点に隣接する位置のうち、経路ポテンシャルの値が最も小さな座標を次の経由点として特定する。これを繰り返し行うことにより、経路生成部40は、組付位置から初期位置までの間を複数の経由点を介して繋ぐ。そして、経路生成部40は、組付位置と、初期位置と、これらの複数の経由点とのうち、隣接する位置同士を線によって結ぶことにより、操作対象Nを組付対象Oに組み付けるまでの経路を生成する。

20

【0118】

なお、経路生成部40は、経由点を特定する際、経路ポテンシャルの値が同じ位置が2つ以上ある場合、例えば、2つ以上ある当該位置のうち、操作対象Nの初期位置との間の距離が最も近い位置を次の経由点として特定する。また、経路生成部40は、この一例において、一度経由した経由点を再び経由点として特定することを行わない場合について説明したが、これに代えて、一度経由した経由点を再び経由点として特定する構成であってもよい。この場合、経路生成部40は、例えば、前回に特定した経由点を次の経由点として選択しない等、ある2つの経由点の間で経由点の特定が繰り返し行われてしまうことを抑制する必要がある。

30

【0119】

ここで、図5に示したステップS160の速さ決定処理について詳しく説明する。速さ決定部41は、ステップS150において経路生成部40が特定した経由点のそれぞれにおいてロボット20が操作対象Nを移動させる速さを決定する。この決定の際、速さ決定部41は、例えば、図15に示した経路ポテンシャルの値を参照し、複数の経由点のそれぞれにおける経路ポテンシャルの値の逆数に所定の速さを乗じた値を、複数の経由点のそれぞれにおいてロボット20が操作対象Nを移動させる速さとして算出する。

【0120】

なお、速さ決定部41は、複数の経由点のそれぞれにおける経路ポテンシャルの値の逆数に所定の速さを加算した値を、複数の経由点のそれぞれにおいてロボット20が操作対象Nを移動させる速さとして算出する方法や、複数の経由点のそれぞれにおける経路ポテンシャルの値の逆数に所定の重みを乗じた値に所定の速さを乗じた値を、複数の経由点のそれぞれにおいてロボット20が操作対象Nを移動させる速さとして算出する方法等、経路ポテンシャルの値の逆数に基づく他の算出方法によって、複数の経由点のそれぞれにおいてロボット20が操作対象Nを移動させる速さを算出する構成であってもよい。また、速さ決定部41は、複数の経由点のそれぞれにおいてロボット20が操作対象Nを移動させる速さに上限を設けてもよい。例えば、速さ決定部41は、複数の経由点のそれぞれにおける経路ポテンシャルの値の逆数の値のうち、所定の上限値を超えている値を所定の上限値に変更することにより、複数の経由点のそれぞれにおいてロボット20が操作対象Nを移動させる速さに上限を設ける構成であってもよい。

40

50

## 【 0 1 2 1 】

ここで、速さ決定部 4 1 が決定した経路点毎の速さは、隣接する経路点同士の間で不連続に変化している。すなわち、速さ決定部 4 1 が決定した不連続に変化している速さに基づいてロボット 2 0 が操作対象 N を移動させた場合、ロボット 2 0 が備えるアクチュエーター等に余計な負荷が掛かってしまうことがある。これを抑制するため、速さ決定部 4 1 は、経路点毎に決定した速さが、隣接する経路点同士の間で連続的に（滑らかに）変化するように、経路点毎に決定した速さの変化を表す関数を、例えば、5 次多項式を用いてフィッティングすることにより平滑化（平均化）する。なお、速さ決定部 4 1 は、5 次多項式に代えて、他の多項式や特殊関数、三角関数等を用いてフィッティングする構成であってもよい。

10

## 【 0 1 2 2 】

以上説明したように、第 1 実施形態におけるロボットシステム 1（あるいは、ロボット 2 0）は、アームとともに動く操作対象 N と組付対象 O との相対的な位置及び姿勢、及び操作対象 N の初期位置に基づいて経路を生成する。これにより、ロボットシステム 1 は、操作対象 N と組付対象 O との相対的な位置及び姿勢、及び操作対象 N の組付位置に基づく経路に沿って操作対象 N を移動させることができる。その結果、ロボットシステム 1 は、例えば、他の物体との干渉の抑制及び操作対象 N を移動させる時間の短縮を図ることができる。

## 【 0 1 2 3 】

また、ロボットシステム 1 は、操作対象 N と組付対象 O との相対的な位置及び姿勢に基づく操作対象 N の移動可能な範囲を用いて経路を生成する。これにより、ロボットシステム 1 は、操作対象 N の移動可能な範囲を用いて生成した経路に沿って操作対象 N を移動させることができる。

20

## 【 0 1 2 4 】

また、ロボットシステム 1 は、操作対象 N の移動可能な範囲に基づいて、操作対象 N と組付対象 O との相対的な位置及び姿勢に応じた操作対象 N の動かし難さ又は動かし易さの指標となる量である可動ポテンシャルを算出し、算出された可動ポテンシャルを用いて経路を生成する。これにより、ロボットシステム 1 は、可動ポテンシャルを用いて生成した経路に沿って操作対象 N を移動させることができる。

## 【 0 1 2 5 】

また、ロボットシステム 1 は、操作対象 N の組付位置と操作対象 N の作業領域内における他の位置とを用いて経路を生成する。これにより、ロボットシステム 1 は、操作対象 N の組付位置と操作対象 N の作業領域内における他の位置とを用いて生成した経路に沿って操作対象 N を移動させることができる。

30

## 【 0 1 2 6 】

また、ロボットシステム 1 は、操作対象 N と組付対象 O との相対的な位置及び姿勢と操作対象 N の組付位置に基づいてアームを移動させる速さを決定する。これにより、ロボットシステム 1 は、操作対象 N と組付対象 O との相対位置姿勢と操作対象 N の組付位置に基づいて決定した速さによって操作対象 N を移動させることができる。

## 【 0 1 2 7 】

また、ロボットシステム 1 は、操作対象 N と組付対象 O との相対的な位置及び姿勢に基づく操作対象 N の移動可能な範囲を用いてアームを移動させる速さを決定する。これにより、ロボットシステム 1 は、操作対象 N の移動可能な範囲を用いて決定した速さによって操作対象 N を移動させることができる。

40

## 【 0 1 2 8 】

また、ロボットシステム 1 は、操作対象 N と組付対象 O との相対的な位置及び姿勢に応じた操作対象 N の動かし難さ又は動かし易さの指標となる量である可動ポテンシャルを算出し、算出された可動ポテンシャルを用いてアームを移動させる速さを決定する。これにより、ロボットシステム 1 は、可動ポテンシャルを用いて決定した速さによって操作対象 N を移動させることができる。

50

## 【0129】

また、ロボットシステム1は、操作対象Nの組付位置と操作対象Nの作業領域内における他の位置とを用いてアームを移動させる速さを決定する。これにより、ロボットシステム1は、操作対象Nの組付位置と操作対象Nの作業領域内における他の位置とを用いて決定した速さによって操作対象Nを移動させることができる。

## 【0130】

## &lt;第2実施形態&gt;

以下、本発明の第2実施形態について、図面を参照して説明する。なお、第2実施形態では、第1実施形態と同様な構成部には、同じ符号を付してある。

第2実施形態では、制御部36は、可動領域算出部37とポテンシャル算出部38に代えて、可動領域算出部37aとポテンシャル算出部38aを備える。

10

## 【0131】

可動領域算出部37aは、可動領域算出部37によるステップS100の可動領域算出処理に代えて、以下で説明する可動領域算出処理を行う。より具体的には、可動領域算出部37aは、ステップS100において、図16に示す並進可動領域算出処理と、図17に示す回転可動領域算出処理とを行う。

## 【0132】

ポテンシャル算出部38aは、ポテンシャル算出部38によるステップS120の距離ポテンシャル算出処理に代えて、以下で説明する距離ポテンシャル算出処理を行う。より具体的には、ポテンシャル算出部38aは、ステップS120において、図18に示す距離ポテンシャル算出処理を行う。

20

## 【0133】

以下、図16を参照して、可動領域算出部37aによる並進可動領域算出処理について説明する。図16は、可動領域算出部37aによる並進可動領域算出処理の流れの一例を示すフローチャートである。図16に示した並進可動領域算出処理は、図6に示した並進可動領域を位置の関数として算出する並進可動領域算出処理と異なり、並進可動領域を位置と姿勢の関数として算出する。

## 【0134】

まず、可動領域算出部37aは、作業領域を示す情報を記憶部32から読み込み、読み込まれた作業領域を示す情報に基づいて作業領域を表す仮想空間を生成する。そして、可動領域算出部37aは、生成された仮想空間の各点を示す座標を1つずつ選択し、選択された座標毎に、ステップS610からステップS650までの処理を繰り返し行う(ステップS600)。

30

## 【0135】

次に、可動領域算出部37aは、ステップS600で選択された座標が示す仮想空間の点に、仮想的に操作対象Nを配置する。この時、可動領域算出部37aは、仮想空間の点に操作対象Nの重心の位置が一致するように操作対象Nが配置する。また、可動領域算出部37aは、仮想空間の点に配置された操作対象Nの姿勢を、所定の初期姿勢となるまで操作対象Nを回転させる。そして、可動領域算出部37aは、操作対象Nの姿勢を所定角度ずつ変え、それぞれの姿勢毎にステップS620からステップS650までの処理を繰り返し行う(ステップS610)。

40

## 【0136】

次に、可動領域算出部37aは、数値を示す情報を格納可能な変数Dを生成し、変数Dの値をゼロに初期化する(ステップS620)。次に、可動領域算出部37aは、ステップS600で選択された座標が示す仮想空間の点に、ステップS610で選択された姿勢で配置された操作対象Nが4つの方向(すなわち、x軸の正方向、x軸の負方向、y軸の正方向、y軸の負方向)のそれぞれへの移動可能な距離の和を算出する(ステップS630)。次に、可動領域算出部37aは、ステップS630で算出された4つの方向への移動可能な距離の和を、ステップS620で生成された変数Dに格納されている値に加算する(ステップS640)。

50

## 【0137】

以下、図17を参照して、可動領域算出部37aによる回転可動領域算出処理について説明する。図17は、可動領域算出部37aによる回転可動領域算出処理の流れの一例を示すフローチャートである。図17に示した回転可動領域算出処理は、図7に示した回転可動領域を位置の関数として算出する回転可動領域算出処理と異なり、回転可動領域を位置と姿勢の関数として算出する。

## 【0138】

まず、可動領域算出部37aは、作業領域を示す情報を記憶部32から読み込み、読み込まれた作業領域を示す情報に基づいて作業領域を表す仮想空間を生成する。そして、可動領域算出部37aは、生成された仮想空間の各点を示す座標を1つずつ選択し、選択された座標毎に、ステップS710からステップS850までの処理を繰り返し行う（ステップS700）。なお、可動領域算出部37aにより回転可動領域算出処理が図13に示した並進可動領域算出処理の後に（又は並列に）行われる場合、すでに仮想空間が生成されているため、ステップS700において新たに仮想空間を生成しなくてもよい。

## 【0139】

次に、可動領域算出部37aは、ステップS700で選択された座標が示す仮想空間の点に、仮想的に操作対象Nを配置する。この時、可動領域算出部37aは、仮想空間の点に操作対象Nの重心の位置が一致するように操作対象Nが配置する。また、可動領域算出部37aは、仮想空間の点に配置された操作対象Nの姿勢を、所定の初期姿勢となるまで操作対象Nを回転させる。そして、可動領域算出部37aは、操作対象Nの姿勢を所定角度ずつ変え、それぞれの姿勢毎にステップS720からステップS850までの処理を繰り返し行う（ステップS710）。

## 【0140】

次に、可動領域算出部37aは、数値を示す情報を格納可能な変数D+を生成し、変数D+の値をゼロに初期化する（ステップS720）。次に、可動領域算出部37aは、ステップS700で選択された座標が示す仮想空間の点に配置された操作対象Nであって、現在の姿勢の操作対象Nが、図8に示した仮想空間VSの枠や組付対象Oの輪郭に干渉（すなわち、接触）しているか否かを判定する（ステップS730）。

## 【0141】

操作対象Nが仮想空間VSの枠や組付対象Oの輪郭に接触していないと判定した場合（ステップS730 - No）、可動領域算出部37aは、ステップS750における操作対象Nの回転に係る回転角の累積値が360°に達したか否かを判定する（ステップS740）。操作対象Nの回転に係る回転角の累積値が360°に達していないと判定した場合（ステップS740 - No）、可動領域算出部37aは、ステップS700で選択された座標が示す仮想空間の点に配置された操作対象Nであって、現在の姿勢の操作対象Nを、例えば、操作対象Nの重心を中心として正方向に所定角度だけ回転させ（ステップS750）、再びステップS730の判定を行う。ここで、所定角度は、その倍数に360°を含む任意の角度であることが望ましく、この一例において、5°や10°等であるとするが、その倍数に360°を含まない他の角度であってもよい。一方、操作対象Nの回転に係る回転角の累積値が360°に達していると判定した場合（ステップS740 - Yes）、可動領域算出部37aは、ステップS750における操作対象Nの回転に係る回転角の累積値を変数D+に代入する（ステップS760）。

## 【0142】

一方、ステップS730で操作対象Nが仮想空間VSの枠や組付対象Oの輪郭に干渉（すなわち、接触）していると判定した場合（ステップS730 - Yes）、可動領域算出部37aは、ステップS760に遷移し、ステップS750における操作対象Nの回転に係る回転角の累積値を変数D+に代入する。次に、可動領域算出部37aは、ステップS700で選択された座標が示す仮想空間の点に配置された操作対象Nの姿勢を、ステップS710で選択された姿勢に戻す（ステップS770）。

## 【0143】

次に、可動領域算出部 37a は、数値を示す情報を格納可能な変数 D - を生成し、変数 D - の値をゼロに初期化する（ステップ S 780）。次に、可動領域算出部 37a は、ステップ S 700 で選択された座標が示す仮想空間の点に配置された操作対象 N であって、現在の姿勢の操作対象 N が、図 8 に示した仮想空間 V S の枠や組付対象 O の輪郭に干渉（すなわち、接触）しているか否かを判定する（ステップ S 790）。

#### 【0144】

操作対象 N が仮想空間 V S の枠や組付対象 O の輪郭に接触していないと判定した場合（ステップ S 790 - No）、可動領域算出部 37a は、ステップ S 810 における操作対象 N の回転に係る回転角の絶対値の累積値が 360° に達したか否かを判定する（ステップ S 800）。操作対象 N の回転に係る回転角の累積値が 360° に達していないと判定した場合（ステップ S 800 - No）、可動領域算出部 37a は、ステップ S 700 で選択された座標が示す仮想空間の点に配置された操作対象 N であって、現在の姿勢の操作対象 N を、例えば、操作対象 N の重心を中心として負方向に所定角度だけ回転させ（ステップ S 810）、再びステップ S 790 の判定を行う。ここで、所定角度は、その倍数に 360° を含む任意の角度であることが望ましく、この一例において、5° や 10° 等であるとするが、その倍数に 360° を含まない他の角度であってもよい。一方、操作対象 N の回転に係る回転角の累積値が 360° に達していると判定した場合（ステップ S 800 - Yes）、可動領域算出部 37a は、ステップ S 810 における操作対象 N の回転に係る回転角の累積値を変数 D - に代入する（ステップ S 820）。

#### 【0145】

一方、ステップ S 790 で操作対象 N が仮想空間 V S の枠や組付対象 O の輪郭に干渉（すなわち、接触）していると判定した場合（ステップ S 790 - Yes）、可動領域算出部 37a は、ステップ S 820 に遷移し、ステップ S 810 における操作対象 N の回転に係る回転角の累積値を変数 D - に代入する。次に、可動領域算出部 37a は、変数 D + に代入されている値が変数 D - に代入されている値未満であるか否かを判定する（ステップ S 830）。変数 D + に代入されている値が変数 D - に代入されている値未満であると判定した場合（ステップ S 830 - Yes）、可動領域算出部 37a は、変数 D + の値を回転可動領域に代入する（ステップ S 840）。一方、変数 D + に代入されている値が変数 D - に代入されている値以上であると判定した場合（ステップ S 830 - No）、可動領域算出部 37a は、変数 D - の値を回転可動領域に代入する（ステップ S 850）。

#### 【0146】

このように、可動領域算出部 37a は、ステップ S 600 からステップ S 650 までの処理と、ステップ S 700 からステップ S 850 までの処理とをそれぞれ繰り返し行うことで、作業領域内の各点における操作対象 N の並進可動領域及び回転可動領域を、操作対象 N の姿勢毎に算出する。

なお、ステップ S 730 において、可動領域算出部 37a は、変数 D - に代入されている値が変数 D + に代入されている値未満であるか否かを判定する構成であってもよい。また、可動領域算出部 37a は、ステップ S 730 からステップ S 760 までの処理と、ステップ S 780 からステップ S 820 までの処理とを逆の順で行う構成であってもよく、同時に行う構成であってもよい。

#### 【0147】

以下、図 18 を参照して、ポテンシャル算出部 38a による距離ポテンシャル算出処理について説明する。図 18 は、ポテンシャル算出部 38a による距離ポテンシャル算出処理の流れの一例を示すフローチャートである。図 18 に示した距離ポテンシャル算出処理は、図 11 に示した距離ポテンシャルを位置の関数（すなわち、並進距離ポテンシャル）として算出する距離ポテンシャル算出処理と異なり、距離ポテンシャルを位置と姿勢の関数として算出する。

#### 【0148】

まず、ポテンシャル算出部 38a は、作業領域を示す情報を記憶部 32 から読み込み、読み込まれた作業領域を示す情報に基づいて作業領域を表す仮想空間を生成する。そして

、ポテンシャル算出部 38 a は、生成された仮想空間の各点を示す座標を 1 つずつ選択し、選択された座標毎に、ステップ S 910 からステップ S 930 までの処理を繰り返し行う（ステップ S 900）。なお、可動領域算出部 37 a により回転可動領域算出処理が図 16 に示した並進可動領域算出処理の後に（又は並列に）行われる場合、すでに仮想空間が生成されているため、ステップ S 900 において新たに仮想空間を生成しなくてもよい。

#### 【0149】

次に、ポテンシャル算出部 38 a は、ステップ S 900 で選択された座標が示す仮想空間の点に、仮想的に操作対象 N を配置する。この時、ポテンシャル算出部 38 a は、仮想空間の点に操作対象 N の重心の位置が一致するように操作対象 N を配置する。また、ポテンシャル算出部 38 a は、仮想空間の点に配置された操作対象 N の姿勢を、所定の初期姿勢となるまで操作対象 N を回転させる。そして、ポテンシャル算出部 38 a は、操作対象 N の姿勢を所定角度ずつ変え、それぞれの姿勢毎にステップ S 920 からステップ S 930 までの処理を繰り返し行う（ステップ S 910）。

#### 【0150】

ポテンシャル算出部 38 a は、作業領域内における操作対象 N の組付位置と、ステップ S 900 において選択した座標（ $x, y$ ）が示す位置との間の距離を並進距離  $S_t(x, y)$  として算出する（ステップ S 920）。次に、ポテンシャル算出部 38 a は、例えば、作業領域内における操作対象 N の組付位置に対応する操作対象 N の組付姿勢を表す角度ベクトル（姿勢ベクトル） $s$  と、ステップ S 920 において選択した操作対象 N の姿勢を表す角度ベクトルとの間の距離を、回転距離  $S_r(\quad)$  として以下の式を用いて算出する。

#### 【0151】

$$S_r(\quad) = (\quad_s - \quad)^2$$

#### 【0152】

ステップ S 900 からステップ S 930 までの繰り返し処理により、ポテンシャル算出部 38 a は、仮想領域内のすべての位置及び姿勢毎に、並進距離  $S_t(x, y)$  と回転距離  $S_r(\quad)$  を算出することができる。

#### 【0153】

次に、ポテンシャル算出部 38 a は、ステップ S 900 からステップ S 930 までの繰り返し処理により算出した並進距離  $S_t(x, y)$  及び回転距離  $S_r(\quad)$  を合算した合計距離  $S_{total}(x, y, \quad)$  を算出する。そして、ポテンシャル算出部 38 a は、算出した合計距離  $S_{total}(x, y, \quad)$  の全ての点の中での最大値  $S_{MAX}$  を検出する（ステップ S 940）。

#### 【0154】

次に、ポテンシャル算出部 38 a は、ステップ S 940 において検出した合計距離  $S_{total}(x, y, \quad)$  の最大値  $S_{MAX}$  に基づいて、作業領域内の各点における距離ポテンシャルを算出する（ステップ S 950）。より具体的には、ポテンシャル算出部 38 a は、以下に示す式（5）に基づいて作業領域内の各点における距離ポテンシャルを算出する。

#### 【0155】

$$PT(x, y, \quad) = S_{MAX} - S_{total}(x, y, \quad) \cdots (5)$$

#### 【0156】

ここで、 $PT(x, y, \quad)$  は、作業領域内の位置及び姿勢（ $x, y, \quad$ ）における距離ポテンシャルであり、並進距離及び回転距離に基づいて算出される距離ポテンシャルを表す。上記の式（5）は、合計距離の最大値  $S_{MAX}$  から、作業領域内の位置及び姿勢（ $x, y, \quad$ ）における合計距離  $S_{total}(x, y, \quad)$  を減算した値を、当該位置及び姿勢（ $x, y, \quad$ ）における距離ポテンシャルとして定義する式である。このように距離ポテンシャルを定義することで、作業領域内の点のうち距離ポテンシャルが大きな点は、操作対象 N が組付位置に近い、すなわち操作対象 N が組付対象 O に近いことを表すこと

10

20

30

40

50



ができる。

【 0 1 5 7 】

以上説明したように、第 2 実施形態におけるロボットシステム 1 は、操作対象 N の組付位置と、当該組付位置に対応する操作対象 N の組付姿勢と、操作対象 N の作業領域内における他の位置及び姿勢とを用いて経路を生成する。これにより、ロボットシステム 1 は、操作対象 N の組付位置と、当該組付位置に対応する操作対象 N の組付姿勢と、操作対象 N の作業領域内における他の位置及び姿勢とを用いて生成した経路に沿って操作対象 N を移動させることができる。

【 0 1 5 8 】

また、ロボットシステム 1 は、操作対象 N の組付位置と、当該組付位置に対応する操作対象 N の組付姿勢と、操作対象 N の作業領域内における他の位置及び姿勢とを用いてアームを移動させる速さを決定する。これにより、ロボットは、操作対象 N の組付位置と、当該組付位置に対応する操作対象 N の組付姿勢と、操作対象 N の作業領域内における他の位置及び姿勢とを用いて決定した速さによって操作対象 N を移動させることができる。

【 0 1 5 9 】

なお、組付位置は、第 1 位置、及び第 3 位置それぞれの一例である。また、組付姿勢は、第 1 位置に対応する第 1 姿勢、及び第 3 位置に対応する第 3 姿勢それぞれの一例である。

【 0 1 6 0 】

以上、この発明の実施形態を、図面を参照して詳述してきたが、具体的な構成はこの実施形態に限られるものではなく、この発明の要旨を逸脱しない限り、変更、置換、削除等されてもよい。

【 0 1 6 1 】

また、以上に説明した装置（例えば、ロボットシステム 1 の制御装置 3 0 ）における任意の構成部の機能を実現するためのプログラムを、コンピューター読み取り可能な記録媒体に記録し、そのプログラムをコンピューターシステムに読み込ませて実行するようにしてもよい。なお、ここでの「コンピューターシステム」とは、OS（Operating System）や周辺機器等のハードウェアを含むものとする。また、「コンピューター読み取り可能な記録媒体」とは、フレキシブルディスク、光磁気ディスク、ROM（Read Only Memory）、CD（Compact Disk）- ROM 等の可搬媒体、コンピューターシステムに内蔵されるハードディスク等の記憶装置のことをいう。さらに「コンピューター読み取り可能な記録媒体」とは、インターネット等のネットワークや電話回線等の通信回線を介してプログラムが送信された場合のサーバーやクライアントとなるコンピューターシステム内部の揮発性メモリー（RAM：Random Access Memory）のように、一定時間プログラムを保持しているものも含むものとする。

【 0 1 6 2 】

また、上記のプログラムは、このプログラムを記憶装置等に格納したコンピューターシステムから、伝送媒体を介して、あるいは、伝送媒体中の伝送波により他のコンピューターシステムに伝送されてもよい。ここで、プログラムを伝送する「伝送媒体」は、インターネット等のネットワーク（通信網）や電話回線等の通信回線（通信線）のように情報を伝送する機能を有する媒体のことをいう。

また、上記のプログラムは、前述した機能の一部を実現するためのものであってもよい。さらに、上記のプログラムは、前述した機能をコンピューターシステムにすでに記録されているプログラムとの組み合わせで実現できるもの、いわゆる差分ファイル（差分プログラム）であってもよい。

【 符号の説明 】

【 0 1 6 3 】

1 ロボットシステム、2 0 ロボット、3 0 制御装置、3 1 CPU、3 2 記憶部、3 3 入力受付部、3 4 通信部、3 5 表示部、3 6 制御部、3 7 可動領域算出

10

20

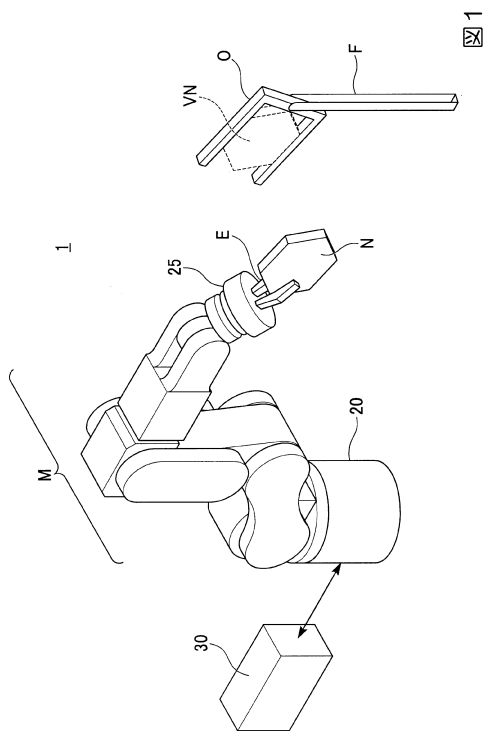
30

40

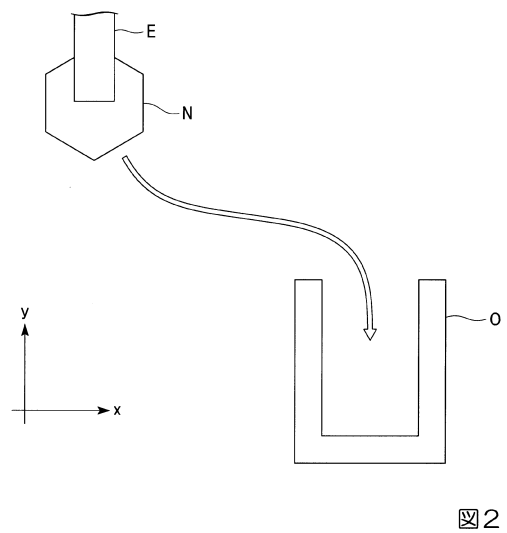
50

部、38 ポテンシャル算出部、39 平滑化处理部、40 経路生成部、41 速さ決定部、42 カセンサー取得部、43 ロボット制御部

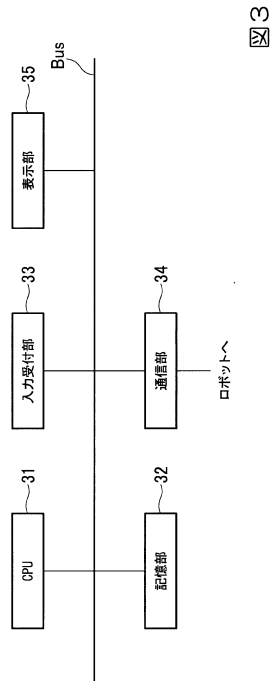
【図1】



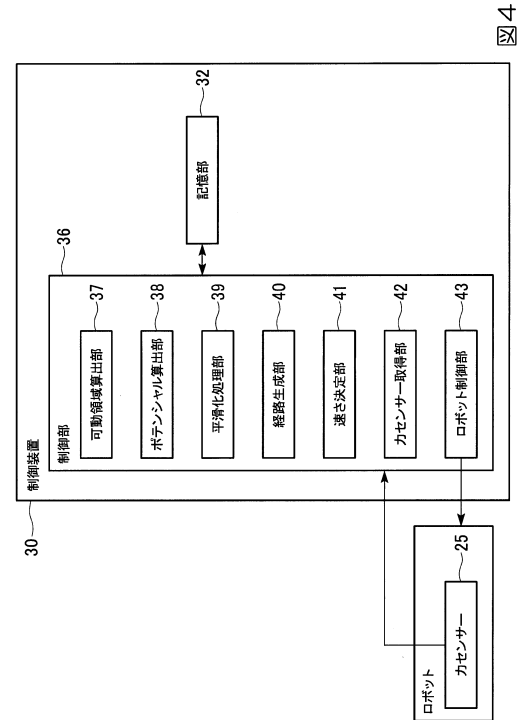
【図2】



【図3】



【図4】



【図5】

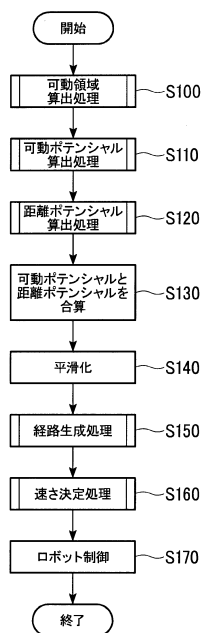


図5

【図6】

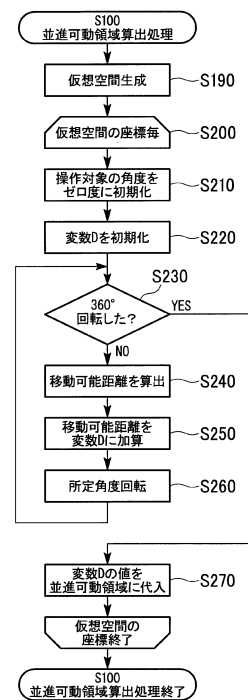


図6

【図 7】

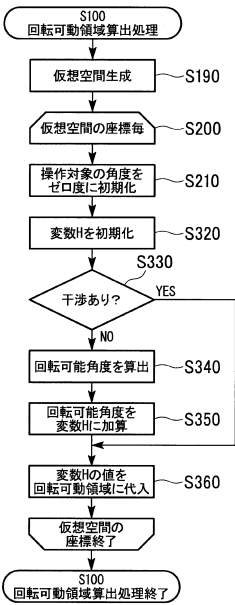
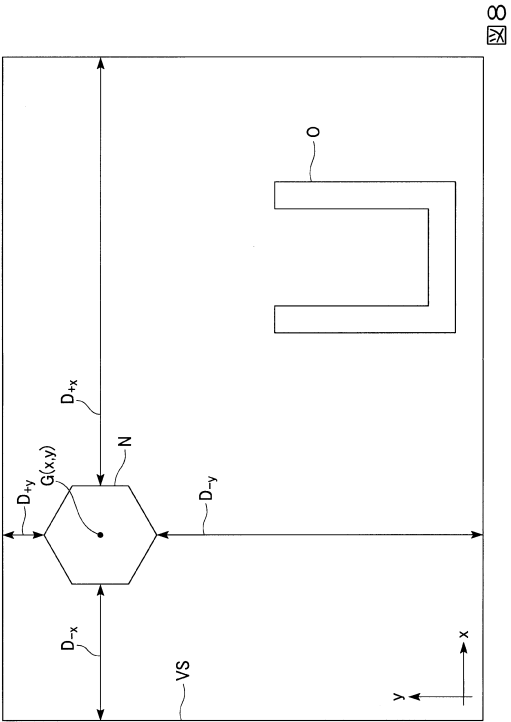


図 7

【図 8】



【図 9】

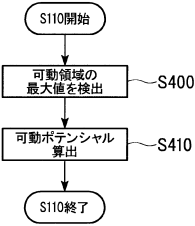


図 9

【図 10】

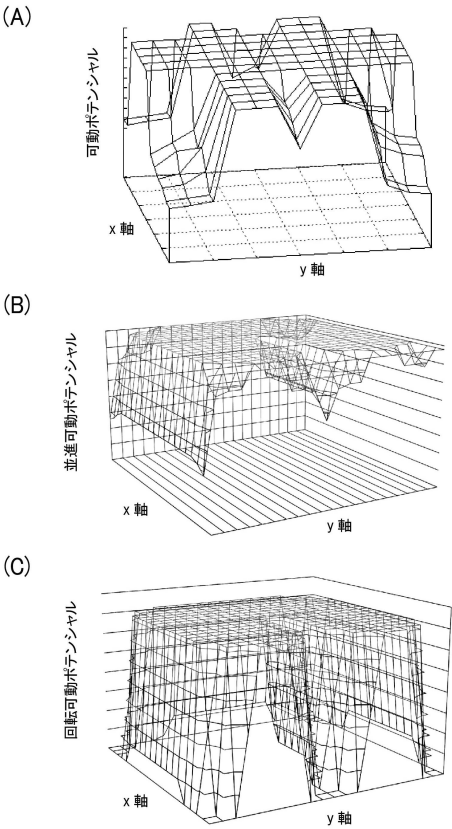


図 10

【図 1 1】

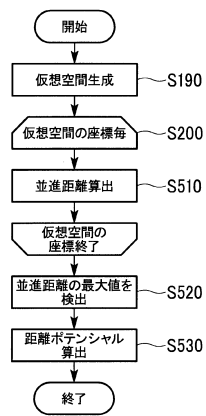


図 1 1

【図 1 2】

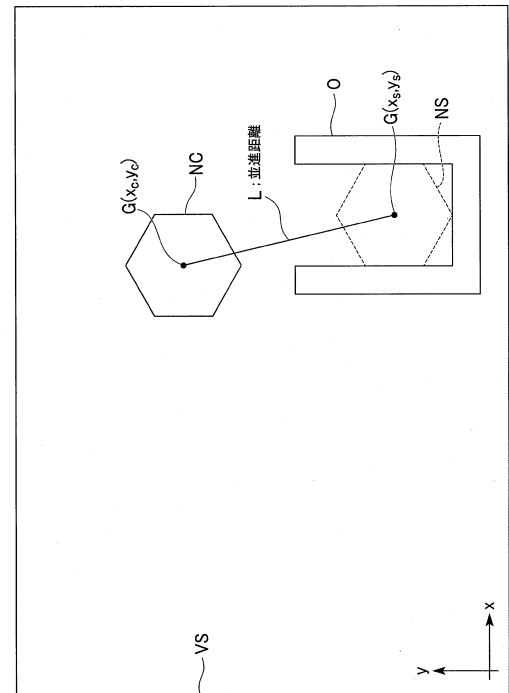


図 1 2

【図 1 3】

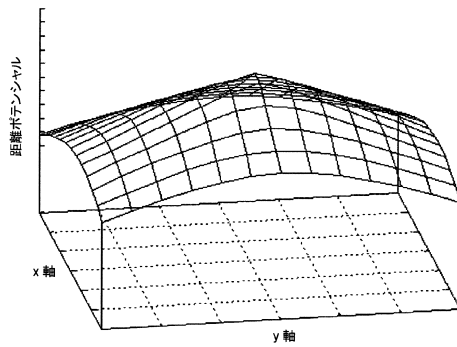


図 1 3

【図 1 4】

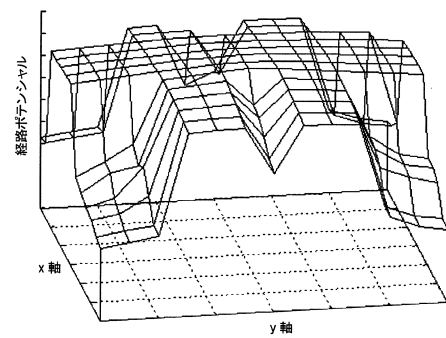
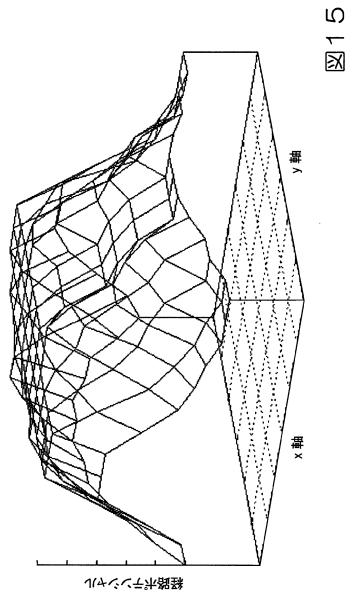


図 1 4

【図15】



【図16】

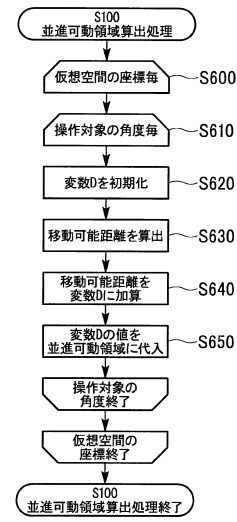


図16

【図17】

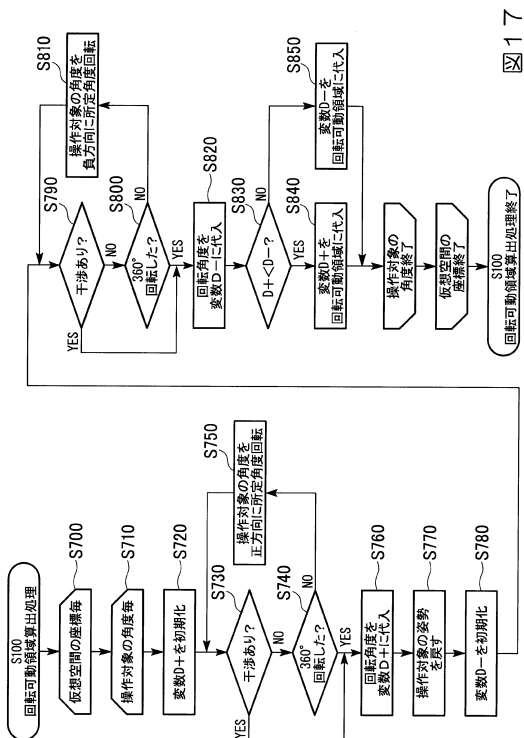


図17

【図18】

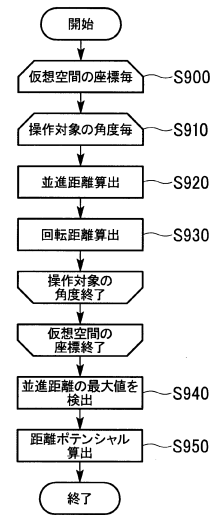


図18

---

フロントページの続き

- (72)発明者 小菅 一弘  
宮城県仙台市青葉区片平二丁目1番1号 国立大学法人東北大学内
- (72)発明者 サクセナ アピラシャ  
宮城県仙台市青葉区片平二丁目1番1号 国立大学法人東北大学内

審査官 藤井 浩介

- (56)参考文献 特開2010-188485(JP,A)  
特開2014-115924(JP,A)  
特開2000-020117(JP,A)  
特開2012-011495(JP,A)  
特開2013-242777(JP,A)  
特開2011-108129(JP,A)  
国際公開第2012/124342(WO,A1)  
特開2003-280710(JP,A)  
鈴木 崇文,他2名,“自律全方位牽引式搬送ロボットの開発”,日本ロボット学会誌,日本,  
社団法人日本ロボット学会,2009年10月22日,第27巻,第8号,p.118-125
- (58)調査した分野(Int.Cl.,DB名)  
B25J 1/00-21/02  
G05D 1/00-1/12