

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5114019号
(P5114019)

(45) 発行日 平成25年1月9日(2013.1.9)

(24) 登録日 平成24年10月19日(2012.10.19)

(51) Int.Cl.		F I			
B 2 5 J	9/10	(2006.01)	B 2 5 J	9/10	A
B 2 5 J	5/00	(2006.01)	B 2 5 J	5/00	F
G 0 5 D	3/00	(2006.01)	G 0 5 D	3/00	Z

請求項の数 6 (全 11 頁)

(21) 出願番号	特願2006-142703 (P2006-142703)	(73) 特許権者	503113186
(22) 出願日	平成18年5月23日 (2006.5.23)		ホンダ リサーチ インスティテュート
(65) 公開番号	特開2006-334774 (P2006-334774A)		ヨーロッパ ゲーエムベーハー
(43) 公開日	平成18年12月14日 (2006.12.14)		Honda Research Institute Europe GmbH
審査請求日	平成19年10月25日 (2007.10.25)		ドイツ連邦共和国 デー-63073 オ
(31) 優先権主張番号	05011727.4		ッフエンバッハアムメイン カール・レギ
(32) 優先日	平成17年5月31日 (2005.5.31)		エン・シュトラーセ 30
(33) 優先権主張国	欧州特許庁 (EP)	(74) 代理人	110000246
			特許業務法人OFH特許事務所
		(72) 発明者	ミヒャエル・ギンガー
			ドイツ国デー-60599フランクフルト
			、アム・メイン・ゴルトベルクヴェック
			8

最終頁に続く

(54) 【発明の名称】 エフェクタの軌道を制御するための方法

(57) 【特許請求の範囲】

【請求項 1】

タスク空間における、エフェクタを現在の状態 (a a c t) から対称軸を有する所与の対象物に関する目標の状態 (a t a r g e t) に至る軌道上で運動させるためのタスク記述を生成することを含み、前記エフェクタの運動を制御するための方法において、

前記タスク記述は、前記現在および目標の状態のベクトルの要素である制御パラメータを用い、前記対象物の対称性に基づいて決定される必要とされない不変制御パラメータを含めずに作成され、すなわち、前記対象物の対称軸周りの前記エフェクタの回転を含めずに作成され、前記不変制御パラメータを含まないタスク記述を使用することによって、タスク空間の次元を減らすことにより前記制御における零空間の自由度を増大させ、該増大された零空間の自由度を、関節限界を避けるため、特異姿勢を避けるため、または障害物を避けるために使用することを特徴とする、前記方法。

【請求項 2】

前記エフェクタは、前記目標の状態に向けて回転すべき軸を有する、請求項 1 に記載の方法。

【請求項 3】

前記エフェクタの軸の回転は、2次元タスク・ベクトル(x)によって記述される、請求項 2 に記載の方法。

【請求項 4】

タスク空間におけるエフェクタの現在の状態 (a a c t) から対称軸を有する所与の対

象物に関する目標の状態 (a t a r g e t) に至る軌道上で運動させるためのタスク記述を生成することを含み、前記エフェクタの運動を制御するための方法において、

前記運動の制御は、前記タスク記述において座標で表される前記現在の状態と前記目標の状態のベクトル要素の差に基づいて実施され、その際、前記エフェクタの軌道は、前記現在の状態と前記目標の状態との差を計算し、該差に基づいて前記エフェクタを動かすためのベクトル要素の増分値を前記タスク空間から配置空間上にマッピングすることによって計算され、

前記配置空間と前記タスク空間との間の次元の差は、前記タスク空間の運動に影響しない零空間の冗長な自由度を提供し、

前記タスク記述は、前記現在および目標の状態のベクトル要素を制御パラメータとして用い、前記対象物の対称性に基づいて決定される必要とされない不変制御パラメータを含めないで作成され、すなわち、前記対象物の対称軸周りの前記エフェクタの回転を含めないで作成され、前記不変制御パラメータを含まないタスク記述を使用することによって、タスク空間の次元を減らすことにより前記制御における零空間の自由度を増大させ、該増大された零空間の自由度を、関節限界を避けるため、特異姿勢を避けるため、または障害物を避けるために使用することを特徴とする、前記方法。

【請求項 5】

前記エフェクタは、前記目標の状態に向けて回転すべき軸を有する、請求項 4 に記載の方法。

【請求項 6】

前記エフェクタの軸の回転は、2次元タスク・ベクトルで記述される、請求項 5 に記載の方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、現在の状態から目標の状態へのエフェクタ軌道の制御およびコンピュータ・ソフトウェア・プログラムに関する。また、本発明は、マニピュレータ、駆動式 (actuated) カメラ・システム、1つまたは複数のマニピュレータを備えるロボット、または運転者支援システムを搭載した自動車に関する。

【背景技術】

【0002】

エフェクタ (たとえばロボットなど) の軌道を制御する場合、目標の状態を定義しなければならない。目標の状態は、たとえば、ロボットの操作アームが扱うべき対象物によって定義される。一般に、対象物の位置は、3つのパラメータによって記述することができる。対象物の位置に加えて、しばしばカルダン角 (Kardan angle) またはオイラー角 (Euler angle) によって作られる空間的な方向 (spatial orientation) を記述することが必要である。

【0003】

ロボットのエフェクタの動きを実行するために、通常では、制御パラメータ空間での増分量を配置空間 (configuration space) 上にマッピングすることによって、起動が生成される。

【0004】

制御パラメータ空間またはタスク空間は、各コマンド要素の空間である。制御パラメータ空間は、各コマンド要素からなる。コマンド (同様に「ターゲット」または「タスク」) の各要素は、それぞれのコマンド・ベクトルの各要素である。これらの要素は、制御すべき対象 (たとえば手の位置または頭部の傾き) についての有用な記述を定義する。配置空間は、制御可能な自由度の空間である。配置空間は、ロボットの個々の関節から構成され、または制御可能な自由度を割り当てることができるさらに複雑な運動学機構から構成される。

【0005】

10

20

30

40

50

マッピングは、以下の3つの異なるケースに分けることができる。

【0006】

第1のケースでは、配置空間次元（または関節空間次元）は、制御パラメータ空間次元（タスク空間次元）に一致する。このような場合、マッピングはたいてい一義的である。

【0007】

第2のケースでは、タスク空間次元は関節空間次元を超える（exceed）。この場合、タスク空間に基づいて動作を実行することができないため、一般的にマッピングに対する解はない。

【0008】

第3のケースは、関節空間の次元がタスク空間の次元よりも高い場合の状況を表す。この結果、関節空間とタスク空間との間の次元の差を表すいわゆる「零空間（Null space）」になる。零空間は冗長な自由度を含み、そこではタスク空間の運動に影響を与えずに動作を実行することができる。

10

【0009】

インターネット上でみつけることができる零空間の数学的な定義は次のとおりである（http://www-robotics.cs.umass.edu/Research/Glossary/null_space.htmlを参照）。

【0010】

零空間：対応する関数値がゼロであるような線形演算子の引数（argument）のセットである。冗長システムは、1次タスクを妨げずに、運動学調整などの2次的な目的に取り組むために使用することができる（局所的な）零空間を有する。

20

【0011】

したがって、零空間を残すタスクをもつロボットは、「冗長ロボット（redundant robots）」、「運動学的冗長マニピュレータ（kinematically redundant manipulators）」などとも呼ばれる。マニピュレータの分野（たとえばロボット）では、障害物を避けるために零空間を使用することが知られている。

【発明の開示】

【発明が解決しようとする課題】

【0012】

多くの問題は、対称性を有しており、したがって、対象物の位置および空間的方向について6個のパラメータの記述を必要としない。この点が考慮されていないことは、エフェクタ軌道の既知の生成方法に関する問題点である。

30

【0013】

本発明の目的は、エフェクタの操作を簡略化し、追加的な制御制約条件の適用を可能にすることである。

【課題を解決するための手段】

【0014】

前述の課題は、請求項1から請求項8に従うエフェクタの運動を制御するための各方法によって解決される。

【0015】

本発明を適用することにより、たとえばロボットの制御の頑強性を改善することができるなど、最終的にその消費電力をより効率的にすることができるなど、物理エンティティで表すことができる効果を、現実世界でもつことができる。

40

【0016】

本発明によれば、現在の状態から目標の状態へのエフェクタ軌道は、不変制御パラメータ（invariant control parameters）が考慮されるように制御される。この際、軌道の不変制御パラメータが決定される。不変制御パラメータは、必要とはされないが所与のタスクを達成する妨げにはならない制御パラメータである。これは、たとえば、後ほど詳細に説明する円筒形対象物の対称軸周りの手の回転である。そして、エフェクタ軌道は、タスク記述中で表され、このタスク記述には不変制御パラメータがない。タスク記述が不変制御パラメータを含まないとき、零空間の次元は増大する。零空間の次元が増大することの

50

利点は、冗長度も増大することである。これにより、冗長な制御技法または他の追加の制御基準を適用することが可能になる。エフェクタ軌道は、不変制御パラメータを含まないタスク記述に基づいて制御される。

【0017】

本発明の他の態様によれば、エフェクタ軌道は、増分値を制御パラメータ空間から配置空間上にマッピングすることによって計算される。配置空間と制御パラメータ空間との間の次元の差に起因する零空間の自由度が増大する。したがって、軌道の不変制御パラメータが決定される。零空間の自由度の増大は、不変制御パラメータを含まないタスク記述の使用に基づく。

【0018】

すでに説明されているように、零空間の自由度におけるこうした増大によって、結果として冗長な制御技法または追加の制御基準を適用することができるようになる。したがって、不変制御パラメータを決定し、次いでエフェクタ軌道をタスク記述で表すことが必要であり、タスク記述は不変制御パラメータを含まないことにより次元が減少する。

【0019】

好ましい実施形態は、下位クレームにおいて権利請求する。

【0020】

好ましい実施形態が図面に示され、詳細に説明される。その説明から、さらなる利点が明白になるであろう。

【発明を実施するための最良の形態】

【0021】

最初に、図1に示した人型ロボット1の運動学モデルを説明する。同様に、本発明は、他のロボット、自動化された操作デバイス (manipulation devices)、または自動車への適用を見いだせることに留意されたい。

【0022】

ロボット1の現在の状態は、配置空間での21自由度を表す座標からなる状態ベクトル q によって定義される。この状態ベクトルは、配置空間の要素を含む。状態ベクトル q を導き出す基準は、慣性系の世界固定座標系 (inertial, world-fixed coordinate frame) 2 である。慣性系 (inertial frame) 2 に対するロボット1の位置は、慣性系2に対するかかと座標系 (heel coordinate system) 3 の位置および回転を定義することによって記述される。かかと座標系3の位置および方向は、3つの座標を有するベクトルによって表される。これらの座標は、 x および y 方向での変換、ならびに慣性系2の垂直軸周りの回転を記述する。

【0023】

次に、上半身は、かかと座標系3の位置および方向に対して記述されなければならない。上半身座標系4の位置および方向は、かかと座標系3に対して定義される。この位置は3つの位置座標によって定義され、その空間的方向はかかと座標系3に対する回転を表す3つの角度によって記述される。ロボット・アームの姿勢は、5つの角度で記述される。これらの角度のうち3つは、肩5を表し、ひじは第4の角度で表され、最後に手首はアームの第5の角度によって表される。各角度は、それぞれの関節を介して接続されている各リンク間の関係を記述する。

【0024】

ロボット1の右腕は5つの関節を有し、頭部は頭部のパンおよびチルトのための他の2つの自由度を有することを考慮に入れると、状態ベクトル q の21自由度が説明される。

【0025】

ロボット1を動かすためには、各単独座標の値を変更することが必要である。これは、タスク・ベクトルまたはコマンド・ベクトルによって実行される。順運動学 (direct kinematics) を使用することで、タスク・ベクトルまたはコマンド・ベクトルは、状態ベクトル q に基づいて次のように計算される。

$$x = f(q)$$

10

20

30

40

50

【 0 0 2 6 】

ロボット 1 がツリー形構造であるために、ロボット 1 を記述する各単独系 3、5、8 の位置および方向は慣性系 2 から段階的に生じるので、コマンド・ベクトル x の計算を再帰的に行うことができる。

【 0 0 2 7 】

さらに、この計算は、2つの別々のステップに分割することができる。第 1 のステップは、系の回転を計算することである。基準に対する具体的な系、たとえば、かかと座標系 3 に対する上半身座標系 4 の方向を計算するために、3つのセグメント回転行列 A_x 、 A_y および A_z が使用される。これらの回転行列は、正の x 、 y および z 軸周りの相対的な回転を記述するために使用される。前に述べたように、かかと座標系 3 は、垂直軸周りだけで回転させられる。したがって、かかと座標系 3 についての回転行列は次式で定義される。

【数 1】

$$\mathbf{A}_{hl-I} = \mathbf{A}_z(\varphi_z, hl) \quad \mathbf{A}_I = \mathbf{A}_z(\varphi_z, hl)$$

ここで、 $\mathbf{A}_I = \mathbf{E}$ であり、 \mathbf{E} は慣性系 2 についての単位行列である。

【 0 0 2 8 】

回転行列 \mathbf{A}_{hl-I} の添字は、システム I (慣性系) からシステム hl (かかと座標系 3) への回転を表す。最初の座標系は、加速されていない基準座標系である。これは、ロボット工学の専門用語では、しばしば「世界座標系 (World coordinate frame)」とも呼ばれている。図 1 に示した運動学モデルによって、上半身座標系 4 を軸 x 、 y 、 z の 3 つすべてに対して回転させることができる。したがって、上半身についての回転行列は次式の通りである。

【数 2】

$$\mathbf{A}_{ub-I} = \mathbf{A}_z(\varphi_z, ub) \mathbf{A}_y(\varphi_y, ub) \mathbf{A}_x(\varphi_x, ub) \mathbf{A}_{hl-I}$$

【 0 0 2 9 】

同様に、3つの肩関節 5、ならびに頭部のパンおよびチルト関節によって構築される系の回転行列を計算することができる。

【 0 0 3 0 】

回転行列に加えて、座標系の原点を計算することが必要である。回転行列の計算と同様に、座標系の原点ベクトル (origin vector) は、慣性系 2 の座標中のかかと座標系 3 の原点ベクトル ${}_I \mathbf{r}_{hl}$ から計算することができる。慣性系 2 の中のかかと座標系 3 の原点ベクトル ${}_I \mathbf{r}_{hl}$ は次式の通りである。

【数 3】

$${}_I \mathbf{r}_{hl} = (x_{hl} \ y_{hl} \ 0)^T$$

【 0 0 3 1 】

ロボット 1 の慣性位置において、上半身座標系 4 の原点は z 軸周りに回転し、したがって、慣性系の座標における上半身座標系 4 の原点ベクトル ${}_I \mathbf{r}_{ub}$ は次式の通りである。

【数 4】

$${}_I \mathbf{r}_{ub} = {}_I \mathbf{r}_{hl} + \mathbf{A}_z^T(\varphi_z, hl) {}_{hl} \mathbf{r}_{ub}$$

【 0 0 3 2 】

他の系の原点ベクトルは、それぞれ計算される。ロボット1の現在の状態は、このように慣性系2の座標における状態ベクトル q によって記述される。

【0033】

慣性系2の座標における各系の回転行列および原点ベクトルは、順運動学の計算および逆運動学の方程式の設定のための基準を構築する。

【0034】

図2には、人型ロボット1用の典型的なタスクが示してある。問題は、たとえばロボット1のエンド・エフェクタを目標の状態にもってこることである。図2では、これを、対象物を握ることになる手9を用いて示している。多くの対象物は円筒形である。

【0035】

したがって、円筒形の対象物の対称軸周りの手の回転は、こうした円筒形の対象物を握ることとは関係がない。こうした対象物は、たとえば円筒形のグラスとすることができる。矢印10は手の握りの軸を表す。タスクは、グラスを握ることができる位置に手9をもってくることであり、それは、手9の握りの軸10が円筒形の対象物の対称軸と一致するまで手を回転させることと同等である。対象物は回転対称なので、握りの軸10の周りの手9の回転は、握りのプロセスに影響を及ぼさない。かかと座標系3に対するロボット1の左手の位置は、次式で表される。

【数5】

$${}_{hl}\mathbf{r}_{ht,i} = \mathbf{A}_{hl-I} \mathbf{I}\mathbf{r}_{ht,i}$$

10

20

ここで、添字「 i 」は、左手または右手を表し、添字「 ht 」は、手の基準点を表す。

【0036】

通常、手9の姿勢（すなわち手9の空間的方向）は、カルダン角またはオイラー角で記述される。手9の姿勢は、慣性系2に対するカルダン角（ x 軸周り）、（回転された y 軸周り）および（回転された z 軸周り）に関するそれぞれの手の回転行列によって導き出すことができる。対応する回転行列は次式の通りである。

【数6】

$$A_{KI} = \begin{pmatrix} \cos\beta \cdot \cos\gamma & \cos\alpha \cdot \sin\gamma + \sin\alpha \cdot \sin\beta \cdot \cos\gamma & \sin\alpha \cdot \sin\gamma - \cos\alpha \cdot \sin\beta \cdot \cos\gamma \\ -\cos\beta \cdot \sin\gamma & \cos\alpha \cdot \cos\gamma - \sin\alpha \cdot \sin\beta \cdot \sin\gamma & \sin\alpha \cdot \cos\gamma + \cos\alpha \cdot \sin\beta \cdot \sin\gamma \\ \sin\beta & -\sin\alpha \cdot \cos\beta & \cos\alpha \cdot \cos\beta \end{pmatrix}$$

30

【0037】

回転行列 A_{KI} から、カルダン角、およびを導き出すことができる。この3次元記述において手9の姿勢は「固定」され、それは手の握りの軸10の方向が固定されていることを意味する。これは、矢印10で表された対象物の対称軸周りの手9の回転を含む。必要ではないが、握りの軸10の周りの手9の回転角度が記述されている。したがって、手の姿勢を3自由度よりも少ない自由度で表すことが有利である。手の姿勢を3自由度よりも少ない自由度で記述することで、零空間の次元が拡張する。本発明によれば、手9の姿勢および結果として生じる軌道を記述するためのタスク記述が使用される。対称性側面を使用することによって、また不変制御パラメータを決定することによって、タスク空間は次元を減らす。軌道は、解決すべき問題にとって不可欠な要素のタスク・ベクトルによってのみ記述される。タスク（またはコマンド）ベクトルは、各制御パラメータを含む。したがって、エフェクタ軌道は、タスク空間で表される。このタスク記述は不変制御パラメータを含まず、図3に対応して記述される。

40

【0038】

図3には、図2での矢印10に対応する手9の実際の握りの軸が、ベクトル \mathbf{a}_{act} で表されている。握る対象物の対称軸は、目標の状態を表すターゲット・ベクトル \mathbf{a}_{tar}

50

get によって表される。目標の状態とは、各エフェクタの目標を定義する 1 組の制御パラメータである。手 9 が実行すべき動きは、握りの軸 a_{act} の角度 ϕ の回転に対応する。角度 ϕ は、現在の握りの軸 a_{act} とターゲット・ベクトル a_{target} との間の角度である。握りの軸 a_{act} およびターゲット・ベクトル a_{target} は、握りのベクトル a_{act} の 2 次元回転が実行される平面を定義する。現状の握りの軸 a_{act} は、エフェクタ、すなわち示された例ではロボット 1 の手 9 の現在の状態である。

【0039】

握りの軸 a_{act} の回転は、回転軸 a_{rot} 周りの回転によって記述することができる。この回転軸 a_{rot} は、現在の握りの軸 a_{act} およびターゲット・ベクトル a_{target} によって定義される平面に垂直である。握りの軸をターゲット・ベクトル a_{target} と一致する位置にもってくるためにさらに回転させる必要はないので、 a_{act} と a_{rot} とに垂直なベクトル $k e^x$ 周りの回転はゼロである。したがって、タスク・ベクトル $x_{2d} = (\quad 0)^T$ の各成分を構築する 2 つの制御コマンドだけが必要である。2 次元タスク・ベクトルは、不変制御パラメータを含まないタスク記述に基づいている。したがって、タスク系が確立され、直交単位ベクトル $k e^x$ 、 $k e^y$ および $k e^z$ によって定義される。単位ベクトル $k e^z$ は、エフェクタの対称軸と一致する。この場合、対称軸は握りの軸 10 である。

10

【0040】

次に、手 9 の軌道は、タスク・ベクトル x_{2d} の増分値 Δx_{2d} を関節の角度にマッピングすることによって計算される。これは、次式で記述することができる。

20

【数 7】

$$\delta q = J_{2d-act}^{\#} (d\phi \ 0)^T$$

ここで、 $J_{2d-act}^{\#}$ は、2 次元ヤコビアン of の擬似逆行列である。

【0041】

制御パラメータの増分値を計算するために、まず実際の握りの軸 a_{act} とターゲット・ベクトル a_{target} との間の回転角度を計算しなければならない。その角度 ϕ は次式の通りである。

【数 8】

$$\phi = \arccos \left(\frac{{}_I e_{sl}^Z \circ {}_I a_{cmd}}{\| {}_I e_{sl}^Z \| \| {}_I a_{cmd} \|} \right)$$

30

【0042】

単位ベクトル $k e^x$ 、 $k e^y$ および $k e^z$ は、以下で添字「sl」で表され、タスク記述のために使用される系を定義する。慣性系 2 の手の先端の系 8 への回転行列はすでに計算されているので、sl の系から手の系 8 への回転行列は次式の通り計算することができる。

40

【数 9】

$$A_{ht-sl} = A_{ht-I} A_{I-sl}$$

【0043】

必要とされる角速度は、コマンド軸（時定数 T）に収束する 1 次低域通過フィルタとして記述することができる。

【数 1 0】

$${}_{sl}\omega = \begin{pmatrix} 0 \\ -\frac{\phi}{T} \\ 0 \end{pmatrix}$$

【0044】

握りの軸を軸 a_{rot} の周りに回転させる既知の角速度 ${}_{sl}\omega$ を用いて、必要とされる手の固定角速度を計算することができる。それは、回転行列 A_{ht-sl} に基づいて、手の固定角度の x および z 方向の増分値を計算することができることを意味する。

10

【数 1 1】

$${}_{ht}\omega = A_{ht-sl} {}_{sl}\omega$$

【0045】

添字「ht」は、手の固定座標系を表す。実際の握りの軸 a_{act} が回転軸 a_{rot} 周りで回転する結果、球の表面上での最短接続としての経路が得られる。この最短接続 11 はエフェクタ軌道であり、図 4 に示されている。球の表面上のこの最短接続は、実際の握りの軸 a_{act} の矢印により、ターゲット・ベクトル a_{target} の位置へと角度だけ回転されることによって記述される。

20

【0046】

球の表面上の最短接続として経路を計算することによる前述の軌道制御に加えて、他のいくつかの軌道生成方法が可能である。たとえば、ベクトル終点間の経路を直線補間によって計算すること、または生物学的に触発を受けた方法（躍度最小モデル）を使用することなどである。

【0047】

本発明はまた、速度および加速度プロファイルの具体的な決定方法に限定されない。これらは、速度傾斜フィルタ (velocity ramp filter)、5 次以上の多項式の高次低域通過フィルタによって決定することができる。

30

【0048】

軌道の不変制御パラメータを決定するために使用することができるこうした回転対称性の他の例は、カメラの軸である。ほとんどの場合、カメラの視軸 (view axis) 周りの回転は重要ではない。カメラの視軸は、カメラの光軸 (optical axis) である。したがって、カメラの動作は、ロボットの手 9 の前述の握り動作に対応する。

【0049】

これは、ロボット 1 がつかむ対象物にカメラが焦点を合わせる場合に、特に明らかである。カメラの視軸がカメラの現在位置と対象物の位置とを結ぶ線に一致する限り、対象物の位置に接近する間にカメラがその対称軸の周りを回転するかどうかは問題ではない。このことは、軌道が、視軸周りのカメラの回転に対して不変であることを意味する。目標の状態の対称性について詳細に記述されてきたように、たとえば不変制御パラメータを決定するために、軌道が対称軸周りのエフェクタの回転に対して不変であることも可能である。この場合も、これによって零空間の次元が増大する。

40

【0050】

一般に、配置空間上のマッピングは、不変次元を含まないタスク空間のマッピング、および零空間のマッピングに分割することができる。このマッピングの分離は次のように公式化することができる。

$$q = J^{\#} x_{cmd} + N$$

【0051】

50

この分離について、 N は関節角度増分値を零空間にマッピングする線形行列である。マッピング行列 N は次式の通り計算することができる。

$$N = E - J \# J$$

【0052】

零空間運動はタスク空間運動と直交し、追加的な基準（消費電力、制御の頑強性など）を最適化するのに利用することができる。ロボット工学の分野におけるこうした追加の基準についての具体例は、たとえば関節限界を避けるため、特異姿勢を避けるため、または障害物を避けるために使用することができる。1つの最適化基準を適用するだけでなく、様々な最適化基準を重畳することが可能である。

【0053】

軌道を制御するための方法は、たとえばマニピュレータまたはカメラ・システムの中にプログラムされる。カメラ・システムが駆動され、その結果、カメラの対称軸をどんな方向に向けることもできる。こうしたカメラ・システムは、運転者支援システムとして自動車に設置することができる。

【0054】

さらに、この方法が実行されるような方法でマニピュレータをプログラムすることが有用である。2つ以上のこうしたマニピュレータは、駆動式カメラ・システムを追加的に装備したロボットに組み込むことができる。駆動式カメラ・システムは、たとえばロボット1の頭部に集積される。次いで、零空間の増大した次元は、改善された動きのための追加の基準を適用するために使用することができる。

【0055】

エフェクタの軌道を制御するための一般的な手順が、図5のブロック図に示されている。ステップ21において、センサ・データ(20)に基づいてタスク記述の順運動学が計算される。このタスク記述は不変制御パラメータを含まない。ステップ21で計算された順運動学から、次のステップ22において、タスク記述のヤコビアンが組み立てられ、逆運動学が計算される。

【0056】

次いで、ステップ24で、現在の状態と目標の状態との間の差が計算される。現在の状態と目標の状態との間の差を計算するために、コマンドまたはタスク・データを定義しなければならない。コマンドまたはタスク・データの定義は、たとえば扱うべき対象物の位置および方向に基づいて行うことができる。

【0057】

現在の状態と目標の状態との間の差を知った上で、ステップ25において、タスク・ベクトルの各要素の増分値が、以前に説明した軌道生成方法に従って計算される。次いで、タスク・ベクトルの各要素の増分値は、状態ベクトル上に射影される(ステップ26)。最終ステップ27において、エフェクタを所望の位置にもってくるために、ロボット関節がそれ相応に動かされる。

【0058】

本発明の範囲が、記述された実施形態によって制限されないことは明白である。

【図面の簡単な説明】

【0059】

【図1】ロボットの運動学モデルを示す図である。

【図2】考慮すべき第1の対称性側面を示す図である。

【図3】図2の第1の対称性側面に関するタスク記述を示す図である。

【図4】球の円周上に生成される軌道を示す図である。

【図5】エフェクタの軌道を制御するための方法を示すブロック図である。

【符号の説明】

【0060】

9 エフェクタ

10 エフェクタ軸 (a a c t)

10

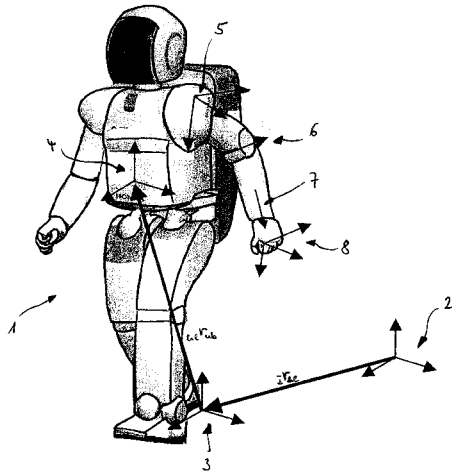
20

30

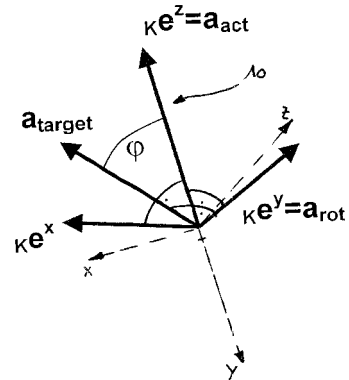
40

50

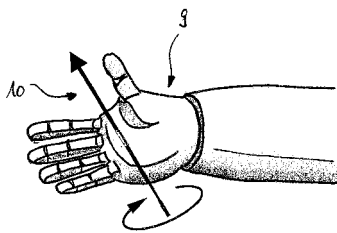
【図1】



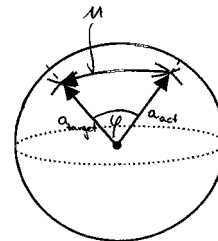
【図3】



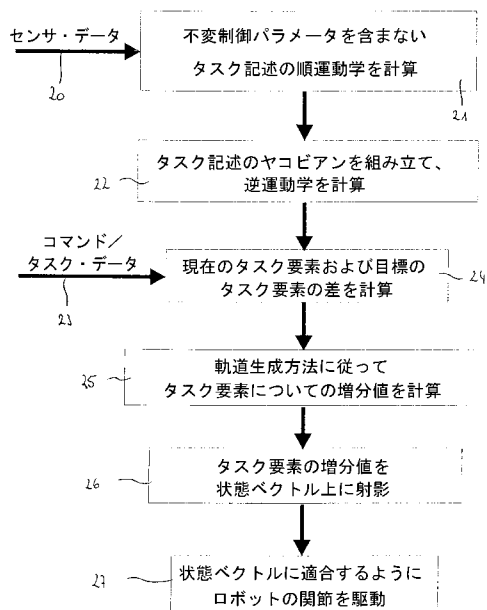
【図2】



【図4】



【図5】



フロントページの続き

審査官 松浦 陽

- (56)参考文献 特開2004-188533(JP,A)
特開2005-074600(JP,A)
特開2006-130580(JP,A)
特開2005-005608(JP,A)
按田 和幸 Kazuyuki Anda, 隠れ構造を用いた動作データの変形法 Deformation of motion data with a hidden structure, 情報処理学会研究報告 Vol.2002 No.77 IPSJ SIG Notes, 日本, 社団法人情報処理学会 Information Processing Society of Japan, 第2002巻

(58)調査した分野(Int.Cl., DB名)

B25J 1/00 - 21/02
G05D 3/00
JSTPlus(JDreamII)
IEEE Xplore
CiNii