

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6781101号
(P6781101)

(45) 発行日 令和2年11月4日(2020.11.4)

(24) 登録日 令和2年10月19日(2020.10.19)

(51) Int.Cl.		F I	
G05B 13/04	(2006.01)	G05B 13/04	
B25J 5/00	(2006.01)	B25J 5/00	E
G05B 13/02	(2006.01)	B25J 5/00	F
		G05B 13/02	J

請求項の数 8 (全 49 頁)

(21) 出願番号	特願2017-88614 (P2017-88614)	(73) 特許権者	504132272 国立大学法人京都大学
(22) 出願日	平成29年4月27日 (2017.4.27)		京都府京都市左京区吉田本町36番地1
(65) 公開番号	特開2018-185747 (P2018-185747A)	(73) 特許権者	000003207 トヨタ自動車株式会社
(43) 公開日	平成30年11月22日 (2018.11.22)		愛知県豊田市トヨタ町1番地
審査請求日	令和1年11月8日 (2019.11.8)	(74) 代理人	100103894 弁理士 冢入 健
特許法第30条第2項適用	(1) 平成29年2月6日 学士論文「状態ジャンプを含む非線形モデル予測制御 に対するC/GMRES法の拡張」に記載 (2) 平成 29年2月13日開催の「国立大学法人京都大学工学部 物理工学科士発表会」にて発表	(72) 発明者	大塚 敏之 京都府京都市左京区吉田本町36番地1 国立大学法人京都大学内
		(72) 発明者	片山 想太郎 京都府京都市左京区吉田本町36番地1 国立大学法人京都大学内

最終頁に続く

(54) 【発明の名称】 非線形システムの制御方法、二足歩行ロボットの制御装置、二足歩行ロボットの制御方法及びそのプログラム

(57) 【特許請求の範囲】

【請求項1】

非線形システムの制御方法であって、
前記非線形システムの状態を示す状態パラメータを取得する取得ステップと、
前記取得された状態パラメータに基づいて、モデル予測制御のアルゴリズムを使用して、
前記非線形システムを制御するための制御入力値を算出する算出ステップと、
前記算出された制御入力値を用いて、前記非線形システムを制御する制御ステップと
を有し、
前記算出ステップにおいて、指定されたタイミングにおいて状態が不連続に変化するよ
うに前記非線形システムの状態を拘束する拘束パラメータを用いて、前記非線形システム
の制御周期ごとに、前記モデル予測制御のアルゴリズムにおける予め定められた評価区
間における前記制御入力値の最適解の変化率を算出し、前記変化率を用いて当該制御周
期の次の制御周期における前記制御入力値の最適解を算出し、前記最適解から、現在
の前記制御入力値を算出する

非線形システムの制御方法。

【請求項2】

2つの脚を用いて二足歩行を行うことが可能な二足歩行ロボットの動作を制御する二足
歩行ロボットの制御装置であって、

前記二足歩行ロボットの歩行に関する状態を示す状態パラメータを取得する状態取得手
段と、

前記取得された状態パラメータに基づいて、モデル予測制御のアルゴリズムを使用して、前記二足歩行ロボットの動作を制御するための制御入力値を算出する算出手段と、前記算出された制御入力値を用いて、前記二足歩行ロボットの動作を制御する制御手段と

を有し、

前記算出手段は、指定されたタイミングにおいて前記2つの脚のうちの遊脚が着地するように前記二足歩行ロボットの状態を拘束する拘束パラメータを用いて、前記二足歩行ロボットの制御周期ごとに、前記モデル予測制御のアルゴリズムにおける予め定められた評価区間における前記制御入力値の最適解の変化率を算出し、前記変化率を用いて当該制御周期の次の制御周期における前記制御入力値の最適解を算出し、前記最適解から、現在の前記制御入力値を算出する

10

二足歩行ロボットの制御装置。

【請求項3】

前記拘束パラメータは、前記モデル予測制御のアルゴリズムで用いられる評価関数に含まれている

請求項2に記載の二足歩行ロボットの制御装置。

【請求項4】

前記拘束パラメータは、前記タイミングにおいて遊脚が着地したときの前記二足歩行ロボットの姿勢を指定する

請求項2又は3に記載の二足歩行ロボットの制御装置。

20

【請求項5】

前記拘束パラメータは、前記タイミングにおいて遊脚が着地したときの前記2つの脚の関節部の目標角度を指定する

請求項4に記載の二足歩行ロボットの制御装置。

【請求項6】

前記拘束パラメータは、調整可能なゲインを含む

請求項2から5のいずれか1項に記載の二足歩行ロボットの制御装置。

【請求項7】

2つの脚を用いて二足歩行を行うことが可能な二足歩行ロボットの動作を制御する二足歩行ロボットの制御方法であって、

30

前記二足歩行ロボットの歩行に関する状態を示す状態パラメータを取得する取得ステップと、

前記取得された状態パラメータに基づいて、モデル予測制御のアルゴリズムを使用して、前記二足歩行ロボットの動作を制御するための制御入力値を算出する算出ステップと、

前記算出された制御入力値を用いて、前記二足歩行ロボットの動作を制御する制御ステップと

を有し、

前記算出ステップにおいて、指定されたタイミングにおいて前記2つの脚のうちの遊脚が着地するように前記二足歩行ロボットの状態を拘束する拘束パラメータを用いて、前記二足歩行ロボットの制御周期ごとに、前記モデル予測制御のアルゴリズムにおける予め定められた評価区間における前記制御入力値の最適解の変化率を算出し、前記変化率を用いて当該制御周期の次の制御周期における前記制御入力値の最適解を算出し、前記最適解から、現在の前記制御入力値を算出する

40

二足歩行ロボットの制御方法。

【請求項8】

2つの脚を用いて二足歩行を行うことが可能な二足歩行ロボットの動作を制御する二足歩行ロボットの制御方法を実現するプログラムであって、

前記二足歩行ロボットの歩行に関する状態を示す状態パラメータを取得する取得ステップと、

前記取得された状態パラメータに基づいて、モデル予測制御のアルゴリズムを使用して

50

、前記二足歩行ロボットの動作を制御するための制御入力値を算出する算出ステップであって、指定されたタイミングにおいて前記2つの脚のうちの遊脚が着地するように前記二足歩行ロボットの状態を拘束する拘束パラメータを用いて、前記二足歩行ロボットの制御周期ごとに、前記モデル予測制御のアルゴリズムにおける予め定められた評価区間における前記制御入力値の最適解の変化率を算出し、前記変化率を用いて当該制御周期の次の制御周期における前記制御入力値の最適解を算出し、前記最適解から、現在の前記制御入力値を算出する、算出ステップと、

前記算出された制御入力値を用いて、前記二足歩行ロボットの動作を制御する制御ステップと

をコンピュータに実行させるプログラム。

10

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、非線形システムの制御方法、二足歩行ロボットの制御装置、二足歩行ロボットの制御方法及びそのプログラムに関する。

【背景技術】

【0002】

二足歩行ロボットのような安定性の低いシステムを制御する際には、未来（有限時間後まで）のシステム挙動を予測しながら制御を行うモデル予測制御（receding horizon control；リシーディングホライズン制御）を用いることが有効である。モデル予測制御は、制御周期（サンプリング周期）ごとに各時刻から有限時間未来までの最適制御問題を解き、制御入力値を決定するフィードバック制御である。

20

【0003】

フィードバック制御において、二足歩行ロボットは歩行動作等を伴う非線形性の高いシステムであるので、二足歩行ロボットの制御は、非線形モデル予測制御によって行われることが好ましい。ここで、非線形モデル予測制御は、一般に、多大な計算時間を要する。したがって、非線形モデル予測制御を用いて実時間（リアルタイム）で制御入力値の最適解を決定することは困難であった。

【0004】

この技術に関連し、非特許文献1は、非線形モデル予測制御を実時間（リアルタイム）で行うことが可能な、C/GMRES法（continuation/generalized minimum residual method）と呼ばれる技術を開示する。C/GMRES法は、連続変形法（continuation method）とGMRES法とを組み合わせたアルゴリズムである。C/GMRES法は、状態変化が連続であるシステムに対し、最適解の連続性を利用して、最適解の変化率を求めながら最適解を追跡していく計算方法である。このC/GMRES法を用いることにより、非線形モデル予測制御においても、実時間でシステムを制御することが可能となる。

30

【先行技術文献】

【非特許文献】

【0005】

【非特許文献1】Toshiyuki OHTSUKA and Hironori A. FUJII、「Real-Time Receding-Horizon Control Algorithm for Nonlinear Systems」、計測自動制御学会論文集、1997年12月、Vol.33, No.12, p. 1131-1139

40

【発明の概要】

【発明が解決しようとする課題】

【0006】

非線形性を有するシステムである非線形システムでは、周囲の環境との物理的な接触を伴って移動するとき、物理的な接触により状態変化が不連続となる場合がある。一方、非特許文献1にかかる技術は、システムの状態変化が連続であることを前提としている。したがって、非特許文献1にかかる技術を用いて、物理的な接触を行う可能性がある非線形システムを制御することは困難である。したがって、非線形モデル予測制御を用いて、不連

50

続な状態変化を伴う非線形システムを実時間で制御することは困難であった。

【0007】

本発明は、不連続な状態変化を伴う非線形システムを実時間で制御することが可能な非線形システムの制御方法、二足歩行ロボットの制御装置、二足歩行ロボットの制御方法及びそのプログラムを提供する。

【課題を解決するための手段】

【0008】

本発明にかかる非線形システムの制御方法は、前記非線形システムの状態を示す状態パラメータを取得する取得ステップと、前記取得された状態パラメータに基づいて、モデル予測制御のアルゴリズムを使用して、前記非線形システムを制御するための制御入力値を算出する算出ステップと、前記算出された制御入力値を用いて、前記非線形システムを制御する制御ステップとを有し、前記算出ステップにおいて、指定されたタイミングにおいて状態が不連続に変化するように前記非線形システムの状態を拘束する拘束パラメータを用いて、前記非線形システムの制御周期ごとに、前記モデル予測制御のアルゴリズムにおける予め定められた評価区間における前記制御入力値の最適解の変化率を算出し、前記変化率を用いて当該制御周期の次の制御周期における前記制御入力値の最適解を算出し、前記最適解から、現在の前記制御入力値を算出する。

【0009】

本発明は、上述したように、指定されたタイミングにおいて状態が不連続に変化するように非線形システムの状態を拘束する拘束パラメータを用いることで、想定しているタイミングで、不連続な状態変化を起こさせることができる。したがって、非線形システムの制御に非線形モデル予測制御の理論を容易に適用でき、さらにC/GMRES法を適用することも可能となる。したがって、本発明は、不連続な状態変化を伴う非線形システムを実時間で制御することが可能となる。

【0010】

また、本発明にかかる二足歩行ロボットの制御装置は、2つの脚を用いて二足歩行を行うことが可能な二足歩行ロボットの動作を制御する二足歩行ロボットの制御装置であって、前記二足歩行ロボットの歩行に関する状態を示す状態パラメータを取得する状態取得手段と、前記取得された状態パラメータに基づいて、モデル予測制御のアルゴリズムを使用して、前記二足歩行ロボットの動作を制御するための制御入力値を算出する算出手段と、前記算出された制御入力値を用いて、前記二足歩行ロボットの動作を制御する制御手段とを有し、前記算出手段は、指定されたタイミングにおいて前記2つの脚のうちの遊脚が着地するように前記二足歩行ロボットの状態を拘束する拘束パラメータを用いて、前記二足歩行ロボットの制御周期ごとに、前記モデル予測制御のアルゴリズムにおける予め定められた評価区間における前記制御入力値の最適解の変化率を算出し、前記変化率を用いて当該制御周期の次の制御周期における前記制御入力値の最適解を算出し、前記最適解から、現在の前記制御入力値を算出する。

【0011】

また、本発明にかかる二足歩行ロボットの制御方法は、2つの脚を用いて二足歩行を行うことが可能な二足歩行ロボットの動作を制御する二足歩行ロボットの制御方法であって、前記二足歩行ロボットの歩行に関する状態を示す状態パラメータを取得する取得ステップと、前記取得された状態パラメータに基づいて、モデル予測制御のアルゴリズムを使用して、前記二足歩行ロボットの動作を制御するための制御入力値を算出する算出ステップと、前記算出された制御入力値を用いて、前記二足歩行ロボットの動作を制御する制御ステップとを有し、前記算出ステップにおいて、指定されたタイミングにおいて前記2つの脚のうちの遊脚が着地するように前記二足歩行ロボットの状態を拘束する拘束パラメータを用いて、前記二足歩行ロボットの制御周期ごとに、前記モデル予測制御のアルゴリズムにおける予め定められた評価区間における前記制御入力値の最適解の変化率を算出し、前記変化率を用いて当該制御周期の次の制御周期における前記制御入力値の最適解を算出し、前記最適解から、現在の前記制御入力値を算出する。

【0012】

また、本発明にかかるプログラムは、2つの脚を用いて二足歩行を行うことが可能な二足歩行ロボットの動作を制御する二足歩行ロボットの制御方法を実現するプログラムであって、前記二足歩行ロボットの歩行に関する状態を示す状態パラメータを取得する取得ステップと、前記取得された状態パラメータに基づいて、モデル予測制御のアルゴリズムを使用して、前記二足歩行ロボットの動作を制御するための制御入力値を算出する算出ステップであって、指定されたタイミングにおいて前記2つの脚のうちの遊脚が着地するように前記二足歩行ロボットの状態を拘束する拘束パラメータを用いて、前記二足歩行ロボットの制御周期ごとに、前記モデル予測制御のアルゴリズムにおける予め定められた評価区間における前記制御入力値の最適解の変化率を算出し、前記変化率を用いて当該制御周期の次の制御周期における前記制御入力値の最適解を算出し、前記最適解から、現在の前記制御入力値を算出する、算出ステップと、前記算出された制御入力値を用いて、前記二足歩行ロボットの動作を制御する制御ステップとをコンピュータに実行させる。

10

【0013】

本発明は、上述したように、指定されたタイミングにおいて遊脚が着地するように二足歩行ロボットの状態を拘束する拘束パラメータを用いることで、想定しているタイミングで、遊脚の着地といった不連続な状態変化を起こさせることができる。したがって、二足歩行ロボットの制御に非線形モデル予測制御の理論を容易に適用でき、さらにC/GMRES法を適用することも可能となる。したがって、本発明は、不連続な状態変化を伴う二足歩行ロボットの動作を実時間で制御することが可能となる。

20

【0014】

また、好ましくは、前記拘束パラメータは、前記モデル予測制御のアルゴリズムで用いられる評価関数に含まれている。これにより、不連続な状態変化の無い非線形モデル予測制御と同様に最適化問題を扱うことが可能となる。

【0015】

また、好ましくは、前記拘束パラメータは、前記タイミングにおいて遊脚が着地したときの前記二足歩行ロボットの姿勢を指定する。これにより、想定した姿勢で遊脚を着地させるように二足歩行ロボットを制御することが可能となる。

【0016】

また、好ましくは、前記拘束パラメータは、前記タイミングにおいて遊脚が着地したときの前記2つの脚の関節部の目標角度を指定する。これにより、想定した関節角度で遊脚を着地させるように関節部を制御することが可能となる。

30

【0017】

また、好ましくは、前記拘束パラメータは、調整可能なゲインを含む。これにより、制御装置の性能によらないで制御を安定化させることが可能となる。

【発明の効果】

【0018】

本発明によれば、不連続な状態変化を伴う非線形システムを実時間で制御することが可能な非線形システムの制御方法、二足歩行ロボットの制御装置、二足歩行ロボットの制御方法及びそのプログラムを提供できる。

40

【図面の簡単な説明】

【0019】

【図1】実施の形態1にかかるロボットシステムを示す概略図である。

【図2】実施の形態1にかかるロボットシステムの構成を示す機能ブロック図である。

【図3】非線形モデル予測制御を説明するための図である。

【図4】入力系列の更新について説明するための図である。

【図5】実施の形態1にかかるロボットをコンパス型モデルに適用する方法を説明するための図である。

【図6】実施の形態1にかかるロボットをコンパス型モデルに適用した例を示す図である。

50

【図 7】状態ジャンプを説明するための図である。

【図 8】遊脚リンクの衝突直前のロボットの状態を示す図である。

【図 9】遊脚リンクの衝突直後のロボットの状態を示す図である。

【図 10】実施の形態 1 にかかる制御装置によって行われるロボットの制御方法を示すフローチャートである。

【図 11】実施の形態 2 にかかるロボットを示す図である。

【図 12】実施の形態 2 にかかるロボットを膝屈曲モデルに適用した状態を示す図である。

【図 13】実施の形態 2 にかかるロボットを膝屈曲モデルに適用した例を示す図である。

【図 14】本実施の形態にかかる非線形システムに非線形モデル予測制御のアルゴリズムを適用したシミュレーション結果を示す図である。

10

【図 15】本実施の形態にかかる非線形システムに非線形モデル予測制御のアルゴリズムを適用したシミュレーション結果を示す図である。

【図 16】本実施の形態にかかる非線形システムに非線形モデル予測制御のアルゴリズムを適用したシミュレーション結果を示す図である。

【図 17】本実施の形態にかかる非線形システムに非線形モデル予測制御のアルゴリズムを適用したシミュレーション結果を示す図である。

【図 18】本実施の形態にかかる非線形システムに非線形モデル予測制御のアルゴリズムを適用したシミュレーション結果を示す図である。

【図 19】本実施の形態にかかる非線形システムに非線形モデル予測制御のアルゴリズムを適用したシミュレーション結果を示す図である。

20

【図 20】シミュレーション結果において定常状態の制御入力のグラフを示す図である。

【発明を実施するための形態】

【0020】

(実施の形態 1)

以下、図面を参照して本発明の実施の形態について説明する。なお、各図面において、同一の要素には同一の符号が付されており、必要に応じて重複説明は省略されている。

【0021】

図 1 は、実施の形態 1 にかかるロボットシステム 1 を示す概略図である。また、図 2 は、実施の形態 1 にかかるロボットシステム 1 の構成を示す機能ブロック図である。ロボットシステム 1 は、ロボット 100 と、ロボットの動作を制御する制御装置 2 とを有する。

30

【0022】

ロボット 100 は、胴体 102 と、2 つの脚である右脚 110R 及び左脚 110L とを有する。ロボット 100 は、2 つの脚（右脚 110R 及び左脚 110L）を用いて歩行動作を行うことが可能な二足歩行ロボットである。右脚 110R 及び左脚 110L は、ロボット 100 の胴体 102 の下部に設けられている。ここで、図 1 に示すように、ロボット 100 の前方向を X 軸方向、上方向を Y 軸方向とする。また、以下、右脚 110R に関する構成要素の符号に「R」を付し、左脚 110L に関する構成要素の符号に「L」を付すが、それぞれの構成要素について左右を区別しない場合には、「R」及び「L」は、適宜、省略され得る。

40

【0023】

右脚 110R は、胴体 102 に近い方から順に、股関節部 120R と、上腿部 112R と、膝関節部 122R と、下腿部 114R と、足首関節部 124R と、足部 116R とを有する。同様に、左脚 110L は、胴体 102 に近い方から順に、股関節部 120L と、上腿部 112L と、膝関節部 122L と、下腿部 114L と、足首関節部 124L と、足部 116L とを有する。足部 116R 及び足部 116L の底部には、それぞれ足裏センサ 118 が設けられている。足裏センサ 118 は、足部 116 の底部に加わる荷重を検出する。

【0024】

股関節部 120R 及び股関節部 120L は、胴体 102 の下部に取り付けられている。

50

そして、股関節部 1 2 0 R 及び股関節部 1 2 0 L を介して、それぞれ、上腿部 1 1 2 R 及び上腿部 1 1 2 L が胴体 1 0 2 と接続されている。言い換えると、右脚 1 1 0 R 及び左脚 1 1 0 L は、それぞれ、股関節部 1 2 0 R 及び股関節部 1 2 0 L を介して、胴体 1 0 2 と接続されている。

【 0 0 2 5 】

また、膝関節部 1 2 2 R を介して、上腿部 1 1 2 R と下腿部 1 1 4 R とが接続されている。同様に、膝関節部 1 2 2 L を介して、上腿部 1 1 2 L と下腿部 1 1 4 L とが接続されている。また、足首関節部 1 2 4 R を介して、下腿部 1 1 4 R と足部 1 1 6 R とが接続されている。同様に、足首関節部 1 2 4 L を介して、下腿部 1 1 4 L と足部 1 1 6 L とが接続されている。

10

【 0 0 2 6 】

股関節部 1 2 0 は、X Y 平面に垂直な軸（つまりロボット 1 0 0 の横方向に水平な軸）の周りに回転する。これにより、右脚 1 1 0 R 及び左脚 1 1 0 L は、前後に動作し得る。したがって、ロボット 1 0 0 は、右脚 1 1 0 R 及び左脚 1 1 0 L を交互に前に出すことにより歩行動作を行うことができる。

【 0 0 2 7 】

膝関節部 1 2 2 は、X Y 平面に垂直な軸の周りに回転する。これにより、右脚 1 1 0 R 及び左脚 1 1 0 L は、膝関節部 1 2 2 で屈曲動作を行うことができる。また、足首関節部 1 2 4 は、X Y 平面に垂直な軸の周りに回転する。これにより、足部 1 1 6 は、下腿部 1 1 4 に対して上下に動作し得る。

20

【 0 0 2 8 】

図 2 に示すように、ロボット 1 0 0 の各関節部（股関節部 1 2 0、膝関節部 1 2 2 及び足首関節部 1 2 4）は、角度センサ 1 3 0 と、モータ 1 4 0 とを有する。角度センサ 1 3 0 は、例えばエンコーダであって、各関節部の関節角度を検出する。モータ 1 4 0 は、各関節部を動作させる、アクチュエータとしての機能を有する。また、各関節部は、各関節部のモータ 1 4 0 のトルクを検出するトルクセンサ 1 3 6 を有してもよい。また、ロボット 1 0 0 の周囲の状態を検出するためのカメラが、胴体 1 0 2 に内蔵されていてもよい。

【 0 0 2 9 】

制御装置 2 は、例えばコンピュータとしての機能を有する。制御装置 2 は、ロボット 1 0 0 の内部（例えば胴体 1 0 2）に搭載されてもよい。また、制御装置 2 は、ロボット 1 0 0 と物理的に離れていてもよく、その場合、ロボット 1 0 0 と有線又は無線を介して通信可能に接続されてもよい。制御装置 2 は、ロボット 1 0 0 の動作、特に、右脚 1 1 0 R 及び左脚 1 1 0 L の動作を制御する。さらに具体的には、制御装置 2 は、各関節部のモータのトルクを制御することで、右脚 1 1 0 R 及び左脚 1 1 0 L の姿勢を制御する。つまり、ロボットシステム 1 において、制御装置 2 はマスタ装置としての機能を有し、ロボット 1 0 0 はスレーブ装置としての機能を有する。

30

【 0 0 3 0 】

制御装置 2 は、主要なハードウェア構成として、C P U (Central Processing Unit) 4 と、R O M (Read Only Memory) 6 と、R A M (Random Access Memory) 8 とを有する。C P U 4 は、制御処理及び演算処理等を行う演算装置としての機能を有する。R O M 6 は、C P U 4 によって実行される制御プログラム及び演算プログラム等を記憶するための機能を有する。R A M 8 は、処理データ等を一時的に記憶するための機能を有する。

40

【 0 0 3 1 】

また、制御装置 2 は、状態取得部 1 2、非線形モデル予測制御部 1 4、及びサーボ制御部 1 6（以下、「各構成要素」と称する）を有する。各構成要素は、例えば、C P U 4 が R O M 6 に記憶されたプログラムを実行することによって実現可能である。また、各構成要素は、必要なプログラムを任意の不揮発性記録媒体に記録しておき、必要に応じてインストールするようにして、実現するようにしてもよい。なお、各構成要素は、上記のようにソフトウェアによって実現されることに限定されず、何らかの回路素子等のハードウェアによって実現されてもよい。

50

【 0 0 3 2 】

状態取得部 1 2 は、ロボット 1 0 0 の現在の歩行に関する状態を示すデータ（状態パラメータ）を取得する、状態取得手段としての機能を有する。状態取得部 1 2 は、各センサ（角度センサ 1 3 0、足裏センサ 1 1 8 及びトルクセンサ 1 3 6）から、各センサの検出値を取得する。そして、状態取得部 1 2 は、取得された検出値（及び検出値から得られた値）を非線形モデル予測制御部 1 4 に対して出力する。なお、「検出値から得られた値」とは、例えば、「検出値」が角度センサ 1 3 0 から検出された関節角度である場合、関節角度の速度（変化量，時間微分）であってもよい。この場合、状態パラメータは、関節角度及び関節角度の速度を示してもよい。

【 0 0 3 3 】

非線形モデル予測制御部 1 4 は、ロボット 1 0 0 の動作を制御するための制御入力値（入力値）を算出する算出手段としての機能を有する。非線形モデル予測制御部 1 4 は、状態取得部 1 2 からの検出値（及び検出値から得られた値）の少なくとも一部を状態パラメータとして、その状態パラメータに基づいて、モデル予測制御のアルゴリズムを使用してロボット 1 0 0 の動作を制御するための制御入力値を算出する。また、非線形モデル予測制御部 1 4 は、算出された制御入力値をサーボ制御部 1 6 に対して出力する。詳しくは後述する。また、非線形モデル予測制御部 1 4 は、ロボットシステム 1 の外部の上位コントローラ（図示せず）によって、必要な指示値（歩幅、歩行周期等）を入力されてもよい。

【 0 0 3 4 】

サーボ制御部 1 6 は、非線形モデル予測制御部 1 4 によって算出された制御入力値を用いてロボット 1 0 0 の動作を制御する制御手段としての機能を有する。サーボ制御部 1 6 は、算出された制御入力値となるように、ロボット 1 0 0 の各関節部を制御する。また、サーボ制御部 1 6 は、サーボアンプの機能を有してもよい。また、サーボ制御部 1 6 は、トルク制御を行う場合、各関節部のトルク（関節トルク）が算出された制御入力値となるように、各関節部のモータ 1 4 0 を制御する。このとき、サーボ制御部 1 6 は、各関節部のトルクセンサ 1 3 6 によって検出されたトルク値を用いてフィードバック制御を行ってもよい。

【 0 0 3 5 】

（モデル予測制御）

ここで、本実施の形態にかかる非線形モデル予測制御部 1 4 によって行われる、モデル予測制御（非線形モデル予測制御）の手法の概要について説明する。非線形モデル予測制御とは、非線形システムに対し、各サンプリング時刻で有限時刻未来までの最適入力（制御入力値の最適解）を求め、得られた入力のうち初期値を実際の入力とする制御である。非線形モデル予測制御には、非線形最適制御である、フィードバック制御である、及び、拘束条件を組み込み易いという、3つの利点がある。

【 0 0 3 6 】

このように、非線形モデル予測制御は、フィードバック制御であるため外乱に対して強く、拘束条件も多様に組み合わせることができる。このような特徴があるため、非線形モデル予測制御は、多くのシステムへの導入が期待されている。しかしながら、ニュートン法などの従来の反復法では、サンプリング周期内で最適解に収束させることは困難であった。

【 0 0 3 7 】

近年、この問題に対する有効な数値計算法として、C / G M R E S 法が新たに考案された。C / G M R E S 法を用いることで有限時刻未来までの最適制御問題をサンプリング周期内で解くことが可能になった。しかしながら、以下に示すように、現状では、C / G M R E S 法による非線形モデル予測制御を適用できない場合もある。

【 0 0 3 8 】

すなわち、ロボット 1 0 0 等の非線形システムの多くは、状態が不連続に変化する事象（「状態ジャンプ」と称する）を伴い得る。この状態ジャンプを含む問題に対し、現状の C / G M R E S 法による非線形モデル予測制御を適用することは困難であった。状態ジャ

10

20

30

40

50

ンプを伴うシステムを直接最適化しようとする、状態ジャンプの時刻及び入力を同時に最適化する必要がある。また、状態ジャンプにかかるラグランジュ乗数がさらに追加されることになる。

【0039】

また、これらの問題をC/GMRES法を用いて数値的に解くには、C/GMRES法のアルゴリズムを大きく変える必要がある。そこで、本願の発明者らは、ペナルティ関数(ペナルティ項)を非線形モデル予測制御の評価関数に加えるという手法を見出した。ペナルティ関数を評価関数に加えることで、状態ジャンプが発生する時刻(本実施の形態においては遊脚が着地する時刻)を指定できる。また、ラグランジュ乗数を増やさずにC/GMRES法を適用することができる。

10

【0040】

まず、非線形モデル予測制御の概要について説明する。制御対象として、以下の式(1)で示すような状態方程式で表される非線形システムを考える。

【数1】

$$\dot{x}(t) = f(x(t), u(t)) \quad \cdot \cdot \cdot (1)$$

$$x(t) \in \mathcal{R}^n, u(t) \in \mathcal{R}^m$$

【0041】

ただし、 $x(t)$ は状態ベクトルであり、 $u(t)$ は制御入力ベクトル(制御入力値を示すベクトル)である。また、 \mathcal{R}^n 及び \mathcal{R}^m は、それぞれ、 n 次元実数ベクトル全体の集合及び m 次元実数ベクトル全体の集合を示す。非線形モデル予測制御とは、式(1)で表されるシステムに対し、各時刻 t において、以下の式(2)で表される評価関数を最小にする入力 $u_{opt}(t)$ を求め、その初期値 $u_{opt}(t)$ を時刻 t における実際の制御入力値 $u(t)$ とする制御である。

20

【数2】

$$J = \varphi(x(t+T)) + \int_0^T L(x(t+\tau), u(t+\tau))d\tau \quad \cdot \cdot \cdot (2)$$

30

【0042】

ここで、 T は時刻 t における評価区間の長さである。関数 $\varphi(x)$ は終端コストと呼ばれるスカラー値関数である。関数 $L(x, u)$ はステージコストと呼ばれるスカラー値関数である。 T は評価区間における時間パラメータであって、 $0 < T$ である。なお、 T は、通常、正のスカラー値 T_f 及び $(\alpha > 0)$ を用いて以下の式(3)のように与えられる。

【数3】

$$T(t) = T_f(1 - e^{-\alpha t}) \quad \cdot \cdot \cdot (3)$$

40

【0043】

このように、非線形モデル予測制御は、各時刻 t で状態に基づいた最適入力を求めているため、フィードバック制御となっている。

また、以下の式(4)で表されるベクトル関数を拘束条件として与えることもできる。

【数4】

$$C(x(t), u(t)) = 0 \quad \cdot \cdot \cdot (4)$$

なお、拘束条件については、等式拘束条件だけでなく、不等式拘束条件を組み込むこと

50

もできる。

【0044】

図3は、非線形モデル予測制御を説明するための図である。図3に示すように、非線形モデル予測制御では、現在時刻 t_0 において T 秒後までの期間のモデル挙動を予測して最適化計算を行って入力 $u_{opt}(t_0 + \tau)$ を求める。そして、その初期値 $u_{opt}(t_0)$ を現在時刻 t_0 における実際の制御入力値 $u(t_0)$ とする。

【0045】

同様に、制御周期であるサンプリング周期 Δt 秒後に、その時点での現在時刻 t_1 において T 秒後までの期間のモデル挙動を予測して最適化計算を行って入力 $u_{opt}(t_1 + \tau)$ を求める。そして、その初期値 $u_{opt}(t_1)$ を現在時刻 t_1 における実際の制御入力値 $u(t_1)$ とする。以下同様に、サンプリング周期ごとに、制御入力値 $u(t)$ が算出されることとなる。

10

【0046】

非線形モデル予測制御で各時刻 t において解くべき問題は、評価区間上の時刻 τ について以下の式(5)~(8)に示すような最適制御問題である。

【数5】

$$\frac{d}{d\tau} x^*(\tau; t) = f(x^*(\tau; t), u^*(\tau; t)) \quad \dots (5)$$

20

$$x^*(0; t) = x(t) \quad \dots (6)$$

$$J = \varphi(x^*(\tau; T)) + \int_0^T L(x^*(\tau; t), u^*(\tau; t)) d\tau \quad \dots (7)$$

$$C(x^*(\tau; t), u^*(\tau; t)) = 0 \quad \dots (8)$$

ただし、時刻 t における評価区間上の時刻 τ ($0 \leq \tau \leq T$) の状態変数ベクトルと制御入力ベクトルとを、それぞれ、 $x^*(\tau; t) = x^*(\tau + \Delta t; t)$ 、 $u^*(\tau; t) = u^*(\tau + \Delta t; t)$ とした。また、添字 $*$ は、評価区間上の値であることを示す。

30

【0047】

式(7)の J を汎関数とみなして変分法を用いて停留条件を求めると、最適制御の必要条件(オイラー・ラグランジュ方程式)が、以下の式(9)~(14)のように得られる。

【数6】

$$\frac{d}{d\tau} x^*(\tau; t) = f(x^*(\tau; t), u^*(\tau; t)) \quad \cdot \cdot \cdot (9)$$

$$x^*(0; t) = x(t) \quad \cdot \cdot \cdot (10)$$

$$\frac{d}{d\tau} \lambda^*(\tau; t) = - \left(\frac{\partial H}{\partial x} \right)^T (x^*(\tau; t), u^*(\tau; t), \lambda^*(\tau; t), \mu^*(\tau; t)) \quad \cdot \cdot \cdot (11)$$

$$\lambda^*(T; t) = \left(\frac{\partial \varphi}{\partial x} \right)^T (x^*(T; t)) \quad \cdot \cdot \cdot (12)$$

$$\left(\frac{\partial H}{\partial u} \right)^T (x^*(\tau; t), u^*(\tau; t), \lambda^*(\tau; t), \mu^*(\tau; t)) = 0 \quad \cdot \cdot \cdot (13)$$

$$C(x^*(\tau; t), u^*(\tau; t)) = 0 \quad \cdot \cdot \cdot (14)$$

【0048】

ここで、

【数7】

$$H(x, u, \lambda, \mu) = L(x, u) + \lambda^T f(x, u) + \mu^T C(x, u) \quad \cdot \cdot \cdot (15)$$

は、ハミルトン関数と呼ばれるスカラー値関数であり、 $x^*(\tau; t)$ は式(9)に対する随伴変数、 $\mu^*(\tau; t)$ は式(14)に対するラグランジュ乗数である。

【0049】

一方、実際の数値計算は、すべて離散化して行われる。したがって、式(9)～(14)は、すべて離散近似して扱わなければならない。そこで、評価区間(0, T)をNステップに離散近似することを考える。その際の評価区間の時間刻みを、以下の式(16)で示すようにする。

【数8】

$$\Delta\tau = \frac{T}{N} \quad \cdot \cdot \cdot (16)$$

【0050】

その上で、評価区間上のi番目(1 ≤ i ≤ N)のステップ、つまり時刻 t + iΔτ における状態を、

【数9】

$$x^*(i\Delta\tau; t) = x_i^*(t)$$

と表す。u^{*}_i(t)、λ^{*}_i(t) 及び μ^{*}_i(t) についても同様に表される。なお、T 及び N は、予め定められた値である。したがって、Δτ も、予め定められた値である。

【0051】

10

20

30

40

50

この条件の下で式(9)~(14)を離散近似すると、以下の式(17)~(22)で示すような、離散近似されたオイラー・ラグランジュ方程式が得られる。

【数10】

$$x_{i+1}^*(t) = x_i^*(t) + f(x_i^*(t), u_i^*(t))\Delta\tau \quad \cdot \cdot \cdot (17)$$

$$x_0^*(t) = x(t) \quad \cdot \cdot \cdot (18)$$

$$\lambda_i^*(t) = \lambda_{i+1}^*(t) - \left(\frac{\partial H}{\partial x} \right)^T (x_i^*(t), u_i^*(t), \lambda_{i+1}^*(t), \mu_i^*(t))\Delta\tau \quad 10$$

$$\cdot \cdot \cdot (19)$$

$$\lambda_N^*(t) = \left(\frac{\partial \varphi}{\partial x} \right)^T (x_N^*(t)) \quad \cdot \cdot \cdot (20)$$

$$\left(\frac{\partial H}{\partial u} \right)^T (x_i^*(t), u_i^*(t), \lambda_{i+1}^*(t), \mu_i^*(t)) = 0 \quad \cdot \cdot \cdot (21) \quad 20$$

$$C(x_i^*(t), u_i^*(t)) = 0 \quad \cdot \cdot \cdot (22)$$

【0052】

但し、離散近似されたハミルトン関数は、以下の式(23)で定義される。

【数11】

$$H(x_i, u_i, \lambda_{i+1}, \mu_i) = L(x_i, u_i) + \lambda_{i+1}^T f(x_i, u_i) + \mu_i^T C(x_i, u_i) \quad \cdot \cdot \cdot (23) \quad 30$$

【0053】

以上より、非線形モデル予測制御における最適入力を求める問題というのは、上記の式(17)~(22)を解いて、 $i=0$ から $i=N$ について、 $x_i^*(t)$ 、 $u_i^*(t)$ 、 $\lambda_i^*(t)$ 及び $\mu_i^*(t)$ を求めるという問題に帰着される。

ここで、 $x_i^*(t)$ は、上記式(17)、(18)より陽に求められる。また、 $\lambda_i^*(t)$ は、求められた $x_i^*(t)$ と上記式(19)、(20)とから陽に求められる。したがって、 $x_i^*(t)$ 、 $u_i^*(t)$ 、 $\lambda_i^*(t)$ 及び $\mu_i^*(t)$ のうちの本質的な未知量は、以下の式(24)で表されるベクトル $U(t)$ で定義される。

【数12】

$$U(t) := \begin{bmatrix} u_0^*(t) \\ \mu_0^*(t) \\ \vdots \\ u_{N-1}^*(t) \\ \mu_{N-1}^*(t) \end{bmatrix} \quad \cdot \cdot \cdot (24) \quad 40$$

なお、「:=」は、定義を意味する等号である。つまり、上記式(24)において、左辺 $U(t)$ は、右辺のベクトルで定義される。

【0054】

そして、この $U(t)$ は、以下の式(25)で示される方程式を解くことによって得られる。

【数13】

$$F(U(t), x(t), t)$$

$$:= \begin{bmatrix} \left(\frac{\partial H}{\partial u} \right)^T (x_0^*(t), u_0^*(t), \lambda_1^*(t), \mu_0^*(t)) \\ C(x_0^*(t), u_0^*(t)) \\ \vdots \\ \left(\frac{\partial H}{\partial u} \right)^T (x_{N-1}^*(t), u_{N-1}^*(t), \lambda_N^*(t), \mu_{N-1}^*(t)) \\ C(x_{N-1}^*(t), u_{N-1}^*(t)) \end{bmatrix} = 0$$

10

20

• • • (25)

【0055】

次に、C/GMRES法について説明する。ニュートン法などの反復法では、式(25)を各サンプリング時間内で解くのは難しい。一方、 $U(t)$ が時刻 t に関して連続であれば、 $U(t)$ は、サンプリング周期 t ごとに、以下の式(26)で示されるようにして更新され得る。

30

【数14】

$$U(t + \Delta t) = U(t) + \dot{U}(t)\Delta t \quad \cdot \cdot \cdot (26)$$

【0056】

したがって、 $U(t)$ を求めるためには、 $t = 0$ では $U(0)$ を求め、 $t > 0$ では、 $U(t)$ の時間微分、つまり $\dot{U}(t)$ の変化量である

【数15】

$$\dot{U}(t)$$

40

を求めればよい。なお、式(26)の計算は、上記の $U(t)$ の変化量を数値積分することに対応する。

【0057】

ここで、

【数16】

$$\dot{U}(t)$$

50

を求めるために、式(25)が全ての t で成り立つことを考慮して、上記式(25)と等価である、以下の式(27)で表される方程式を扱うことを考える。

【0058】

【数17】

$$\frac{d}{dt}F(U(t), x(t), t) = 0, \quad F(U(0), x(0), 0) = 0 \quad \cdot \cdot \cdot (27)$$

【0059】

さらに、式(27)は、以下の式(28)で示すように書き換えられ得る。

10

【数18】

$$\frac{d}{dt}F(U(t), x(t), t) = -\zeta F(U(t), x(t), t), \quad F(U(0), x(0), 0) = 0 \quad \cdot \cdot \cdot (28)$$

但し、 ζ は正の実数で、安定化パラメータと呼ばれる。

【0060】

式(28)の全微分を実行して整理すると、 U の変化量

20

【数19】

$$\dot{U}$$

は、次の式(29)で表される連立方程式を解くことで得られる。

【0061】

【数20】

$$\frac{\partial F}{\partial U} \dot{U} = -\zeta F - \frac{\partial F}{\partial x} \dot{x} - \frac{\partial F}{\partial t} \quad \cdot \cdot \cdot (29)$$

30

したがって、 $t > 0$ のとき、各時刻で解くべき問題は、上記式(29)で示される連立方程式のみとなる。さらに、式(29)で示される連立方程式の数値解法として、連立方程式から少ない反復回数で高精度な解を得ることが可能なGMRES法を用いることができる。

【0062】

上述した手法が、非線形モデル予測制御の実時間最適化アルゴリズムである。この $U(t)$ の連続性を利用した変形法(連続変形法; Continuation method)とGMRES法を組み合わせたアルゴリズムを、C/GMRES法と称する。

【0063】

40

次に、状態ジャンプを考慮した非線形モデル予測制御について説明する。状態 $x(t) \in \mathbb{R}^n$ が以下の式(30)で表される条件を満たすと、その直後より状態ジャンプが生じるシステムを仮定する。

【数21】

$$\xi(x(t)) = 0 \quad \cdot \cdot \cdot (30)$$

ただし、 $x(t) \in \mathbb{R}^n$ はベクトル値である。

【0064】

ここで、状態ジャンプが、時刻 t_j の前後で起こるとする。つまり、 t_j の直前及び直

50

後の時刻をそれぞれ「 t_j^- 」及び「 t_j^+ 」と表すと、ここで仮定されるシステムは、以下の式(31)を満たしている。

【数22】

$$\xi(x(t_j^-)) = 0 \quad \cdot \cdot \cdot (31)$$

【0065】

また、ここで仮定されるシステムは、状態ジャンプ直後の状態 $x(t_j^+)$ が、以下の式(32)で表されるように、状態ジャンプ直前の状態 $x(t_j^-)$ から陽に求められるとする。

【数23】

$$x(t_j^+) = \psi(x(t_j^-)) \quad \cdot \cdot \cdot (32)$$

【0066】

さらに、状態ジャンプとともにシステムの状態方程式が切り替わり得るとする。このとき、状態ジャンプの前後の状態方程式は、以下の式(33)、(34)で示すように記述される。

【数24】

$$\dot{x}(t) = f_1(x(t), u(t)) \quad (t < t_j) \quad \cdot \cdot \cdot (33)$$

$$\dot{x}(t) = f_2(x(t), u(t)) \quad (t_j < t) \quad \cdot \cdot \cdot (34)$$

なお、状態方程式は必ずしも切り替わる必要はなく、 $f_1(x, u) = f_2(x, u)$ であってもよい。また、説明の明確化のため、ここで扱うシステムでは、式(4)で示したような拘束条件は存在しないと仮定する。

【0067】

非線形モデル予測制御の問題を考えるために、以下の説明では、評価区間上でのジャンプ時刻を τ_j とする。つまり、 τ_j は、以下の式(35)、(36)を満たす。

【数25】

$$\tau_j = t_j - t \quad \cdot \cdot \cdot (35)$$

$$0 < \tau_j < T \quad \cdot \cdot \cdot (36)$$

【0068】

一般的に、上記のシステムを最適化するためには、変分法により導かれた停留条件を数値的に解けばよい。システムの方程式が上記式(33)、(34)、(32)で与えられ、 t_j が固定されていないとき、停留条件は、以下の式(37)~(48)で与えられる。

10

20

30

40

【数 2 6】

$$\frac{d}{d\tau} x^*(\tau; t) = f_1(x^*(\tau; t), u^*(\tau; t)) \quad (\tau < \tau_j) \quad \cdot \cdot \cdot (37)$$

$$\frac{d}{d\tau} x^*(\tau; t) = f_2(x^*(\tau; t), u^*(\tau; t)) \quad (\tau > \tau_j) \quad \cdot \cdot \cdot (38)$$

$$x^*(\tau_j+; t) = \psi(x^*(\tau_j-; t)) \quad \cdot \cdot \cdot (39)$$

$$x^*(0; t) = x(t) \quad \cdot \cdot \cdot (40)$$

$$\xi(x^*(\tau_j-; t)) = 0 \quad \cdot \cdot \cdot (41)$$

$$\lambda^*(T; t) = \left(\frac{\partial \varphi}{\partial x} \right)^T (x^*(T; t)) \quad \cdot \cdot \cdot (42)$$

$$\dot{\lambda}^*(\tau; t) = - \left(\frac{\partial H_1}{\partial x} \right)^T (x^*(\tau; t), u^*(\tau; t), \lambda^*(\tau; t)) \quad (\tau < \tau_j) \quad \cdot \cdot \cdot (43)$$

$$\dot{\lambda}^*(\tau; t) = - \left(\frac{\partial H_2}{\partial x} \right)^T (x^*(\tau; t), u^*(\tau; t), \lambda^*(\tau; t)) \quad (\tau > \tau_j) \quad \cdot \cdot \cdot (44)$$

$$\lambda^*(\tau_j-; t) = \left(\frac{\partial \psi}{\partial x} \right)^T (x^*(\tau_j-; t)) \lambda^*(\tau_j+; t) + \left(\frac{\partial \xi}{\partial x} \right)^T (x^*(\tau_j-; t)) \nu \quad \cdot \cdot \cdot (45)$$

$$\left(\frac{\partial H_1}{\partial u} \right)^T (x^*(\tau; t), u^*(\tau; t), \lambda^*(\tau; t)) = 0 \quad (\tau < \tau_j) \quad \cdot \cdot \cdot (46)$$

$$\left(\frac{\partial H_2}{\partial u} \right)^T (x^*(\tau; t), u^*(\tau; t), \lambda^*(\tau; t)) = 0 \quad (\tau > \tau_j) \quad \cdot \cdot \cdot (47)$$

$$H_1(x^*(\tau_j-; t), u^*(\tau_j-; t), \lambda^*(\tau_j-; t)) - H_2(x^*(\tau_j+; t), u^*(\tau_j+; t), \lambda^*(\tau_j+; t)) = 0 \quad \cdot \cdot \cdot (48)$$

但し、 λ^* は、は式 (31) に対するラグランジュ乗数である。

【0069】

また、ハミルトン関数 H_1 , H_2 は、それぞれ以下の式 (49) , (50) で表される 50

【数 2 7】

$$H_1(x, u, \lambda) = L(x, u) + \lambda^T f_1(x, u) \quad \cdot \cdot \cdot (49)$$

$$H_2(x, u, \lambda) = L(x, u) + \lambda^T f_2(x, u) \quad \cdot \cdot \cdot (50)$$

【0070】

上記の停留条件の中で、時刻 t_j に関する条件を示す式は、式(41)、(48)である。ここで、上述したように実際には離散化して数値計算することを考慮すると、これらの式から t_j を求めることは困難である。 t_j を求めることができないと、評価区間上の状態 $x^*(\cdot; t)$ 及び評価区間上の随伴変数 $\lambda^*(\cdot; t)$ を求めることもできない。さらに、 t_j も新たな未知量として追加されている。したがって、仮に t_j が求められたとしても、本質的な未知量を算出するための式として上記の式(24)をそのまま用いることはできず、新たに未知量及び未知方程式を定義しなければならない。

10

【0071】

そこで、本実施の形態にかかるアルゴリズムでは、十分大きな正のスカラー値 p を用いた以下の式(51)で表されるペナルティ関数が、評価関数に追加されている。

【数 2 8】

$$P(x(\tau_j -)) = p \xi^2(x(\tau_j -)) \quad \cdot \cdot \cdot (51)$$

20

【0072】

式(51)で表されるペナルティ関数を加えた評価関数を最適化すれば、指定した時刻 t_j に対して、以下の式(52)が成り立ち得る。

【数 2 9】

$$\xi(x(\tau_j -; t)) = 0 \quad \cdot \cdot \cdot (52)$$

【0073】

つまり、上記のペナルティ関数を用いることにより、指定した時刻 t_j において状態ジャンプを生じさせることが可能となる。言い換えると、上記のペナルティ関数は、指定したタイミングで状態が不連続となるようにシステムの状態を拘束する拘束パラメータである。また、式(52)で表される拘束条件がペナルティ関数について追加されているため、ラグランジュ乗数 λ は追加されなくてもよい。したがって、本実施の形態にかかるアルゴリズムでは、非線形モデル予測制御の最適化問題を、状態ジャンプの無い非線形モデル予測制御と同数の未知数の問題として扱うことができる。

30

【0074】

したがって、状態ジャンプを含む非線形モデル予測制御の問題は、式(51)を評価関数に加えることにより、状態ジャンプの無い非線形モデル予測制御と同様に扱うことができる。但し、ペナルティ関数を用いるため、あらかじめ適切な状態ジャンプ時刻 t_j を指定する必要がある。

40

【0075】

このように、本実施の形態にかかるアルゴリズムでは、ペナルティ関数を用いることにより、想定しているタイミングで、不連続な状態変化を起こさせることができる。したがって、元々の非線形モデル予測制御の理論を容易に適用でき、さらに C / G M R E S 法を適用することも可能となる。したがって、後述するように、二足歩行ロボットのような非線形システムに対しても実時間で制御を行うことが可能となる。

【0076】

次に、ペナルティ関数を用いた場合の停留条件について説明する。ここで、仮に、連続

50

時間で停留条件を導出した場合、以下の式(53), (54)で表される項が生じる。

【数30】

$$\frac{\partial H_1}{\partial u}(x^*(\tau;t), u^*(\tau;t), \lambda^*(\tau;t)) = 0 \quad (\tau < \tau_j) \quad \dots (53)$$

$$\frac{\partial H_2}{\partial u}(x^*(\tau;t), u^*(\tau;t), \lambda^*(\tau;t)) = 0 \quad (\tau > \tau_j) \quad \dots (54)$$

【0077】

10

上述したように、コンピュータで数値計算を行うためには、停留条件の各方程式を離散近似して考える必要がある。しかしながら、式(53), (54)については、 i_j の前後で離散近似を行うことができない。したがって、本実施の形態にかかる問題に対しては、状態方程式及び評価関数を離散近似したのち、変分法より停留条件を導出することとする。

【0078】

まず、以下の式(55)を満たすステップ*i*をジャンプステップ*i_j*と定義する。

【数31】

$$i\Delta\tau \leq \tau_j < (i+1)\Delta\tau \quad \dots (55)$$

20

【0079】

i_j を用いると、上記の式(33), (34), (32)は、それぞれ、以下の式(56), (57), (58)で示すように、離散近似される。

【数32】

$$x_{i+1}(t) = x_i(t) + f_1(x_i(t), u_i(t))\Delta\tau \quad (i < i_j) \quad \dots (56)$$

$$x_{i+1}(t) = x_i(t) + f_2(x_i(t), u_i(t))\Delta\tau \quad (i_j < i) \quad \dots (57)$$

30

$$x_{i_j+1}(t) = \psi(x_{i_j}(t)) \quad \dots (58)$$

【0080】

また、式(51)で表されるペナルティ項(ペナルティ関数)は、以下の式(59)で示すように、離散近似される。

【数33】

$$P(x_{i_j}(t)) = p\xi^2(x_{i_j}(t)) \quad \dots (59)$$

【0081】

40

次に、変分法より停留条件を求める。離散近似された本システムに対する最適制御問題とは、式(59)が追加された、以下の式(60)で表されるような離散近似された評価関数を最小にする入力系列 $u_0(t), \dots, u_{N-1}(t)$ を求める問題である。

【数34】

$$J = \varphi(x(N)) + \sum_{i=0}^{N-1} L(x_i(t), u_i(t))\Delta\tau + P(x_{i_j}(t)) \quad \dots (60)$$

【0082】

ここで、式(56), (57), (58)は、それぞれ、以下の式(61), (62)

50

, (63) で示されるような等式拘束条件とみなすことができる。

【数35】

$$f_1(x_i(t), u_i(t))\Delta\tau - (x_{i+1}(t) - x_i(t)) = 0 \quad (i < i_j) \quad \cdot \cdot \cdot (61)$$

$$f_2(x_i(t), u_i(t))\Delta\tau - (x_{i+1}(t) - x_i(t)) = 0 \quad (i_j < i) \quad \cdot \cdot \cdot (62)$$

$$\psi(x_{i_j}(t)) - x_{i_j+1}(t) = 0 \quad \cdot \cdot \cdot (63)$$

【0083】

式(61), (62) のラグランジュ乗数ベクトルを $\lambda_{i+1}(t)$ 、式(63) のラグランジュ乗数ベクトルを

【数36】

$$\lambda_{i_j+1}(t)$$

とする。このとき、ラグランジュ関数は、以下の式(64) で定義される。

【0084】

【数37】

$$\begin{aligned} \bar{J} = J &+ \sum_{i=i_j+1}^{N-1} \lambda_{i+1}^T(t) \{f_2(x_i(t), u_i(t))\Delta\tau - (x_{i+1}(t) - x_i(t))\} \\ &+ \lambda_{i_j+1}^T(t) \{\psi(x_{i_j}(t)) - x_{i_j+1}(t)\} + P(x_{i_j}(t)) \\ &+ \sum_{i=0}^{i_j-1} \lambda_{i+1}^T(t) \{f_1(x_i(t), u_i(t))\Delta\tau - (x_{i+1}(t) - x_i(t))\} \\ &\cdot \cdot \cdot (64) \end{aligned}$$

【0085】

式(64) の J に式(60) で示された J を代入すると、以下の式(65) が得られる。

【数 3 8】

$$\begin{aligned}
\bar{J} = & \varphi(x_N(t)) \\
& + \sum_{i=i_j+1}^{N-1} \left\{ H_2(x_i(t), u_i(t), \lambda_{i+1}(t)) \Delta \tau + (\lambda_{i+1}^T(t) - \lambda_i^T(t)) x_i(t) \right\} \\
& + L(x_{i_j}(t), u_{i_j}(t)) \Delta \tau + \lambda_{i_j+1}(t) \psi(x_{i_j}(t)) P(x_{i_j}(t)) - \lambda_{i_j}^T(t) x_{i_j}(t) \\
& + \sum_{i=0}^{i_j-1} \left\{ H_1(x_i(t), u_i(t), \lambda_{i+1}(t)) \Delta \tau + (\lambda_{i+1}^T(t) - \lambda_i^T(t)) x_i(t) \right\} \\
& + \lambda_0(t) x_0(t) \\
& \cdot \cdot \cdot (65)
\end{aligned}$$

【0086】

但し、ハミルトン関数 H_1 , H_2 は、それぞれ以下の式 (66) , (67) で定義される。

【数 3 9】

$$H_1(x_i(t), u_i(t), \lambda_{i+1}(t)) = L(x_i(t), u_i(t)) + \lambda_{i+1}^T(t) f_1(x_i(t), u_i(t)) \quad \cdot \cdot \cdot (66)$$

$$H_2(x_i(t), u_i(t), \lambda_{i+1}(t)) = L(x_i(t), u_i(t)) + \lambda_{i+1}^T(t) f_2(x_i(t), u_i(t)) \quad \cdot \cdot \cdot (67)$$

【0087】

また、式 (65) の変分は、以下の式 (68) のように表される。

【数 4 0】

$$\begin{aligned}
\delta \bar{J} = & \left(\frac{\partial \varphi}{\partial x}(x_N(t)) - \lambda_N^T \right) \delta x_N(t) \\
& + \sum_{i=i_j+1}^{N-1} \left\{ \left(\frac{\partial H_2}{\partial x} \Delta \tau - \lambda_{i+1}^T(t) + \lambda_i^T(t) \right) \delta x_i(t) + \frac{\partial H_2}{\partial u} \Delta \tau \delta u_i(t) \right\} \\
& + \left\{ \frac{\partial L}{\partial x}(x_{i_j}(t), u_{i_j}(t)) \Delta \tau + \frac{\partial P}{\partial x}(x_{i_j}(t)) + \lambda_{i_j+1}^T(t) \frac{\partial \psi}{\partial x}(x_{i_j}(t)) \right. \\
& \quad \left. - \lambda_{i_j}^T(t) \right\} \delta x_{i_j}(t) + \frac{\partial L}{\partial u}(x_{i_j}(t), u_{i_j}(t)) \Delta \tau \delta u_{i_j}(t) \\
& + \sum_{i=0}^{i_j-1} \left\{ \left(\frac{\partial H_1}{\partial x} \Delta \tau - \lambda_{i+1}^T(t) + \lambda_i^T(t) \right) \delta x_i(t) + \frac{\partial H_1}{\partial u} \Delta \tau \delta u_i(t) \right\} \\
& + \lambda_0(t) \delta x_0(t) \\
& \cdot \cdot \cdot (68)
\end{aligned}$$

【0088】

制御入力値が最適であれば、式(68)において、任意の $x_i(t)$, $u_i(t)$ について、

【数41】

$$\delta \bar{J} = 0$$

が成り立つ。ここで、 $x_0(t) = x(t)$ より $x_0(t) = 0$ であることに注意すると、以下の式(69)~(79)で表される停留条件が導かれる。但し、添字*は、評価区間上の値であることを示す。

【0089】

【数 4 2】

$$x_{i+1}^*(t) = x_i^*(t) + f_1(x_i^*(t), u_i^*(t))\Delta\tau \quad (i < i_j - 1) \quad \cdot \cdot \cdot (69)$$

$$x_{i+1}^*(t) = x_i^*(t) + f_2(x_i^*(t), u_i^*(t))\Delta\tau \quad (i_j + 1 < i) \quad \cdot \cdot \cdot (70)$$

$$x_{i_j+1}^*(t) = \psi(x_{i_j}^*(t)) \quad (i = i_j) \quad \cdot \cdot \cdot (71)$$

$$x_0^*(t) = x(t) \quad \cdot \cdot \cdot (72)$$

$$\lambda_i^*(t) = \lambda_{i+1}^*(t) - \left(\frac{\partial H_1}{\partial x} \right)^T (x_i^*(t), u_i^*(t), \lambda_{i+1}^*(t))\Delta\tau \quad (i < i_j - 1) \quad \cdot \cdot \cdot (73)$$

$$\lambda_i^*(t) = \lambda_{i+1}^*(t) - \left(\frac{\partial H_2}{\partial x} \right)^T (x_i^*(t), u_i^*(t), \lambda_{i+1}^*(t))\Delta\tau \quad (i_j + 1 < i) \quad \cdot \cdot \cdot (74)$$

$$\lambda_{i_j}^* = \left(\frac{\partial \psi}{\partial x} \right)^T (x_{i_j}^*(t))\lambda_{i_j+1}^*(t) + \left(\frac{\partial P}{\partial x} \right)^T (x_{i_j}^*(t)) + \left(\frac{\partial L}{\partial x} \right)^T (x_{i_j}^*(t), u_{i_j}^*(t))\Delta\tau \quad \cdot \cdot \cdot (75)$$

$$\lambda_N^*(t) = \left(\frac{\partial \phi}{\partial x} \right)^T (x_N^*(t)) \quad \cdot \cdot \cdot (76)$$

$$\left(\frac{\partial H_1}{\partial u} \right)^T (x_i^*(t), u_i^*(t), \lambda_{i+1}^*(t)) = 0 \quad (i < i_j - 1) \quad \cdot \cdot \cdot (77)$$

$$\left(\frac{\partial H_2}{\partial u} \right)^T (x_i^*(t), u_i^*(t), \lambda_{i+1}^*(t)) = 0 \quad (i_j + 1 < i) \quad \cdot \cdot \cdot (78)$$

$$\frac{\partial L}{\partial u} (x_{i_j}^*(t), u_{i_j}^*(t)) = 0 \quad \cdot \cdot \cdot (79)$$

【0090】

ここで、評価区間上の状態 $x_{i_j}^*(t)$ は、上記式(69)、(70)、(71)、(72)から算出され得る。また、随伴変数 $\lambda_{i_j}^*(t)$ は、上記式(73)、(74)、(75)、(76)から算出され得る。したがって、本実施の形態にかかるモデルでは、拘束条件を考えていないため、未知量を示すベクトル $U(t)$ は、以下の式(80)で定義される。

10

20

30

40

【数 4 3】

$$U(t) := \begin{bmatrix} u_0^*(t) \\ \vdots \\ u_{N-1}^*(t) \end{bmatrix} \quad \cdot \cdot \cdot (80)$$

【0091】

この $U(t)$ は、以下の式 (81) で示される方程式を解くことによって得られる。

10

【数 4 4】

$$F(U(t), x(t))$$

$$:= \begin{bmatrix} \left(\frac{\partial H_1}{\partial u} \right)^T (x_0^*(t), u_0^*(t), \lambda_1^*(t)) \\ \vdots \\ \left(\frac{\partial H_1}{\partial u} \right)^T (x_{i_j-1}^*(t), u_{i_j-1}^*(t), \lambda_{i_j}^*(t)) \\ \left(\frac{\partial L}{\partial u} \right)^T (x_{i_j}^*(t), u_{i_j}^*(t)) \\ \left(\frac{\partial H_2}{\partial u} \right)^T (x_{i_j+1}^*(t), u_{i_j+1}^*(t), \lambda_{i_j+2}^*(t)) \\ \vdots \\ \left(\frac{\partial H_2}{\partial u} \right)^T (x_{N-1}^*(t), u_{N-1}^*(t), \lambda_N^*(t)) \end{bmatrix} = 0 \quad \cdot \cdot \cdot (81)$$

20

30

40

この方程式を、上述した C / G M R E S 法を用いて解けばよい。

【0092】

ここで、サンプリング周期ごとの入力系列 (最適解) の更新について考える。図 4 は、入力系列の更新について説明するための図である。図 4 の矢印 A に示すように、C / G M R E S 法では、上記式 (26) で示すように前進差分近似を行っている。ここで、時刻 t における評価区間での i_j ステップ目の時刻つまり $(t + i_j)$ が状態ジャンプ前の時刻 (つまり時刻 t_j より前の時刻) であり、時刻 $t + t$ における評価区間での i_j ステップ目の時刻つまり $(t + t + i_j)$ が状態ジャンプ後の時刻 (つまり時刻 t_j より後の時刻) であるとする。このとき、時刻 t におけるジャンプステップは i_j である

50

が、時刻 $t + \tau$ におけるジャンプステップは $i_j - 1$ である。このとき、
【数 4 5】

$$u_{i_j}^*(t) \cdots (U_{j_b})$$

は状態ジャンプ前の入力（以下、式 U_{j_b} と表記）であり、
【数 4 6】

$$u_{i_j}^*(t + \Delta t) \cdots (U_{j_a})$$

は状態ジャンプ後の入力（以下、式 U_{j_a} と表記）となる。
【0093】

したがって、本実施の形態にかかるシステムは、状態ジャンプの前後で状態が不連続であることから、 $i = i_j$ のとき、単純に、

【数 4 7】

$$u_{i_j}^*(t + \Delta t) = u_{i_j}^*(t) + \dot{u}_{i_j}^*(t)\Delta t \cdots (26)'$$

とすることはできない。つまり、式 (26)' のように計算すると、上記の状態ジャンプ後の入力 U_{j_a} は、最適入力として更新されていないこととなる。

【0094】

そこで、本実施の形態においては、状態ジャンプ後の入力 U_{j_a} を、 $i = i_j$ の $u_{i_j}^*$ より近似することと考える。状態ジャンプ後の入力 U_{j_a} は、時刻 $t + \tau + i_j$ の評価区間上の入力である。適切に近似するためには、図 4 の矢印 B で示すように、この時刻に最も近い時刻における入力、つまり時刻 $t + (i_j + 1)\tau$ の入力

【数 4 8】

$$u_{i_j+1}^*(t)$$

から近似すればよい。

【0095】

したがって、時刻 t におけるジャンプステップと時刻 $t + \tau$ におけるジャンプステップとが異なるとき、状態ジャンプ後の入力 U_{j_a} は、以下の式 (82) で示すように更新される。

【数 4 9】

$$u_{i_j}^*(t + \Delta t) = u_{i_j+1}^*(t) + \dot{u}_{i_j+1}^*(t)(\Delta t - \Delta\tau) \cdots (82)$$

【0096】

但し、 $i_j = N - 1$ のときは、上記式 (82) で示した近似は行えないので、通常通り、以下の式 (83) で示すように入力は更新される。

【数 5 0】

$$u_{N-1}^*(t + \Delta t) = u_{N-1}^*(t) + \dot{u}_{N-1}^*(t)\Delta t \cdots (83)$$

【0097】

なお、上式は、 t と $t + \tau$ とが、以下の式 (84) を満たすことを仮定している。

$$t < 2\tau \cdots (84)$$

しかしながら、

$$2\tau < t < 3\tau \cdots (85)$$

10

20

30

40

50

であるときも、 $u^*_i(t + \Delta t)$ の系列は、実時間上の $u^*_i(t)$ の系列から適切に近似され得る。

【0098】

なお、C/GMRES法においては、 $U(t)$ の変化量を求める際に、 $F(U, x, t)$ を用いて、以下の式(86)で示すように、前進差分近似を行っている。

【数51】

$$\frac{\partial F}{\partial U}(U, x, t)W + \frac{\partial F}{\partial x}w(U, x, t) + \frac{\partial F}{\partial t}(U, x, t)w$$

$$\approx \frac{F(U + hW, x + hw, t + hw) - F(U, x, t)}{h} \quad \dots (86)$$

【0099】

しかしながら、差分近似の差分時間を h とすると、時刻 t と時刻 $t + h$ との間で状態ジャンプが生じた場合、状態量が不連続に大きく変化してしまい、差分近似を正確に行うことができない。したがって、時刻 t が以下の式(87)で示される

$$t < t_j \quad t + h \quad \dots (87)$$

を満たすとき、以下の式(88)で示すように後退差分近似を行う。

【数52】

$$\frac{\partial F}{\partial U}(U, x, t)W + \frac{\partial F}{\partial x}w(U, x, t) + \frac{\partial F}{\partial t}(U, x, t)w$$

$$\approx \frac{F(U, x, t) - F(U - hW, x - hw, t - hw)}{h} \quad \dots (88)$$

【0100】

(二足歩行ロボットへの適用)

次に、上述した非線形モデル予測制御を、本実施の形態にかかるロボット100の動作の制御に適用した例について説明する。なお、実施の形態1においては、ロボット100がコンパス型モデルである例について説明するが、後述するように、非線形モデル予測制御は、ロボット100がコンパス型モデルでなくても適用可能である。

【0101】

なお、ロボット100の歩行動作は、遊脚が地面と衝突する(着地する)という動作を含む。この衝突の前後で、ロボット100の一般化速度が不連続に変化する。つまり、このとき、状態ジャンプが発生する。また、一般的に、歩行動作は、周期的な運動である。したがって、ロボット100を、予め定められた周期ごとに状態ジャンプを生じさせる(つまり遊脚を着地させる)ように制御を行うことが可能である。なお、「着地」とは、遊脚が地面と衝突(接触)することに限定されない。つまり、「着地」とは、ロボット100がその上を歩行している面(歩行面)に遊脚が接触することを意味する。

【0102】

図5は、実施の形態1にかかるロボット100をコンパス型モデルに適用する方法を説明するための図である。図5に示した例では、右脚110Rが支持脚であり、左脚110Lが遊脚(振り脚)である。制御装置2は、支持脚が地面と点接触していることを模擬するため、支持脚(図5の例では右脚110R)の足首関節部124に設けられたトルクセンサ136を用いて、支持脚の足首関節部124のトルクを0に制御する。また、制御装置2は、右脚110R及び左脚110Lの膝関節部122を、伸展状態でロックするように制御する。つまり、制御装置2は、右脚110R及び左脚110Lの膝関節部122の関節角度が伸展状態に対応する角度(例えば0)となるように、膝関節部122のモータ

140を制御する。さらに、制御装置2は、遊脚(図5の例では左脚110L)の足裏センサ118を用いて、遊脚の着地を検出する。このようにして、ロボットシステム1は、コンパス型モデルを模擬することができる。

【0103】

図6は、実施の形態1にかかるロボット100をコンパス型モデルに適用した例を示す図である。図6に示す例では、ロボット100は、関節150と、支持脚リンク151と、遊脚リンク152とから構成されるコンパス型モデルにモデル化されている。ここで、関節150は、胴体102及び股関節部120に対応する。また、支持脚リンク151は、右脚110R及び左脚110Lのうちの支持脚に対応する。また、遊脚リンク152は、右脚110R及び左脚110Lのうちの遊脚に対応する。

10

【0104】

関節150の質量を m_0 とする。また、図6の矢印で示すように、関節150の周りに、制御入力値として入力トルク u が入力される。ここで、支持脚リンク151及び遊脚リンク152の物理的性質は、互いに同じであるとする。支持脚リンク151及び遊脚リンク152の長さを、 l とする。また、支持脚リンク151及び遊脚リンク152の質量を、 m とする。

【0105】

また、鉛直方向に対する支持脚リンク151の角度を θ_1 とし、鉛直方向に対する遊脚リンク152の角度を θ_2 とする。但し、図6において時計回り(各リンクの下端を中心に関節150が前方に回る方向)を正とする。したがって、図6の状態では、 $\theta_2 < 0$ である。

20

【0106】

次に、図6に例示したコンパス型モデルの歩行動作に関して、以下のような仮定があるとする。

- ・遊脚リンク152と地面90との衝突(着地)は一瞬である。
- ・遊脚リンク152と地面90との衝突は完全非弾性衝突である。
- ・リンクと地面との摩擦係数は μ である。
- ・両脚(両リンク)が同時に地面90から力を受けることはない。

【0107】

上記の仮定より、両脚(支持脚リンク151及び遊脚リンク152)が同時に地面90に着くことはない。また、衝突時に遊脚リンク152の速度は0となる。したがって、本モデルの歩行制御に必要な方程式は、片脚支持期の運動方程式(状態方程式)と、遊脚リンク152の衝突時の方程式(衝突方程式)との2つである。

30

【0108】

片脚(つまり支持脚リンク151)だけが地面90に接触しているとき、ラグランジュの運動方程式より、以下の式(89)で示す方程式が導き出される。

【数53】

$$M(q)\ddot{q} + H(q, \dot{q}) = Nu \quad \cdot \cdot \cdot (89)$$

$$M(q) \in \mathcal{R}^{2 \times 2}, H(q, \dot{q}) \in \mathcal{R}^2, Nu \in \mathcal{R}^2$$

40

【0109】

但し、 q は、以下の式(S1)で示される一般化座標ベクトルである。

【数54】

$$q = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \quad \cdot \cdot \cdot (S1)$$

50

また、 $M(q)$ は慣性行列、 $H(q, \dot{q})$ は重力とコリオリ力の項、 Nu は q に対する一般化力である。なお、 $R^{n \times m}$ は、 $n \times m$ の実数行列全体の集合を示す。

【0110】

ここで、式(89)に示した運動方程式の詳細を、以下の式(E1)、(E2)、(E3)に示す。なお、以下の式において、 I^m は、支持脚リンク151及び遊脚リンク152の重心(重心151m及び重心152m)周りの慣性モーメントである。また、 l_G は、関節150から各リンクの重心(重心151m及び重心152m)までの長さである。

【数55】

$$M = \begin{bmatrix} (m_0 + 2m)l^2 - 2mll_G + ml_G^2 + I^m & 0 \\ -mll_G \cos(\theta_1 - \theta_2) & ml_G^2 + I^m \end{bmatrix} \cdot \cdot \cdot (E1) \quad 10$$

$$H = \begin{bmatrix} -mll_G \dot{\theta}_2^2 \sin(\theta_1 - \theta_2) - ((m_0 + 2m) - ml_G)g \sin(\theta_1) \\ ml_G (l \dot{\theta}_1^2 \sin(\theta_1 - \theta_2) + g \sin(\theta_2)) \end{bmatrix} \cdot \cdot \cdot (E2)$$

$$Nu = \begin{bmatrix} u \\ -u \end{bmatrix} \cdot \cdot \cdot (E3)$$

【0111】

また、状態ベクトル $x \in R^4$ を、以下の式(S2)に示す。

【数56】

$$x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \cdot \cdot \cdot (S2) \quad 20$$

この場合、状態方程式は、上記式(89)より、以下の式(90)で表される。

【数57】

$$\dot{x} = f(x, u) = \begin{bmatrix} \dot{q} \\ M(q)^{-1} (Nu - H(q, \dot{q})) \end{bmatrix} \cdot \cdot \cdot (90) \quad 30$$

なお、本実施の形態かかるコンパス型モデルでは、歩行の拘束条件として、ZMP (zero moment point) は考慮されないものとする。

【0112】

次に、衝突方程式について説明する。衝突方程式の説明の前に、状態ジャンプについて説明する。

図7は、状態ジャンプを説明するための図である。図7は、右脚110R又は左脚110Lの状態を示す図である。ここでは、右脚110Rの状態を示すとする。図7は、横軸が右脚110Rの角度を示し、縦軸が右脚110Rの角速度を示す、グラフ(位相線図)である。

【0113】

状態 I において、右脚 1 1 0 R が地面から離れて遊脚となる。したがって、状態 I から、右脚 1 1 0 R (遊脚リンク 1 5 2) の角度及び角速度は、 $(\theta_2, \dot{\theta}_2)$ (ドット) である。このとき、状態 I から後述する状態 I I までの期間では、角度及び角速度は、連続的に変化している。そして、状態 I I で、遊脚であった右脚 1 1 0 R (遊脚リンク 1 5 2) が地面 9 0 に着地する。そして、直ちに、状態は状態 I I I に移行して右脚 1 1 0 R は支持脚 (支持脚リンク 1 5 1) となる。このとき、状態 I I から状態 I I I に遷移するときに、角度はほとんど変わらないが、角速度が急激に変化する。したがって、状態 I I から状態 I I I に遷移する際に、状態ジャンプが発生している。

【 0 1 1 4 】

状態 I I I から、右脚 1 1 0 R (支持脚リンク 1 5 1) の角度及び角速度は、 $(\theta_1, \dot{\theta}_1)$ (ドット) である。このとき、状態 I I I から後述する状態 I V までの期間では、角度及び角速度は、連続的に変化している。そして、状態 I V で、遊脚であった左脚 1 1 0 L (遊脚リンク 1 5 2) が地面 9 0 に着地する。そして、直ちに、状態は状態 I に移行して右脚 1 1 0 R は遊脚 (遊脚リンク 1 5 2) となる。このとき、状態 I V から状態 I に遷移するときに、角度はほとんど変わらないが、角速度が急激に変化する。したがって、状態 I V から状態 I に遷移する際に、状態ジャンプが発生している。

このように、遊脚が地面に着地すると、状態を示すパラメータ (図 7 の例では脚の角度及び角速度) が、不連続に変化する。この現象が、状態ジャンプである。

【 0 1 1 5 】

次に、衝突前後の一般化座標及び一般化速度の定義について説明する。

図 8 は、遊脚リンク 1 5 2 の衝突直前のロボット 1 0 0 の状態を示す図である。図 8 に示すように、衝突直前の一般化座標及び一般化速度は、それぞれ以下の式 (G 1) 及び式 (G 2) で表される。

【数 5 8】

$$q^- = \begin{bmatrix} \theta_1^- \\ \theta_2^- \end{bmatrix} \quad \cdot \cdot \cdot (G1)$$

$$\dot{q}^- = \begin{bmatrix} \dot{\theta}_1^- \\ \dot{\theta}_2^- \end{bmatrix} \quad \cdot \cdot \cdot (G2)$$

【 0 1 1 6 】

図 9 は、遊脚リンク 1 5 2 の衝突直後のロボット 1 0 0 の状態を示す図である。図 9 に示すように、衝突直後の一般化座標及び一般化速度は、それぞれ以下の式 (G 3) 及び式 (G 4) で表される。

【数 5 9】

$$q^+ = \begin{bmatrix} \theta_1^+ \\ \theta_2^+ \end{bmatrix} \quad \cdot \cdot \cdot (G3)$$

$$\dot{q}^+ = \begin{bmatrix} \dot{\theta}_1^+ \\ \dot{\theta}_2^+ \end{bmatrix} \quad \cdot \cdot \cdot (G4)$$

【 0 1 1 7 】

ここで、遊脚リンク 1 5 2 と地面 9 0 との衝突について、以下の 2 つの角運動量についての保存則を適用する。

- ・衝突前に遊脚であった脚（遊脚リンク 1 5 2）の先端まわりの系全体の角運動量。
- ・関節 1 5 0 まわりの、衝突前に支持脚であった脚（支持脚リンク 1 5 1）の角運動量。

上記の角運動量保存則は、上記式（G 1）,（G 2）,（G 3）,（G 4）より、以下の式（9 1）で表される。

【数 6 0】

$$Q^+(q^+)\dot{q}^+ = Q^-(q^-)\dot{q}^- \quad \cdot \cdot \cdot (91)$$

10

$$Q^+, Q^- \in \mathcal{R}^{2 \times 2}$$

【0 1 1 8】

なお、 Q^+ 及び Q^- は、それぞれ以下の式（E 4）,（E 5）で表される。

【数 6 1】

$$Q^- = \begin{bmatrix} (m_0 l^2 + 2ml(l-l_G))\cos(\theta_1 - \theta_2) - m(l-l_G)l_G + I^m & -m(l-l_G)l_G + I^m \\ -m(l-l_G)l_G + I^m & 0 \end{bmatrix} \quad \cdot \cdot \cdot (E4)$$

20

$$Q^+ = \begin{bmatrix} (m_0 l^2 + m(l-l_G)^2 + ml(l-l_G))\cos(\theta_1 - \theta_2) & ml_G(l_G - l\cos(\theta_1 - \theta_2)) + I^m \\ -m(l-l_G)l_G + I^m & ml_G^2 + I^m \\ -mll_G \cos(\theta_1 - \theta_2) & \end{bmatrix} \quad \cdot \cdot \cdot (E5)$$

【0 1 1 9】

ここで、衝突前後で、一般化座標は変わらない。したがって、衝突直前の一般化座標 q^- 及び衝突直後の一般化座標 q^+ について、以下の式（9 2）が成り立つ。

30

【数 6 2】

$$q^+ = q^- \quad \cdot \cdot \cdot (92)$$

【0 1 2 0】

また、上記式（9 1）,（9 2）は、衝突直前の状態空間ベクトル x^- 及び衝突直後の状態空間ベクトル x^+ を用いて、以下の式（9 3）で表される。

【数 6 3】

$$x^+ = \begin{bmatrix} I_2 & 0 \\ 0 & Z(q^-) \end{bmatrix} x^- \quad \cdot \cdot \cdot (93)$$

40

但し、 I_2 は 2×2 の単位行列である。

【0 1 2 1】

また、 $Z(q^-)$ は、以下の式（9 4）を満たす 2×2 行列である。

【数 6 4】

$$Z(q^-) = (Q^+(q^-))^{-1} Q^-(q^-) \quad \cdot \cdot \cdot (94)$$

50

【 0 1 2 2 】

次に、実施の形態 1 にかかるコンパス型モデルに、上述した非線形モデル予測制御を適用することを考える。上述したように、歩行動作とは、「連続して脚を前に出す」という動作である。本モデルにおいては、脚（支持脚リンク 1 5 1 及び遊脚リンク 1 5 2）の開き角を目標値に近づける、という評価関数を設定し、遊脚リンク 1 5 2 が着地するたびに各リンクの座標を入れ替えることで歩行制御を行う。

【 0 1 2 3 】

このとき、上述した評価関数 J の終端コスト 及びステージコスト L は、それぞれ、次の式 (9 5) , (9 6) のように表される。

【 数 6 5 】

$$\varphi = \frac{1}{2} s_f (\theta_1 - \theta_2 - \theta_{ref})^2 \quad \cdot \cdot \cdot (95)$$

$$L = \frac{1}{2} q_0 (\theta_1 - \theta_2 - \theta_{ref})^2 + \frac{1}{2} r u^2 \quad \cdot \cdot \cdot (96)$$

但し、 s_f 、 q_0 及び r は、それぞれ重みを表す正のスカラ値である。また、 θ_{ref} は、脚の開き角の目標値（目標角度）である。

【 0 1 2 4 】

また、遊脚の着地の度に θ_1 と θ_2 とを入れ替え、 $\dot{\theta}_1$ （ドット）と $\dot{\theta}_2$ （ドット）とを入れ替えることを考慮すると、状態ジャンプの方程式である式 (9 3) は、以下の式 (9 7) のように書き換えられる。

【 数 6 6 】

$$\begin{aligned} x(t_j+) &= \psi(x(t_j-)) \\ &= \begin{bmatrix} I'_2 & 0 \\ 0 & I'_2 \end{bmatrix} \begin{bmatrix} I_2 & 0 \\ 0 & Z(q(t_j-)) \end{bmatrix} x(t_j-) \\ &= \begin{bmatrix} I'_2 & 0 \\ 0 & I'_2 Z(q(t_j-)) \end{bmatrix} x(t_j-) \quad \cdot \cdot \cdot (97) \end{aligned}$$

【 0 1 2 5 】

但し、 $I'_2 \in \mathbb{R}^{2 \times 2}$ は、以下の行列である。

【 数 6 7 】

$$I'_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

【 0 1 2 6 】

次に、歩行周期を T_{step} とすると、ジャンプ時刻 t_j は、整数 k を用いて以下の式 (9 8) で表される。

$$t_j = k T_{step} \quad \cdot \cdot \cdot (98)$$

なお、評価区間中に状態ジャンプが 2 回以上生じないように、式 (3) で示した $T(t)$

10

20

30

40

50

)を設定する。このとき、周期 T_{step} ごとに式(91)で表される脚(遊脚リンク152)と地面90との衝突が起こり、それ以外ときは、式(97)で表される運動方程式でモデルの状態を記述することができる。つまり、モデルの状態は、以下の式(99)、(100)で表される。

【数68】

$$\dot{x}(t) = f(x(t), u(t)) \quad (t \neq t_j) \quad \dots (99)$$

$$x(t_j+) = \psi(x(t_j-)) \quad (t = t_j) \quad \dots (100)$$

10

【0127】

ここで、時刻 t_j で状態ジャンプ、つまり脚(遊脚リンク152)と地面90との衝突が起こらなければならない。また、支持脚リンク151及び遊脚リンク152の長さが互いに同じであることから、遊脚リンク152が地面90に着地したとき、 $\theta_1 = \theta_2$ である。したがって、状態ジャンプが起こる条件は、以下の式(101)で表される。

【数69】

$$\xi(x(t_j-)) = \theta_1(t_j-) + \theta_2(t_j-) = 0 \quad \dots (101)$$

【0128】

したがって、評価関数に追加されるペナルティ関数(ペナルティ項)は、以下の式(51)'で表される。ここで、 p_1 は、ゲイン(重み)であって、十分大きな正のスカラー値である。このゲイン p_1 は、制御装置2のコンピュータの性能等に応じて、適宜調整可能である。

【数70】

$$p_1 \xi^2(x(\tau_j-)) = p_1 (\theta_1(\tau_j-) + \theta_2(\tau_j-))^2 \quad \dots (51)'$$

【0129】

さらに、上記のペナルティ項とは別に、遊脚リンク152の着地時の脚の開き角を目標値に近づけるような項を付け加えることを考える。したがって、以下の式(51)''で示す項も、ペナルティ項として評価関数に加える。ここで、 p_2 は、ゲイン(重み)であって、十分大きな正のスカラー値である。このゲイン p_2 は、制御装置2のコンピュータの性能等に応じて、適宜調整可能である。

【数71】

$$p_2 (\theta_1(\tau_j-) - \theta_2(\tau_j-) - \theta_{ref})^2 \quad \dots (51)''$$

【0130】

式(51)'及び式(51)''を足し合わせると、実施の形態1にかかるペナルティ項は、以下の式(102)で表される。これにより、ペナルティ項(拘束パラメータ)は、予め指定したタイミング(時刻 t_j)において遊脚が着地するようにロボット100の状態を指定することとなる。言い換えると、ペナルティ項(拘束パラメータ)は、予め指定したタイミング(時刻 t_j)において遊脚が着地したときのロボット100の姿勢(各関節部の関節角度)を指定することとなる。

【数72】

$$P(x(\tau_j-)) = p_1 (\theta_1(\tau_j-) + \theta_2(\tau_j-))^2 + p_2 (\theta_1(\tau_j-) - \theta_2(\tau_j-) - \theta_{ref})^2 \quad \dots (102)$$

50

したがって、実施の形態 1 にかかるコンパス型モデルにおける非線形モデル予測制御を用いたロボット 100 の制御では、式 (51) で示したペナルティ関数として、上記式 (102) で表したものが適用される。

【0131】

図 10 は、実施の形態 1 にかかる制御装置 2 によって行われるロボット 100 の制御方法を示すフローチャートである。図 10 に示した制御方法は、上述した非線形モデル予測制御を用いている。したがって、実施の形態 1 にかかるロボット 100 の制御方法は、式 (69) ~ (79) で表される停留条件から、式 (80) で表されるベクトル $U(t)$ を求め、このベクトル $U(t)$ の各成分の値を、式 (81) で示される方程式を解くことによって算出する。これにより、ロボット 100 の動作を制御するための制御入力値が算出される。ここで、図 10 に示したフローチャートにおいて、S102 は、状態パラメータを取得する取得ステップに対応し、S104 ~ S114 は、制御入力値を算出する算出ステップに対応し、S116 は、ロボット 100 の動作を制御する制御ステップに対応する。

10

【0132】

まず、制御装置 2 は、ロボット 100 の状態を示す状態ベクトル (状態パラメータ) を取得して、状態観測を行う (ステップ S102)。具体的には、制御装置 2 の状態取得部 12 は、股関節部 120R 及び股関節部 120L の角度センサ 130 から、股関節部 120R 及び股関節部 120L の関節角度を取得する。そして、状態取得部 12 は、これらの関節角度から、現在時刻 t における θ_1 及び θ_2 (図 6) を算出する。なお、状態取得部 12 は、例えば、支持脚である方の脚にかかる股関節部 120 の関節角度と、胴体 102 の傾きとから、鉛直方向に対する支持脚リンク 151 の角度 θ_1 を取得できる。同様に、状態取得部 12 は、例えば、遊脚である方の脚にかかる股関節部 120 の関節角度と、胴体 102 の傾きとから、鉛直方向に対する遊脚リンク 152 の角度 θ_2 を取得できる。なお、胴体 102 の傾きは、例えばジャイロセンサ等の傾斜センサを用いて取得可能である。

20

【0133】

また、状態取得部 12 は、 θ_1 及び θ_2 の変化量 $\Delta\theta_1$ (ドット) 及び $\Delta\theta_2$ (ドット) を算出する。例えば、状態取得部 12 は、時刻 t の 1 つ前のサンプリング周期 (制御周期) における時刻 $t - \Delta t$ における θ_1 及び θ_2 と時刻 t における θ_1 及び θ_2 との差分から、それぞれ変化量 $\Delta\theta_1$ (ドット) 及び $\Delta\theta_2$ (ドット) を算出してもよい。

30

【0134】

これにより、状態取得部 12 は、時刻 t における状態ベクトル $x(t)$ を取得する。さらに、状態取得部 12 は、式 (72) で示すように、この $x(t)$ を、時刻 t についての評価区間における、状態ベクトルの初期値とする。つまり、 $x^*_0(t) = x(t)$ とする。なお、この式 (72) についての処理は、非線形モデル予測制御部 14 が行ってもよい。

【0135】

次に、制御装置 2 の非線形モデル予測制御部 14 は、時刻 t についての評価区間における状態変数を、各 $i = 1 \sim N$ について更新する (ステップ S104)。具体的には、非線形モデル予測制御部 14 は、式 (69) ~ (71) で示したように、状態変数 $x^*_i(t)$ を更新する。なお、実施の形態 1 にかかるコンパス型モデルの遊脚の着地の例では、状態方程式に関する f は、ジャンプステップの前後で同じであるとする。つまり、 $f_1(x, u) = f_2(x, u)$ である。

40

【0136】

そして、非線形モデル予測制御部 14 は、式 (90) に示した状態方程式を用いて、式 (69) ~ (70) で示すように、 $i = i_j$ における状態変数を更新する。また、非線形モデル予測制御部 14 は、 $i = i_j$ においては、式 (97) に示した状態ジャンプの方程式を用いて、式 (71) で示すように、状態変数を更新する。また、制御装置 2 の RAM 8 は、得られた状態変数 $x^*_i(t)$ を記憶する。

50

【0137】

次に、制御装置2は、時刻tについての評価区間における随伴変数を、各 $i = 1 \sim N$ について更新する(ステップS106)。具体的には、非線形モデル予測制御部14は、式(73)~(76)で示したように、S104で更新された各 x^*_i を用いて、随伴変数 $^*_i(t)$ を更新する。また、制御装置2のRAM8は、得られた随伴変数 $^*_i(t)$ を記憶する。

【0138】

なお、上述したように、 $f_1(x, u) = f_2(x, u)$ であるので、 $H_1 = H_2$ である。そして、非線形モデル予測制御部14は、式(90)に示した状態方程式及び式(96)に示したステージコストLの式を用いてハミルトン関数Hを算出し、式(73)~(74)で示すように、 $i = i_j$ における随伴変数を更新する。また、非線形モデル予測制御部14は、 $i = i_j$ においては、式(97)に示した状態ジャンプの方程式、式(102)に示したペナルティ関数の式及び式(96)に示したステージコストLの式を用いて、式(75)で示すように、随伴変数を更新する。また、非線形モデル予測制御部14は、 $i = N$ においては、式(95)に示した終端コストの式を用いて、式(76)で示すように、随伴変数を更新する。

10

【0139】

次に、制御装置2は、ベクトル関数Fを導出する(ステップS108)。具体的には、非線形モデル予測制御部14は、式(80)で示されたベクトル $U(t)$ を算出するため、S104及びS106の処理で得られた状態変数 $x^*_i(t)$ 及び随伴変数 $^*_i(t)$ を用いて、式(81)で示されたベクトル関数 $F(U, x)$ の方程式を導出する。なお、上述したように、実施の形態1においては、 $H_1 = H_2$ である。

20

【0140】

次に、制御装置2は、ベクトル $U(t)$ の全微分を計算する(ステップS110)。具体的には、非線形モデル予測制御部14は、式(29)を変形した以下の式(103)から、C/GMRES法を用いて、 U の全微分($U(\text{ドット})$)、つまり $U(t)$ の変化率を算出する。言い換えると、非線形モデル予測制御部14は、制御周期ごとに、制御入力値の最適解の変化率を算出する。

【数73】

$$\dot{U} = -\left(\frac{\partial F}{\partial U}\right)^{-1} \left(\zeta F + \frac{\partial F}{\partial x} \dot{x} + \frac{\partial F}{\partial t} \right) \quad \dots (103)$$

30

これにより、式(80)から明らかのように、 $i = 0 \sim N - 1$ について、 $u^*_i(t)$ の時間微分($u^*_i(\text{ドット})$)が得られることとなる。RAM8は、得られた $u^*_i(t)$ の時間微分の値を記憶する。

【0141】

次に、制御装置2は、入力系列 $u^*_i(t)$ の更新を行う(ステップS112)。具体的には、非線形モデル予測制御部14は、式(26)及び式(82)から、以下の式(104)及び(105)により、入力系列 $u^*_i(t)$ の更新を行う。

40

$$u^*_i(t + \Delta t) = u^*_i(t) + \dot{u}^*_i(t) \Delta t \quad (i \neq i_j) \quad \dots (104)$$

$$u^*_{i_j}(t + \Delta t) = u^*_{i_j+1}(t) + \dot{u}^*_{i_j+1}(t) (\Delta t - \Delta \tau) \quad (i = i_j) \quad \dots (105)$$

【0142】

ここで、式(104)、(105)から、非線形モデル予測制御部14は、ある時刻tにおける入力系列 $u^*_i(t)$ 及び $u^*_{i_j}(t)$ の変化率を用いて、次のサンプリング周期 t である時刻 $t + t$ における入力系列を算出する。したがって、時刻tにおける入

50

力系列 $u^*_i(t)$ を算出するためには、現在の時刻 t の 1 つ前のサンプリング周期 $t - \Delta t$ の時刻 $t - \Delta t$ における入力系列 $u^*_i(t - \Delta t)$ 及び $u^*_i(t - \Delta t)$ の時間微分を用いることとなる。これにより、式 (80) で示した $U(t)$ の各成分の値が得られる。言い換えると、入力系列 $u^*_i(t)$ の $i = 0 \sim N - 1$ それぞれの値が得られる。RAM 8 は、入力系列 $u^*_i(t)$ の $i = 0 \sim N - 1$ それぞれの値を記憶する。

【0143】

次に、制御装置 2 は、入力値 (制御入力値) を決定する (ステップ S114)。具体的には、非線形モデル予測制御部 14 は、以下の式 (106) により、入力値 u_0 を決定する。

【数75】

$$u_0 = u_0^*(t) = [1 \quad 0 \quad \dots \quad 0]U(t) \quad \cdot \cdot \cdot (106)$$

つまり、非線形モデル予測制御部 14 は、 $U(t)$ の成分のうちの 1 番目の成分を入力値と決定する。非線形モデル予測制御部 14 は、決定された入力値を、サーボ制御部 16 に対して出力する。

【0144】

次に、制御装置 2 は、ロボット 100 の制御を行う (ステップ S116)。具体的には、サーボ制御部 16 は、S114 で決定された入力値から、各関節部に指示する関節トルクを決定する。さらに具体的には、サーボ制御部 16 は、支持脚リンク 151 に対応する脚の股関節部 120 の関節トルク τ_1 と、遊脚リンク 152 に対応する脚の股関節部 120 の関節トルク τ_2 とを、以下の式 (107)、(108) によって決定する。

$$\tau_1 = u_0 \quad \cdot \cdot \cdot (107)$$

$$\tau_2 = -u_0 \quad \cdot \cdot \cdot (108)$$

【0145】

なお、胴体 102 (関節 150) の姿勢が崩れることを防止するため、サーボ制御部 16 は、以下の式 (107)、(108) のように関節トルク τ_1 及び関節トルク τ_2 を決定してもよい。

【数76】

$$\tau_1 = u + \tau_{fb} \quad \cdot \cdot \cdot (107)$$

$$\tau_2 = -u - \tau_{fb} \quad \cdot \cdot \cdot (108)$$

$$\text{但し、} \tau_{fb} = k_p \theta_{torso} + k_d \dot{\theta}_{torso}$$

ここで、 θ_{torso} は、鉛直方向に対する胴体 102 の前方への傾き角度である。また、 k_p 及び k_d は、ゲイン (重み) であって、予め定められた定数である。このゲインは、制御装置 2 のコンピュータの性能に応じて、適宜調整可能である。

サーボ制御部 16 は、決定された関節トルクとなるように、各関節部 (股関節部 120) のモータ 140 を制御する。

【0146】

以上のように、実施の形態 1 にかかる制御装置 2 は、非線形モデル予測制御のアルゴリズムを用いてロボット 100 の動作を制御するに際し、指定したタイミングで遊脚が着地するようにロボット 100 の状態を拘束するペナルティ関数 (拘束パラメータ) を用いている。これにより、想定しているタイミングで、遊脚の着地という不連続な状態変化を起こさせることができる。したがって、元々の非線形モデル予測制御の理論を容易に適用でき、さらに C/GMRES 法を適用することも可能となる。したがって、二足歩行ロボットのような非線形システムに対しても実時間で制御を行うことが可能となる。

【0147】

10

20

30

40

50

また、実施の形態 1 にかかるペナルティ関数は、上記式 (1 0 2) で示すように、予め指定したタイミングにおいて遊脚が着地したときのロボット 1 0 0 の姿勢を指定している。これにより、想定した姿勢で遊脚を着地させるようにロボット 1 0 0 を制御することが可能となる。さらに、実施の形態 1 にかかるペナルティ関数は、予め指定したタイミングにおいて遊脚が着地したときのロボット 1 0 0 の各関節部の関節角度を指定している。これにより、想定した関節角度で遊脚を着地させるようにロボット 1 0 0 の関節部を制御することが可能となる。

【 0 1 4 8 】

また、式 (1 0 2) で示すように、ペナルティ関数は、調整可能なゲイン (p_1 及び p_2) を含む。指定されたタイミングで状態変化を確実に起こさせるため、ゲインは十分な大きなスカラー値とする必要がある。しかしながら、制御装置 2 のコンピュータの性能等によっては、ゲインがあまりにも大きすぎると感度が過大となるため、不安定な制御となる可能性がある。したがって、ゲインを調整することにより、制御を安定化させることが可能となる。

【 0 1 4 9 】

(実施の形態 2)

次に、実施の形態 2 について説明する。実施の形態 2 では、ロボット 1 0 0 がより人間に似たヒューマノイドロボットである点で、実施の形態 1 と異なる。そして、実施の形態 2 では、膝を曲げて二足歩行を行うモデル (膝屈曲モデル) について、上述した非線形モデル予測制御を適用する。その他の点については、実施の形態 1 と実質的に同様であるので、説明を省略する。

【 0 1 5 0 】

図 1 1 は、実施の形態 2 にかかるロボット 1 0 0 を示す図である。実施の形態 1 にかかるロボット 1 0 0 と同様に、実施の形態 2 にかかるロボット 1 0 0 は、胴体 1 0 2 と、右脚 1 1 0 R と、左脚 1 1 0 L とを有する。右脚 1 1 0 R 及び左脚 1 1 0 L の構成については、実施の形態 1 のものと同様である。また、胴体 1 0 2 は、腰部 1 0 2 a と、胸部 1 0 2 b と、腰関節部 1 0 2 c とを有する。また、胴体 1 0 2 の上側には、頭部 1 0 4 が設けられている。この頭部 1 0 4 に、カメラ等のセンサが設けられていてもよい。

【 0 1 5 1 】

また、ロボット 1 0 0 は、胴体 1 0 2 (胸部 1 0 2 b) の右側及び左側に、それぞれ右腕 1 6 0 R 及び左腕 1 6 0 L を有する。実施の形態 2 にかかるロボット 1 0 0 は、右腕 1 6 0 R 及び左腕 1 6 0 L を用いて、所定の動作を行うことが可能である。なお、図 1 1 には図示されていないが、右腕 1 6 0 R 及び左腕 1 6 0 L の先端に、対象物を把持することが可能なエンドエフェクタが設けられていてもよい。

【 0 1 5 2 】

右腕 1 6 0 R は、胴体 1 0 2 に近い方から順に、肩関節部 1 7 0 R と、上腕部 1 6 2 R と、肘関節部 1 7 2 R と、前腕部 1 6 4 R とを有する。同様に、左腕 1 6 0 L は、胴体 1 0 2 に近い方から順に、肩関節部 1 7 0 L と、上腕部 1 6 2 L と、肘関節部 1 7 2 L と、前腕部 1 6 4 L とを有する。肩関節部 1 7 0 R 及び肩関節部 1 7 0 L は、胴体 1 0 2 の右側及び左側にそれぞれ取り付けられている。そして、肩関節部 1 7 0 R 及び肩関節部 1 7 0 L を介して、それぞれ、上腕部 1 6 2 R 及び上腕部 1 6 2 L が胴体 1 0 2 と接続されている。言い換えると、右腕 1 6 0 R 及び左腕 1 6 0 L は、それぞれ、肩関節部 1 7 0 R 及び肩関節部 1 7 0 L を介して、胴体 1 0 2 と接続されている。

【 0 1 5 3 】

また、肘関節部 1 7 2 R を介して、上腕部 1 6 2 R と前腕部 1 6 4 R とが接続されている。同様に、肘関節部 1 7 2 L を介して、上腕部 1 6 2 L と前腕部 1 6 4 L とが接続されている。また、肩関節部 1 7 0 は、互いに直交した 3 軸の周りをそれぞれ回転するように構成され得る。また、肘関節部 1 7 2 は、1 軸の周りを回転する。また、図 2 に示したように、肩関節部 1 7 0 及び肘関節部 1 7 2 は、角度センサ 1 3 0 と、モータ 1 4 0 とを有する。

10

20

30

40

50

【 0 1 5 4 】

図 1 2 は、実施の形態 2 にかかるロボット 1 0 0 を膝屈曲モデルに適用した状態を示す図である。図 1 2 に示した例では、右脚 1 1 0 R が支持脚であり、左脚 1 1 0 L が遊脚である。制御装置 2 は、支持脚が地面と点接触していることを模擬するため、支持脚（図 1 2 の例では右脚 1 1 0 R）の足首関節部 1 2 4 に設けられたトルクセンサ 1 3 6 を用いて、足首関節部 1 2 4 のトルクを 0 に制御する。さらに、制御装置 2 は、遊脚（図 1 2 の例では左脚 1 1 0 L）の足裏センサ 1 1 8 を用いて、遊脚の着地を検出する。ここで、実施の形態 1 とは異なり、実施の形態 2 では、膝関節部 1 2 2 は、ロックされていない。このようにして、ロボットシステム 1 は、膝屈曲モデルを模擬することができる。

【 0 1 5 5 】

図 1 3 は、実施の形態 2 にかかるロボット 1 0 0 を膝屈曲モデルに適用した例を示す図である。図 1 3 に示す例では、ロボット 1 0 0 は、支持脚下腿リンク 2 0 1 と、支持脚上腿リンク 2 0 2 と、胴体リンク 2 0 3 と、遊脚上腿リンク 2 0 4 と、遊脚下腿リンク 2 0 5 と、関節 2 1 1, 2 1 2, 2 1 3, 2 1 4, 2 1 5 とから構成される、膝屈曲モデルにモデル化されている。ここで、胴体リンク 2 0 3 は、ロボット 1 0 0 の胴体 1 0 2 から上の構成要素に対応する。

【 0 1 5 6 】

また、支持脚下腿リンク 2 0 1 は、右脚 1 1 0 R 及び左脚 1 1 0 L のうちの支持脚にかかる下腿部 1 1 4 に対応する。支持脚上腿リンク 2 0 2 は、右脚 1 1 0 R 及び左脚 1 1 0 L のうちの支持脚にかかる上腿部 1 1 2 に対応する。遊脚上腿リンク 2 0 4 は、右脚 1 1 0 R 及び左脚 1 1 0 L のうちの遊脚にかかる上腿部 1 1 2 に対応する。遊脚下腿リンク 2 0 5 は、右脚 1 1 0 R 及び左脚 1 1 0 L のうちの遊脚にかかる下腿部 1 1 4 に対応する。

【 0 1 5 7 】

また、関節 2 1 1 は、支持脚にかかる足首関節部 1 2 4 に対応する。関節 2 1 2 は、支持脚にかかる膝関節部 1 2 2 に対応する。関節 2 1 3 は、股関節部 1 2 0 に対応する。関節 2 1 4 は、遊脚にかかる膝関節部 1 2 2 に対応する。関節 2 1 5 は、遊脚にかかる足首関節部 1 2 4 に対応する。ここで、関節 2 1 1 は、地面 9 0 に接触している。この関節 2 1 1 の位置、つまり支持脚の先端の位置を、 (X_b, Y_b) とする。また、関節 2 1 5 の位置、つまり遊脚の先端の位置を、 (X_c, Y_c) とする。

【 0 1 5 8 】

ここで、各リンク及び各関節 2 1 1 ~ 2 1 5 を区別するパラメータ k ($k = 1 \sim 5$) を設ける。 $k = 1$ は、支持脚下腿リンク 2 0 1 及び関節 2 1 1 に対応する。同様に、 $k = 2$ は、支持脚上腿リンク 2 0 2 及び関節 2 1 2 に対応する。 $k = 3$ は、胴体リンク 2 0 3 及び関節 2 1 3 に対応する。 $k = 4$ は、遊脚上腿リンク 2 0 4 及び関節 2 1 4 に対応する。そして、 $k = 5$ は、遊脚下腿リンク 2 0 5 及び関節 2 1 5 に対応する。

【 0 1 5 9 】

したがって、支持脚下腿リンク 2 0 1、支持脚上腿リンク 2 0 2、胴体リンク 2 0 3、遊脚上腿リンク 2 0 4、及び遊脚下腿リンク 2 0 5 を、それぞれ、リンク # 1、# 2、# 3、# 4、# 5 と示すことがある。そして、各リンクについて一般化して示すときに、リンク # k と示すことがある。関節 2 1 1 ~ 2 1 5 についても同様に、それぞれ、関節 # 1、# 2、# 3、# 4、# 5 と示すことがある。そして、各関節について一般化して示すときに、関節 # k と示すことがある。

【 0 1 6 0 】

支持脚下腿リンク 2 0 1、支持脚上腿リンク 2 0 2、胴体リンク 2 0 3、遊脚上腿リンク 2 0 4、及び遊脚下腿リンク 2 0 5 の長さ（リンク長）を、それぞれ、 l_1 、 l_2 、 l_3 、 l_4 、 l_5 とする。また、支持脚下腿リンク 2 0 1、支持脚上腿リンク 2 0 2、胴体リンク 2 0 3、遊脚上腿リンク 2 0 4、及び遊脚下腿リンク 2 0 5 の質量（リンク質量）を、それぞれ、 m_1 、 m_2 、 m_3 、 m_4 、 m_5 とする。また、支持脚下腿リンク 2 0 1、支持脚上腿リンク 2 0 2、胴体リンク 2 0 3、遊脚上腿リンク 2 0 4、及び遊脚下腿リンク 2 0 5 の各リンクの重心周りの慣性モーメントを、それぞれ、 I^m_1 、 I^m_2 、 I^m_3

10

20

30

40

50

、 I^m_4 、 I^m_5 とする。なお、 l_k 、 m_k 及び I^m_k は、予め定められた値である。

【0161】

また、支持脚下腿リンク201、支持脚上腿リンク202、胴体リンク203、遊脚上腿リンク204、及び遊脚下腿リンク205の鉛直方向に対する角度（リンク角度）を、それぞれ、 θ_1 、 θ_2 、 θ_3 、 θ_4 、 θ_5 とする。なお、リンク角度 θ_1 、 θ_2 、 θ_3 、 θ_4 、 θ_5 は、ロボット100の各関節部（股関節部120、膝関節部122及び足首関節部124）の角度センサ130で検出された関節角度から、幾何学的に一意に算出可能である。したがって、リンク角度 θ_1 、 θ_2 、 θ_3 、 θ_4 、 θ_5 は、ロボット100の各関節部の関節角度に対応する。

【0162】

また、支持脚下腿リンク201の重心201mの位置を、 (X_{c1}, Y_{c1}) とする。支持脚上腿リンク202の重心202mの位置を、 (X_{c2}, Y_{c2}) とする。胴体リンク203の重心203mの位置を、 (X_{c3}, Y_{c3}) とする。遊脚上腿リンク204の重心204mの位置を、 (X_{c4}, Y_{c4}) とする。遊脚下腿リンク205の重心205mの位置を、 (X_{c5}, Y_{c5}) とする。また、 $k = 1 \sim 5$ それぞれについて、関節#kから、リンク#kの重心#k（質点）までの距離を、 r_k とする。この r_k は、予め定められた値である。

【0163】

ここで、重心#kの位置 (X_{ck}, Y_{ck}) は、以下の式(109)、(110)で表される。なお、式(109)、(110)の第2項については、 $k < 4$ の場合は0とする。

【数77】

$$X_{ck} = \sum_{h=1}^{k-1} (bl_h \sin \theta_h) + \sum_{h=4}^k (l_h \sin \theta_h) + ar_k \sin \theta_k + X_b \quad \cdot \cdot \cdot (109)$$

$$Y_{ck} = \sum_{h=1}^{k-1} (bl_h \cos \theta_h) + a \sum_{h=4}^k (l_h \cos \theta_h) + r_k \cos \theta_k + Y_b \quad \cdot \cdot \cdot (110)$$

但し、

$$a = \begin{cases} 1; k < 4 \\ -1; k \geq 4 \end{cases} ; \quad b = \begin{cases} 1; h < 3 \\ 0; h \geq 3 \end{cases}$$

【0164】

式(109)、(110)について時間微分を行うことで、以下の式(111)、(112)で示すように、重心#kの速度が得られる。

10

20

30

40

【数 7 8】

$$\dot{X}_{ck} = \sum_{h=1}^{k-1} (bl_h \dot{\theta}_h \cos \theta_h) + \sum_{h=4}^k (l_h \dot{\theta}_h \cos \theta_h) + ar_k \dot{\theta}_k \cos \theta_k + \dot{X}_b$$

• • • (111)

$$\dot{Y}_{ck} = \sum_{h=1}^{k-1} (bl_h \dot{\theta}_h \sin \theta_h) + a \sum_{h=4}^k (l_h \dot{\theta}_h \sin \theta_h) + r_k \dot{\theta}_k \sin \theta_k + \dot{Y}_b$$

• • • (112) 10

【0 1 6 5】

この場合、ラグランジアン を以下の式 (1 1 3) のように表すことができる。なお、 g は重力加速度である。

【数 7 9】

$$\Gamma = \sum_{k=1}^5 \left(\frac{1}{2} m_k (\dot{X}_{ck}^2 + \dot{Y}_{ck}^2) + \frac{1}{2} I^m_k \dot{\theta}_k^2 - m_k g Y_{ck} \right)$$

• • • (113) 20

【0 1 6 6】

このとき、ラグランジュの運動方程式は、以下の式 (1 1 4) のように表される。

【数 8 0】

$$\frac{d}{dt} \left(\frac{\partial \Gamma}{\partial \dot{q}} \right) - \frac{\partial \Gamma}{\partial q} = N \eta$$

• • • (114) 30

【0 1 6 7】

なお、 q 及び η は、以下に示すベクトルである。

【数 8 1】

$$q = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \end{bmatrix} ; \quad \eta = \begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \eta_4 \\ \eta_5 \end{bmatrix}$$

40

なお、 η は、関節トルクベクトルである。ここで、 η_1 は、足首関節部 1 2 4 のトルクである。また、 η_2 は、支持脚の膝関節部 1 2 2 のトルクである。また、 η_3 は、支持脚の股関節部 1 2 0 のトルク、言い換えると、支持脚に対する胴体リンク 2 0 3 のトルクである。 η_4 は、遊脚の股関節部 1 2 0 のトルクである。また、 η_5 は、遊脚の膝関節部 1 2 2 のトルクである。

【0 1 6 8】

そして、式(114)から、以下の式(115)を導き出すことができる。つまり、実施の形態2にかかる膝屈曲モデルでは、コンパス型モデルにおける式(89)の方程式が、以下の式(115)で示すように修正される。なお、 $M(q)$ 及び $H(q, \dot{q})$ は、膝屈曲モデルに対応するように修正され得る。

【数82】

$$M(q)\ddot{q} + H(q, \dot{q}) = N\eta \quad \cdot \cdot \cdot (115)$$

【0169】

ここで、 N は、以下の式(116)で表される行列である。

10

【数83】

$$N = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \cdot \cdot \cdot (116)$$

20

【0170】

また、実施の形態2にかかる膝屈曲モデルにおいても、式(91)が成り立つので、 Q^+ 及び Q^- についても、膝屈曲モデルに対応するように修正され得る。また、式(95)及び式(96)に示した評価関数 J の終端コスト及びステージコスト L についても、膝屈曲モデルに対応するように修正され得る。

また、実施の形態2にかかる膝屈曲モデルにおけるペナルティ項については、式(102)で示したものを、以下の式(117)で示したものに修正する。なお、 q_{ref} は、遊脚が着地するときの目標姿勢を示す。言い換えると、 q_{ref} は、遊脚が着地するときの各関節の関節角度に対応するリンク角度 q_1, q_2, q_3, q_4, q_5 の目標値(目標角度)のベクトルを示す。具体的には、 q_{ref} は、遊脚が着地した時点において、 $X_c - X_b$ が実現したい歩幅となり、 $Y_c = Y_b$ となる(遊脚の先端の高さが支持脚の先端の高さと同じとなる)ような q である。

30

【数84】

$$P(x(\tau_j^-)) = \left(q(\tau_j^-) - q_{ref} \right)^T R \left(q(\tau_j^-) - q_{ref} \right) \quad \cdot \cdot \cdot (117)$$

【0171】

但し、 R は、以下の式(118)で表されるペナルティ重み行列である。なお、 $p_1 \sim p_5$ は、予め定められた値である。

40

【数 8 5】

$$R = \begin{bmatrix} p_1 & 0 & 0 & 0 & 0 \\ 0 & p_2 & 0 & 0 & 0 \\ 0 & 0 & p_3 & 0 & 0 \\ 0 & 0 & 0 & p_4 & 0 \\ 0 & 0 & 0 & 0 & p_5 \end{bmatrix} \cdot \cdot \cdot (118)$$

10

【 0 1 7 2】

そして、図 1 0 に示したフローチャートにおいて、S 1 0 2 の処理で、状態取得部 1 2 は、股関節部 1 2 0、膝関節部 1 2 2 及び足首関節部 1 2 4 の角度センサ 1 3 0 から、各関節部の関節角度を取得する。そして、状態取得部 1 2 は、これらの関節角度から、現在時刻 t における $\theta_1 \sim \theta_5$ (図 1 3) 及びこれらの変化量を算出して、状態ベクトル $x(t)$ を取得する。そして、非線形モデル予測制御部 1 4 は、修正された各関数を用いて S 1 0 4 ~ S 1 1 4 の計算を行って、関節トルク $\tau_1 \sim \tau_5$ を算出することができる。そして、サーボ制御部 1 6 は、算出された関節トルク $\tau_1 \sim \tau_5$ から、各関節部のモータを制御することができる。

20

【 0 1 7 3】

以上のように、状態方程式及びペナルティ項等を、実施の形態 1 にかかるコンパス型モデルにおけるものから、実施の形態 2 にかかる膝屈曲モデルにおけるものに置き換えることができる。これにより、実施の形態 2 にかかる膝屈曲モデルにかかるロボット 1 0 0 の制御においても、上記のコンパス型モデルの制御で用いた非線形モデル予測制御のアルゴリズムを用いることが可能となる。したがって、実施の形態 2 にかかる膝屈曲モデルについても、実施の形態 1 と同様に、実時間で非線形モデル予測制御を行うことが可能となる。

【 0 1 7 4】

(シミュレーション結果)

次に、本実施の形態にかかる非線形モデル予測制御のアルゴリズムを用いて非線形システムについて行ったシミュレーション結果について説明する。以下に説明するシミュレーションは、本実施の形態にかかる非線形モデル予測制御のアルゴリズムを、実施の形態 1 にかかるコンパス型モデルにかかるロボット 1 0 0 に適用したものである。

30

【 0 1 7 5】

表 1 は、シミュレーションで用いたコンパス型モデルの物理パラメータを示す。また、表 2 は、シミュレーションで用いた非線形モデル予測制御の評価関数の重みを示す。また、脚の開き角の目標値を $\theta_{ref} = 0.32$ [rad] とし、歩行周期を $T_{step} = 0.8$ [s] とする。また、式 (S 2) に示した状態ベクトルの初期値を、 $x(t) = [0.166, 0.165, 0.6, 0.75]^T$ とする。

40

【 0 1 7 6】

【表 1】

表 1: コンパス型モデルの物理パラメータ

パラメータ	値
m_0 [kg]	1.0
m [kg]	1.0
l [m]	1.0
l_G [m]	0.50
I^m [kg · m ²]	0.08333
g [m/s ²]	9.80665

10

【0177】

【表 2】

表 2: 重みパラメータ

パラメータ	値
s_f	1.
q_0	1.0
r	1.0
p_1	1.0×10^3
p_2	1.0×10^2

20

30

【0178】

表 3 は、C / GMRES 法の数値計算に用いる各パラメータを示す。ここで、 h_{dir} は、式 (86) で示した GMRES 法における前進差分近似の差分時間 h である。また、 r_{tol} は、シミュレーション開始時における最適性条件残差の許容値である。また、シミュレーション時間は 10 [s] とした。また、式 (3) に示した評価区間 $T(t)$ において、 $T_f = 0.8$ [s]、 $\epsilon = 1.0$ とした。

40

【0179】

【表 3】

表 3: シミュレーション条件

パラメータ	値
サンプリング周期 Δt [s]	0.001
評価区間の分割数 N	80
ζ	1000
h_{dir}	1.0×10^{-8}
r_{tol}	1.0×10^{-6}

10

【0180】

図 14 ~ 図 19 は、本実施の形態にかかる非線形システムに非線形モデル予測制御のアルゴリズムを適用したシミュレーション結果を示す図である。また、図 20 は、シミュレーション結果において定常状態の制御入力のグラフを示す図である。図 14 は、表 1 ~ 表 3 に示した条件下において、本実施の形態にかかる非線形モデル予測制御のアルゴリズムを、実施の形態 1 にかかるコンパス型モデルにかかるロボット 100 に適用したシミュレーション結果を示す。図 14 ~ 図 17 は、それぞれ、 θ_1 、 θ_2 、 $\dot{\theta}_1$ (ドット) 及び $\dot{\theta}_2$ (ドット) のシミュレーション結果を示す。また、図 18 は、制御入力値 u のシミュレーション結果を示す。また、図 19 は、式 (81) で表されるベクトル F の大きさである $\|F\|$ (エラーノルム) のシミュレーション結果を示す。

20

【0181】

図 20 の $t = 8.0$ [s] の近傍及び $t = 8.8$ [s] の近傍においてグラフが垂直に立っている箇所、状態ジャンプが生じていることが分かる。このように、本シミュレーションでは、定常状態において、周期的な状態ジャンプを生じさせることに成功している。また、図 18 の $u(t)$ のグラフから、 $t = 5$ [s] 以降で、定常的な歩行をシミュレーションしていることが分かる。

30

【0182】

(変形例)

なお、本発明は上記実施の形態に限られたものではなく、趣旨を逸脱しない範囲で適宜変更することが可能である。例えば、ロボット 100 の片方の脚は、股関節部 120、膝関節部 122 及び足首関節部 124 を有するとしたが、このような構成に限られない。ロボット 100 の脚は、3 個よりも少ない数の関節部を有してもよいし、3 個よりも多い数の関節部を有してもよい。この場合、状態ベクトル及び関節トルクベクトル (制御入力値) は、関節部の数に応じて、適宜、変更され得る。そして、状態方程式及びペナルティ関数等の関数も、関節部の数に応じて、適宜、変更され得る。

40

【0183】

また、上述した実施の形態においては、非線形システムが二足歩行ロボットである例について説明したが、本実施の形態にかかる非線形モデル予測制御のアルゴリズムは、二足歩行ロボット以外の非線形システムについても適用可能である。つまり、本実施の形態にかかる非線形システムの制御方法は、以下に例示するような、状態ジャンプを伴う任意の非線形システムに対して、適用可能である。そして、上述したように、非線形モデル予測制御において状態ジャンプが発生するタイミングを指定するようにすればよい。

【0184】

50

例えば、本実施の形態にかかる非線形システムは、図11に示したロボット100の腕（右腕160R又は左腕160L）のようなロボットハンド又はロボットアーム等であってもよい。この例における状態ジャンプは、ロボットハンド又はロボットアームが、周辺環境又は操作対象等の物体を押圧するとき、物体を把持し又は離すとき、球体等の物体を叩く又は打ち返すとき等に、発生し得る。なお、球体等の物体を打ち返す非線形システムの例として、例えば、卓球ロボットがある。

【0185】

また、例えば、本実施の形態にかかる非線形システムは、腕及び脚を同時に床等に着地して移動可能な人型ロボット又は動物型ロボット等であってもよい。この例における状態ジャンプは、人型ロボット又は動物型ロボットが、腕と脚とを同時に、壁、床又はテーブル等に接触して移動するとき、又は、人型ロボット又は動物型ロボットが、梯子又は壁等を登るとき等に、発生し得る。

10

【0186】

また、例えば、本実施の形態にかかる非線形システムは、ドローン等の無人航空機などであってもよい。この例における状態ジャンプは、無人航空機が、操作対象又は検査対象の物体に接触するとき又はその物体から離れるとき、輸送対象又は捕獲対象の物体を把持し又は離すとき等に、発生し得る。

【0187】

また、例えば、本実施の形態にかかる非線形システムは、加工機械の工具等であってもよい。この例における状態ジャンプは、加工機械の工具が、加工対象等の物体に接触し又は離れるとき等に、発生し得る。

20

【0188】

また、例えば、本実施の形態にかかる非線形システムは、自動車のトランスミッション等であってもよい。この例における状態ジャンプは、トランスミッションのクラッチが、接触状態（動力の伝達状態）となったとき又は離間状態（動力の遮断状態）になったとき等に、発生し得る。

【0189】

また、例えば、本実施の形態にかかる非線形システムは、ハイブリッド車の動力源等であってもよい。この例における状態ジャンプは、ハイブリッド車の動力源が、モータとエンジンとの間で切り替わるとき等に、発生し得る。また、例えば、本実施の形態にかかる非線形システムは、電気自動車又はハイブリッド車等のバッテリーであってもよい。この例における状態ジャンプは、バッテリーが充電と放電の間で切り替わるとき等に、発生し得る。

30

【0190】

また、例えば、本実施の形態にかかる非線形システムは、自動車等の自動運転システムであってもよい。この例における状態ジャンプは、自車の車線変更又は合流等によって先行車両や後続車両の有無が変化するとき等に、発生し得る。また、この例における状態ジャンプは、物体との衝突が避けられない場合に衝突後の状況まで含めて可能な範囲で最善の動作を行うように制御するとき等に、発生し得る。

【0191】

また、例えば、本実施の形態にかかる非線形システムは、飛行機等であってもよい。この例における状態ジャンプは、飛行機の離着陸において、接地の前後を含めて運動を最適化するように制御するとき等に、発生し得る。具体的には、所望の経路で着陸しつつ、着陸後すみやかに減速するようにエンジン及び機体を制御するような場合である。

40

【0192】

また、例えば、本実施の形態にかかる非線形システムは、列車等であってもよい。この例における状態ジャンプは、列車の連結において、連結の前後を含めて運動を最適化するように制御するとき等に、発生し得る。具体的には、連結時の衝撃及び駆動モータの負荷を軽減するようにモータを制御するような場合である。

【0193】

50

上述したような任意の非線形システムについて、図10で示したような制御方法が実行され得る。この場合、非線形システムの制御方法は、式(69)~(79)で表される停留条件から、式(80)で表されるベクトル $U(t)$ を求め、このベクトル $U(t)$ の各成分の値を、式(81)で示される方程式を解くことによって算出する。これにより、非線形システムを制御するための制御入力値が算出される。そして、図10に示したフローチャートにおいて、S102は、非線形システムの状態パラメータを取得する取得ステップに対応し、S104~S114は、非線形システムを制御するための制御入力値を算出する算出ステップに対応し、S116は、非線形システムを制御する制御ステップに対応する。そして、算出ステップにおいて、指定されたタイミングにおいて状態が不連続に変化するように非線形システムの状態を拘束する拘束パラメータを用いて、制御入力値が算出される。このとき、非線形システムの制御周期ごとに、モデル予測制御のアルゴリズムにおける予め定められた評価区間における制御入力値の最適解の変化率が算出され、変化率を用いて当該制御周期の次の制御周期における前記制御入力値の最適解を算出され、最適解から、現在の前記制御入力値が算出される。

10

【0194】

上述の例において、プログラムは、様々なタイプの非一時的なコンピュータ可読媒体(non-transitory computer readable medium)を用いて格納され、コンピュータに供給することができる。非一時的なコンピュータ可読媒体は、様々なタイプの実体のある記録媒体(tangible storage medium)を含む。非一時的なコンピュータ可読媒体の例は、磁気記録媒体(例えばフレキシブルディスク、磁気テープ、ハードディスクドライブ)、光磁気記録媒体(例えば光磁気ディスク)、CD-ROM(Read Only Memory)、CD-R、CD-R/W、半導体メモリ(例えば、マスクROM、PROM(Programmable ROM)、EPROM(Erasable PROM)、フラッシュROM、RAM(Random Access Memory))を含む。また、プログラムは、様々なタイプの一時的なコンピュータ可読媒体(transitory computer readable medium)によってコンピュータに供給されてもよい。一時的なコンピュータ可読媒体の例は、電気信号、光信号、及び電磁波を含む。一時的なコンピュータ可読媒体は、電線及び光ファイバ等の有線通信路、又は無線通信路を介して、プログラムをコンピュータに供給できる。

20

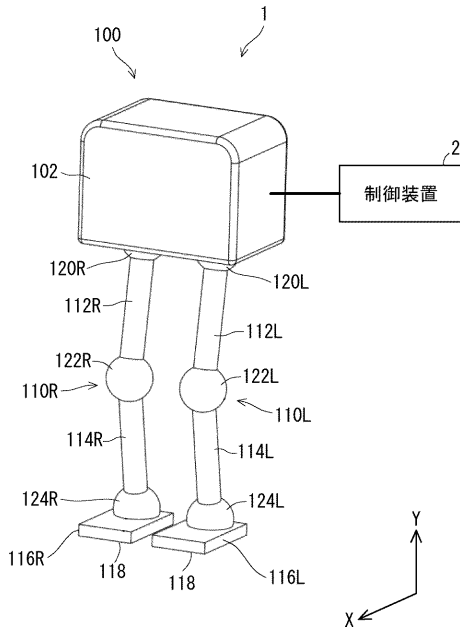
【符号の説明】

【0195】

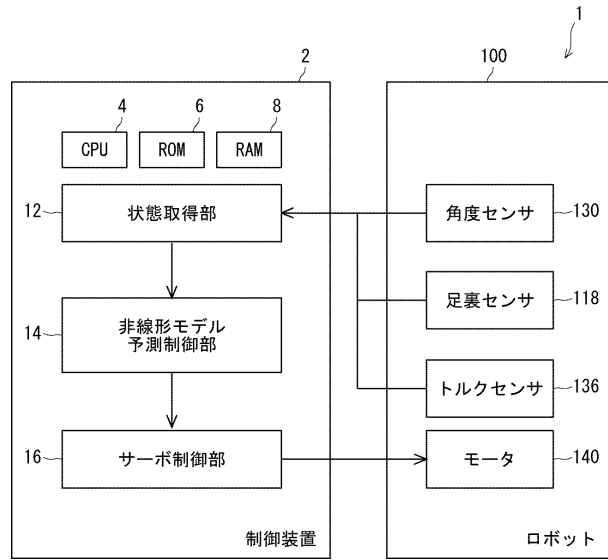
1・・・ロボットシステム、2・・・制御装置、12・・・状態取得部、14・・・非線形モデル予測制御部、16・・・サーボ制御部、100・・・ロボット、102・・・胴体、110L・・・左脚、110R・・・右脚、112・・・上腿部、114・・・下腿部、116・・・足部、118・・・足裏センサ、120・・・股関節部、122・・・膝関節部、124・・・足首関節部、130・・・角度センサ、136・・・トルクセンサ、140・・・モータ、150・・・関節、151・・・支持脚リンク、152・・・遊脚リンク、201・・・支持脚下腿リンク、202・・・支持脚上腿リンク、203・・・胴体リンク、204・・・遊脚上腿リンク、205・・・遊脚下腿リンク、211、212、213、214、215・・・関節

30

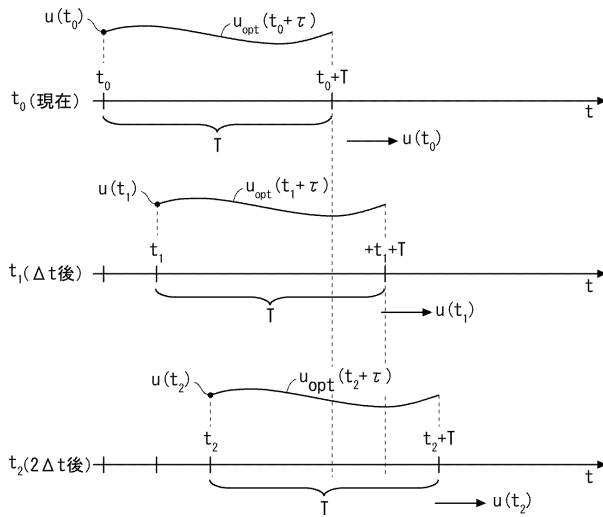
【図1】



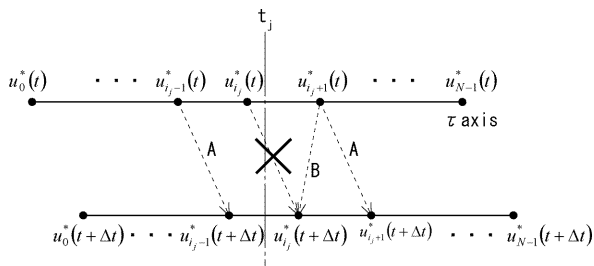
【図2】



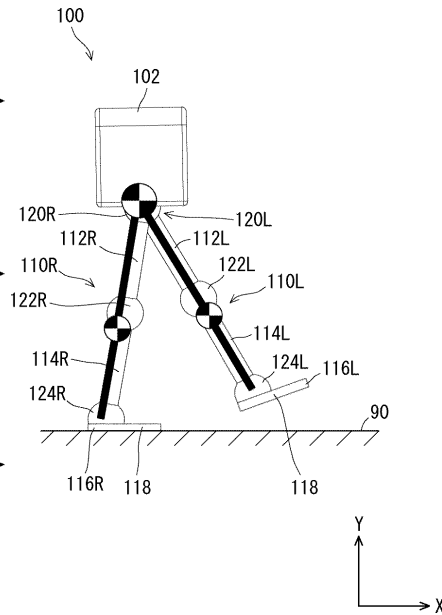
【図3】



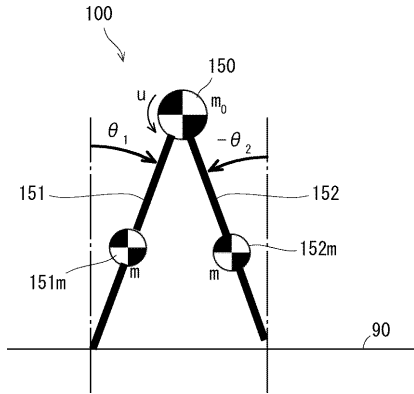
【図4】



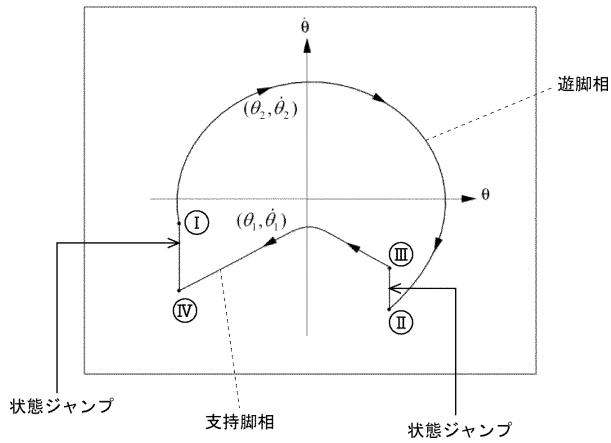
【図5】



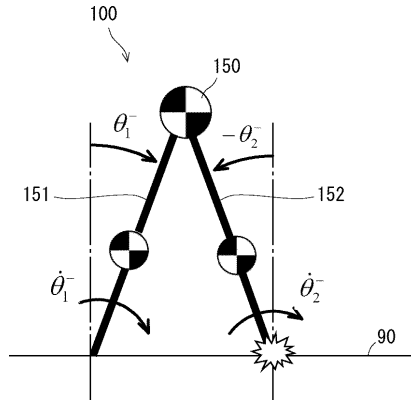
【図6】



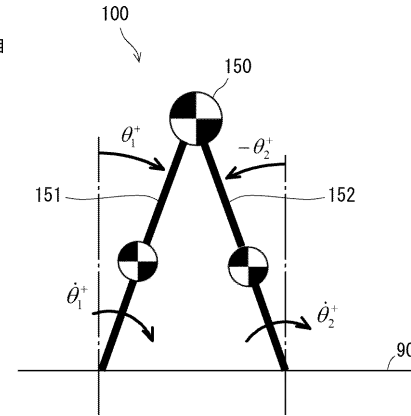
【図7】



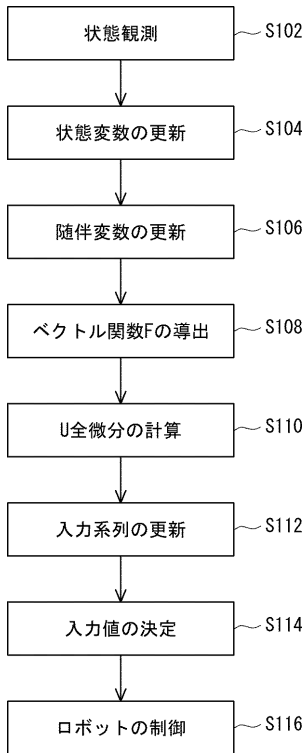
【図8】



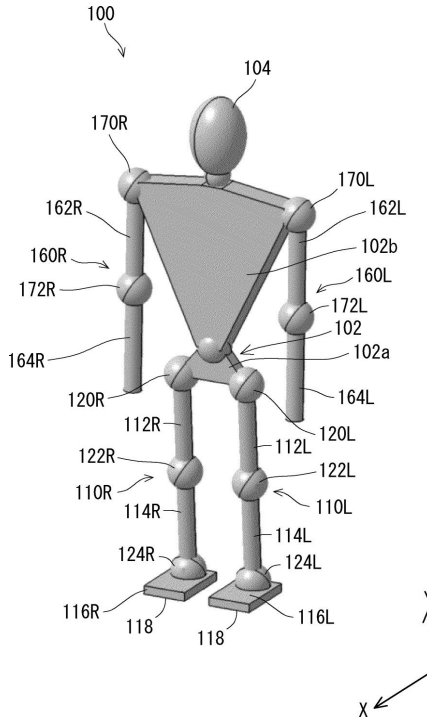
【図9】



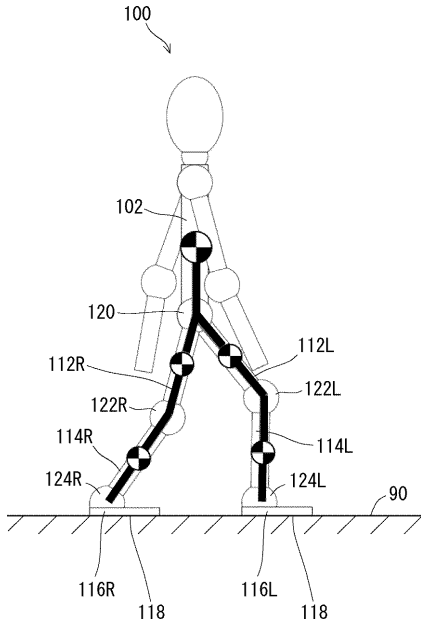
【図10】



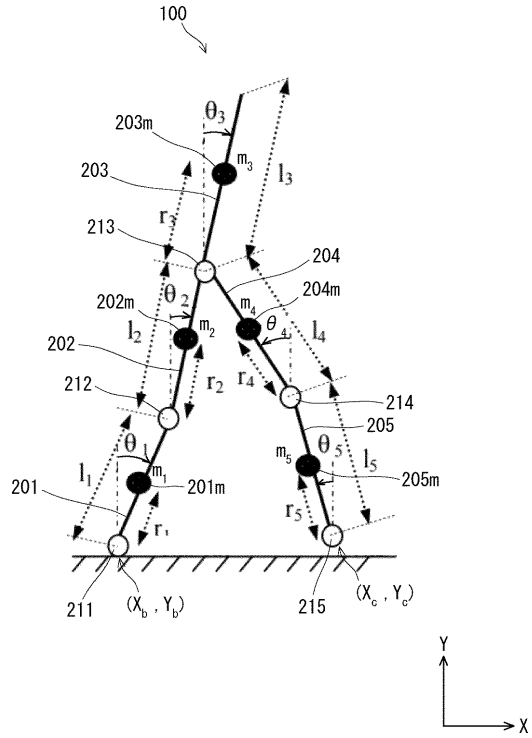
【図11】



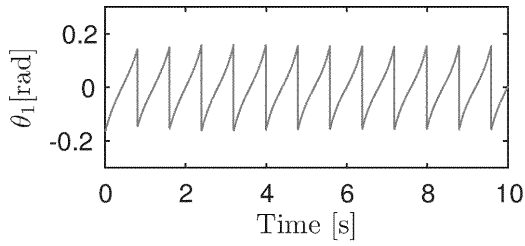
【 図 1 2 】



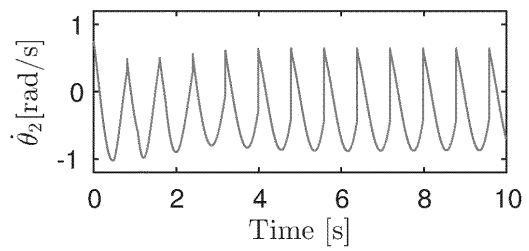
【 図 1 3 】



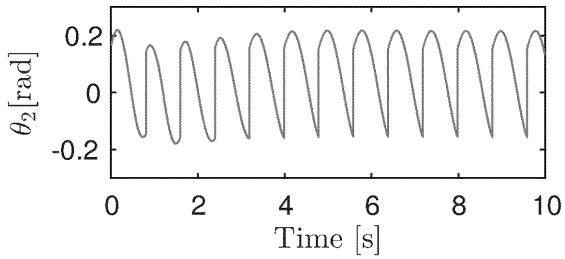
【 図 1 4 】



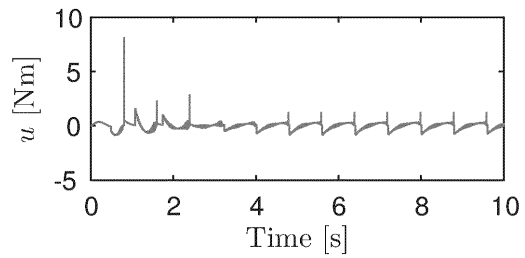
【 図 1 7 】



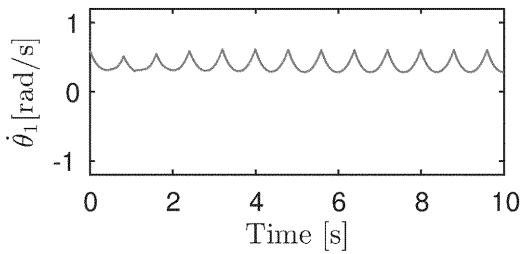
【 図 1 5 】



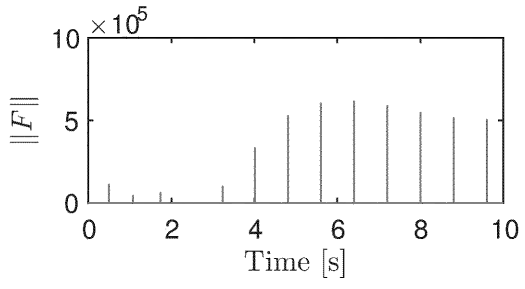
【 図 1 8 】



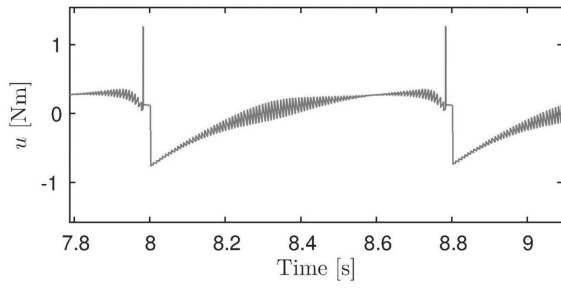
【 図 1 6 】



【図 19】



【図 20】



フロントページの続き

- (72)発明者 佐藤 康之
京都府京都市左京区吉田本町3番地1 国立大学法人京都大学内
- (72)発明者 土井 将弘
愛知県豊田市トヨタ町1番地 トヨタ自動車株式会社内

審査官 中田 善邦

- (56)参考文献 特開2016-022540(JP,A)
特開2007-185756(JP,A)
特開2013-184248(JP,A)
特開2005-219206(JP,A)
特開平05-337849(JP,A)
青木純、大谷哲也、大塚敏之、Continuation/GMRES法による非線形モデル予測制御におけるオフセット補償法、第51回自動制御連合講演会論文集、日本、自動制御連合講演会、2008年1月22日、pp.910-915

(58)調査した分野(Int.Cl., DB名)

G05B 1/00 - 7/04, 11/00 - 13/04,
17/00 - 17/02, 21/00 - 21/02,
B25J 1/00 - 21/02