

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6136494号
(P6136494)

(45) 発行日 平成29年5月31日(2017.5.31)

(24) 登録日 平成29年5月12日(2017.5.12)

(51) Int.Cl.

F 1

G05B 13/04 (2006.01)
F02D 41/18 (2006.01)
F02D 45/00 (2006.01)

GO5B 13/04
F02D 41/18
F02D 45/00

Z
366E

請求項の数 11 (全 37 頁)

(21) 出願番号 特願2013-82605 (P2013-82605)
(22) 出願日 平成25年4月11日 (2013.4.11)
(65) 公開番号 特開2014-206802 (P2014-206802A)
(43) 公開日 平成26年10月30日 (2014.10.30)
審査請求日 平成28年1月13日 (2016.1.13)

(73) 特許権者 000005223
富士通株式会社
神奈川県川崎市中原区上小田中4丁目1番
1号
(74) 代理人 100103528
弁理士 原田 一男
(72) 発明者 梅田 裕平
神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内
審査官 川東 孝至

最終頁に続く

(54) 【発明の名称】情報処理方法、装置及びプログラム

(57) 【特許請求の範囲】

【請求項 1】

(a) モデル予測制御の評価関数とラグランジュ乗数を含み且つ制御対象の状態方程式に関する第1の関数との和を含む第2の関数を前記ラグランジュ乗数について偏微分することで得られ且つ前記制御対象の状態偏差及び前記制御対象に対する線形化制御入力を変数として含む第1の一次式群と、(b) 前記第2の関数を前記制御対象の状態偏差について偏微分することで得られ且つ前記制御対象の状態偏差及び前記ラグランジュ乗数を変数として含む第2の一次式群と、(c) 前記第2の関数を前記制御対象の線形化制御入力について偏微分することで得られ且つ前記ラグランジュ乗数及び前記制御対象の状態偏差を変数として含む第3の一次式群とを格納するデータ格納部から前記第1の一次式群を読み出して、前記状態偏差の現在値を除き前記状態偏差の変数を消し込むように順次代入を行って、前記第1の一次式群を、前記状態偏差の現在値と前記線形化制御入力を変数として含む第4の一次式群に変形し、

前記データ格納部から前記第2の一次式群を読み出して、前記ラグランジュ乗数の変数を消し込むように順次代入を行って、前記第2の一次式群を、前記状態偏差を変数として含む、前記ラグランジュ乗数についての第5の一次式群に変形し、

前記データ格納部から前記第3の一次式群を読み出して、前記第4の一次式群及び前記第5の一次式群を用いて、前記第3の一次式群を、前記状態偏差の現在値と前記線形化制御入力を変数として含む第6の一次式群に変形し、

前記第6の一次式群を、第1の係数行列と前記線形化制御入力についてのベクトルとの

積が、前記状態偏差の現在値の関数ベクトルと等式で表されるように変形し、

前記モデル予測制御の制御対象における状態制約に対応するペナルティ関数を前記状態偏差について1階偏微分した第1の微分関数及び2階偏微分した第2の微分関数を生成し、

前記第4の一次式群から前記線形化制御入力についての係数を抽出し、前記ペナルティ関数について予め定められている第2の係数行列及び係数ベクトルを算出する

処理を、コンピュータに実行させるためのプログラム。

【請求項2】

前記ペナルティ関数が、前記状態偏差が第1の値以上第2の値以下で0となり、前記状態偏差が前記第1の値より小さい場合及び前記第2の値より大きい場合前記状態偏差の値に応じて指數関数的に増加する関数である

請求項1記載のプログラム。

【請求項3】

前記ペナルティ関数が、前記状態偏差が第1の値以上第2の値以下で0となり、前記状態偏差が前記第1の値より小さい場合に前記状態偏差の値に応じて指數関数的に増加する第1の関数と、前記第2の値より大きい場合前記状態偏差の値に応じて指數関数的に増加する第2の関数とで表され、

前記第1及び第2の微分関数を、前記第1の関数及び前記第2の関数の各々について生成する

請求項1記載のプログラム。

【請求項4】

モデル予測制御の制御対象についての状態偏差の現在値についての関数ベクトルと、制御期間内における前記制御対象に対する線形化制御入力のベクトルとの積が前記関数ベクトルと等しくなるように予め算出されている第1の係数行列と、前記モデル予測制御の制御対象における状態制約に対応するペナルティ関数について予め算出されている第2の係数行列及び係数ベクトルと、前記ペナルティ関数を前記状態偏差について1階偏微分した第1の式及び2階偏微分した第2の式とを格納するデータ格納部から前記第1の式及び第2の式を読み出して、前記状態偏差の現在値から得られる基点の値を代入することで、第1の値及び第2の値を算出し、

前記データ格納部から前記第1の係数行列及び前記第2の係数行列を読み出して、前記第2の値と前記第2の係数行列との積と前記第1の係数行列との和である第1の行列を算出し、

前記データ格納部から前記関数ベクトルを読み出して、前記関数ベクトルに前記状態偏差の現在値を代入することで得られる第1のベクトルと、前記第1の値と前記第2の値との差に前記線形化制御入力の1単位時間前の値から得られる第3の値を乗じた結果と前記係数ベクトルとの積との差である第2のベクトルを算出し、

前記第1の行列の逆行列と前記第2のベクトルとの積により、前記線形化制御入力のベクトルを生成し、

前記線形化制御入力のベクトルの要素のうち最初の時刻のための線形化制御入力を抽出する

処理をプロセッサに実行させるためのプログラム。

【請求項5】

前記状態偏差の現在値が所定範囲内であるか判断する処理をさらに含み、

前記状態偏差の現在値が前記所定範囲内であれば、前記第1の値及び第2の値を算出する処理と、前記第1の行列を算出する処理と、前記第2のベクトルを算出する処理と、前記線形化制御入力のベクトルを生成する処理と、前記線形化制御入力を抽出する処理とを実行する

請求項4記載のプログラム。

【請求項6】

前記データ格納部が、

10

20

30

40

50

前記状態偏差の現在値の値域に応じた前記第1の式及び第2の式を格納しており、
 前記第1の値及び前記第2の値を算出する処理が、
 前記状態偏差の現在値が属する値域を特定し、当該特定された値域に応じた前記第1の
 式及び前記第2の式を読み出す処理
 を含む請求項4又は5記載のプログラム。

【請求項7】

前記状態偏差の現在値が前記所定範囲外であれば、
 前記データ格納部から前記関数ベクトル及び前記第1の係数行列を読み出して、前記関
 数ベクトルに前記状態偏差の現在値を代入することで得られる第1のベクトルと、前記第
 1の係数行列の逆行列との積により、前記線形化制御入力の第2のベクトルを生成し、
 前記線形化制御入力の第2のベクトルの要素のうち最初の時刻のための線形化制御入力
 を抽出する

処理をさらに含む請求項4乃至6のいずれか1つ記載のプログラム。

【請求項8】

(a) モデル予測制御の評価関数とラグランジュ乗数を含み且つ制御対象の状態方程式
 に関する第1の関数との和を含む第2の関数を前記ラグランジュ乗数について偏微分する
 ことで得られ且つ前記制御対象の状態偏差及び前記制御対象に対する線形化制御入力を変
 数として含む第1の一次式群と、(b) 前記第2の関数を前記制御対象の状態偏差について偏
 微分することで得られ且つ前記制御対象の状態偏差及び前記ラグランジュ乗数を変数として含
 む第2の一次式群と、(c) 前記第2の関数を前記制御対象の線形化制御入力について偏微分す
 ることで得られ且つ前記ラグランジュ乗数及び前記制御対象の状態偏差を変数として含む第3
 の一次式群とを格納するデータ格納部から前記第1の一次式群を読み出して、前記状態偏差の現在
 値を除き前記状態偏差の変数を消し込むように順次代入を行って、前記第1の一次式群を、前記状態偏差の現在値と前記線形化制御入力を変数として含む第4の一次式群に変形し、

前記データ格納部から前記第2の一次式群を読み出して、前記ラグランジュ乗数の変数を消し込むように順次代入を行って、前記第2の一次式群を、前記状態偏差を変数として含む、前記ラグランジュ乗数についての第5の一次式群に変形し、

前記データ格納部から前記第3の一次式群を読み出して、前記第4の一次式群及び前記第5の一次式群を用いて、前記第3の一次式群を、前記状態偏差の現在値と前記線形化制
 御入力を変数として含む第6の一次式群に変形し、

前記第6の一次式群を、第1の係数行列と前記線形化制御入力についてのベクトルとの積が、前記状態偏差の現在値の関数ベクトルと等式で表されるように変形し、

前記モデル予測制御の制御対象における状態制約に対応するペナルティ関数を前記状態
 偏差について1階偏微分した第1の微分関数及び2階偏微分した第2の微分関数を生成し、

前記第4の一次式群から前記線形化制御入力についての係数を抽出し、前記ペナルティ
 関数について予め定められている第2の係数行列及び係数ベクトルを算出する

処理を含み、コンピュータにより実行されるオフライン処理方法。

【請求項9】

モデル予測制御の制御対象についての状態偏差の現在値についての関数ベクトルと、制
 御期間内における前記制御対象に対する線形化制御入力のベクトルとの積が前記関数ベクト
 ルと等しくなるように予め算出されている第1の係数行列と、前記モデル予測制御の制
 御対象における状態制約に対応するペナルティ関数について予め算出されている第2の係数
 行列及び係数ベクトルと、前記ペナルティ関数を前記状態偏差について1階偏微分した
 第1の式及び2階偏微分した第2の式とを格納するデータ格納部から前記第1の式及び第
 2の式を読み出して、前記状態偏差の現在値から得られる基点の値を代入することで、第
 1の値及び第2の値を算出し、

前記データ格納部から前記第1の係数行列及び前記第2の係数行列を読み出して、前記
 第2の値と前記第2の係数行列との積と前記第1の係数行列との和である第1の行列を算

10

20

30

40

50

出し、

前記データ格納部から前記関数ベクトルを読み出して、前記関数ベクトルに前記状態偏差の現在値を代入することで得られる第1のベクトルと、前記第1の値と前記第2の値との差に前記線形化制御入力の1単位時間前の値から得られる第3の値を乗じた結果と前記係数ベクトルとの積との差である第2のベクトルを算出し、

前記第1の行列の逆行列と前記第2のベクトルとの積により、前記線形化制御入力のベクトルを生成し、

前記線形化制御入力のベクトルの要素のうち最初の時刻のための線形化制御入力を抽出する

処理を含み、プロセッサにより実行される制御方法。

10

【請求項10】

(a) モデル予測制御の評価関数とラグランジュ乗数を含み且つ制御対象の状態方程式に関する第1の関数との和を含む第2の関数を前記ラグランジュ乗数について偏微分することで得られ且つ前記制御対象の状態偏差及び前記制御対象に対する線形化制御入力を変数として含む第1の一次式群と、(b) 前記第2の関数を前記制御対象の状態偏差について偏微分することで得られ且つ前記制御対象の状態偏差及び前記ラグランジュ乗数を変数として含む第2の一次式群と、(c) 前記第2の関数を前記制御対象の線形化制御入力について偏微分することで得られ且つ前記ラグランジュ乗数及び前記制御対象の状態偏差を変数として含む第3の一次式群とを格納するデータ格納部から前記第1の一次式群を読み出して、前記状態偏差の現在値を除き前記状態偏差の変数を消し込むように順次代入を行って、前記第1の一次式群を、前記状態偏差の現在値と前記線形化制御入力を変数として含む第4の一次式群に変形する手段と、

20

前記データ格納部から前記第2の一次式群を読み出して、前記ラグランジュ乗数の変数を消し込むように順次代入を行って、前記第2の一次式群を、前記状態偏差を変数として含む、前記ラグランジュ乗数についての第5の一次式群に変形する手段と、

前記データ格納部から前記第3の一次式群を読み出して、前記第4の一次式群及び前記第5の一次式群を用いて、前記第3の一次式群を、前記状態偏差の現在値と前記線形化制御入力を変数として含む第6の一次式群に変形する手段と、

前記第6の一次式群を、第1の係数行列と前記線形化制御入力についてのベクトルとの積が、前記状態偏差の現在値の関数ベクトルと等式で表されるように変形する手段と、

30

前記モデル予測制御の制御対象における状態制約に対応するペナルティ関数を前記状態偏差について1階偏微分した第1の微分関数及び2階偏微分した第2の微分関数を生成する手段と、

前記第4の一次式群から前記線形化制御入力についての係数を抽出し、前記ペナルティ関数について予め定められている第2の係数行列及び係数ベクトルを算出する手段と、

を有するオフライン処理装置。

【請求項11】

モデル予測制御の制御対象についての状態偏差の現在値についての関数ベクトルと、制御期間内における前記制御対象に対する線形化制御入力のベクトルとの積が前記関数ベクトルと等しくなるように予め算出されている第1の係数行列と、前記モデル予測制御の制御対象における状態制約に対応するペナルティ関数について予め算出されている第2の係数行列及び係数ベクトルと、前記ペナルティ関数を前記状態偏差について1階偏微分した第1の式及び2階偏微分した第2の式とを格納するデータ格納部から前記第1の式及び第2の式を読み出して、前記状態偏差の現在値から得られる基点の値を代入することで、第1の値及び第2の値を算出する第1の算出部と、

40

前記データ格納部から前記第1の係数行列及び前記第2の係数行列を読み出して、前記第2の値と前記第2の係数行列との積と前記第1の係数行列との和である第1の行列を算出し、前記データ格納部から前記関数ベクトルを読み出して、前記関数ベクトルに前記状態偏差の現在値を代入することで得られる第1のベクトルと、前記第1の値と前記第2の値との差に前記線形化制御入力の1単位時間前の値から得られる第3の値を乗じた結果と

50

前記係数ベクトルとの積との差である第2のベクトルを算出する第2の算出部と、

前記第1の行列の逆行列と前記第2のベクトルとの積により、前記線形化制御入力のベクトルを生成する生成部と、

前記線形化制御入力のベクトルの要素のうち最初の時刻のための線形化制御入力を出力する出力部と、

を有する制御装置。

【発明の詳細な説明】

【技術分野】

【0001】

本技術は、モデル予測制御に関する。

10

【背景技術】

【0002】

モデル予測制御は、各時刻において有限時間未来までの応答を最適化することによって制御対象への入力を決定する制御方法である。操作量飽和などの不等式制約条件を陽に扱うことができる制御手法として高い制御性能を期待されている。

【0003】

このモデル予測制御の適用については様々な分野について考察されているが、ディーゼルエンジンの吸気系制御への適用も考察されている。ディーゼルエンジンの吸気系は、EGR (Exhaust Gas Recirculator: 排気循環器) と VNT (Variable Nozzle Turbo: 可変ノズルターボ) を用いて、MAF (Mass Air Flow: 新気量) と MAP (Manifold Absolute Pressure: 吸気圧) を独立に、SISO (Single Input Single Output) として PID (Proportional, Integral and Differential) 制御を行っていることが多い。しかしながら、EGR と VNT は排気を共有して互いに干渉することから、現行の手法では低環境負荷性能を達成するために十分な目標追従性を確保することが難しくなっている。

20

【0004】

このため、EGR 及び VNT で MAF 及び MAP を制御する 2 入力 2 出力の MIMO (Multi-Input and Multi-Output) 制御が検討されているが、ここで EGR バルブと VNT ノズルの操作量の飽和に起因する目標値追従性能の劣化が大きな問題となっている。この問題の解決策として、上で述べたモデル予測制御が期待されている。

【0005】

30

しかしながら、吸気系のような応答の速い対象に対してモデル予測制御を適用する場合、大きな課題となるのが、サンプル時間間隔毎に行う最適化計算に要する計算時間である。この問題に対しては様々な取り組みがなされているが、その中で使われる手法が、操作量（制御入力とも呼ぶ）飽和条件付きの最適化問題をペナルティ関数を用いて緩和する方法である。

【0006】

一方、エンジンなどの制御対象となるプラントに状態制約が存在する場合については、あまり検討がなされていない。そして、どのようにすれば実時間処理における計算時間を抑えることができるかという点については、具体的に考察されていない。

【先行技術文献】

40

【非特許文献】

【0007】

【非特許文献1】梅田裕平, 丸山次人, 志村知洋, 穴井宏和, "代数的簡略化を用いたモデル予測制御の効率的解法", 第11回計測自動制御学会制御部門大会予稿集, 171-1-3 (2011)

【非特許文献2】Y. Umeda, T. Maruyama, H. Anai, A. Ejiri and K. Shimotani, "A Fast Model Predictive Control Algorithm for Diesel Engines", Proceedings of 4th IFAC Nonlinear Model Predictive Control Conference International Federation of Automatic Control., Paper SuDR.6, 2012.

【非特許文献3】梅田裕平, 丸山次人, 志村知洋, 穴井宏和, 下谷 圭司, "ディーゼル工

50

ンジンの効率的なモデル予測制御のためのペナルティ関数の設定に関する一考察”， 第12回計測自動制御学会制御部門大会予稿集，P0066（2012）

【非特許文献4】梅田裕平，丸山次人，志村知洋，穴井宏和，下谷圭司，”GAを用いたディーゼルエンジンの効率的なモデル予測制御のためのペナルティ関数の設定”，第3回進化計算学会研究会予稿集，P1-5（2012）

【発明の概要】

【発明が解決しようとする課題】

【0008】

従って、本技術の目的は、一側面として、モデル予測制御において制御対象の状態制約を適切に取り扱うための技術を提供することである。

10

【課題を解決するための手段】

【0009】

本技術の第1の態様に係るオフライン処理方法は、(A)(a)モデル予測制御の評価関数とラグランジュ乗数を含み且つ制御対象の状態方程式に関する第1の関数との和を含む第2の関数をラグランジュ乗数について偏微分することで得られ且つ制御対象の状態偏差及び制御対象に対する線形化制御入力を変数として含む第1の一次式群と、(b)第2の関数を制御対象の状態偏差について偏微分することで得られ且つ制御対象の状態偏差及びラグランジュ乗数を変数として含む第2の一次式群と、(c)第2の関数を制御対象の線形化制御入力について偏微分することで得られ且つラグランジュ乗数及び制御対象の状態偏差を変数として含む第3の一次式群とを格納するデータ格納部から第1の一次式群を読み出して、状態偏差の現在値を除き状態偏差の変数を消し込むように順次代入を行って、第1の一次式群を、状態偏差の現在値と線形化制御入力とを変数として含む第4の一次式群に変形し、(B)データ格納部から第2の一次式群を読み出して、ラグランジュ乗数の変数を消し込むように順次代入を行って、第2の一次式群を、状態偏差を変数として含む、ラグランジュ乗数についての第5の一次式群に変形し、(C)データ格納部から第3の一次式群を読み出して、第4の一次式群及び第5の一次式群を用いて、第3の一次式群を、状態偏差の現在値と線形化制御入力とを変数として含む第6の一次式群に変形し、(D)第6の一次式群を、第1の係数行列と線形化制御入力についてのベクトルとの積が、状態偏差の現在値の関数ベクトルと等式で表されるように変形し、(E)モデル予測制御の制御対象における状態制約に対応するペナルティ関数を状態偏差について1階偏微分した第1の微分関数及び2階偏微分した第2の微分関数を生成し、(F)第4の一次式群から線形化制御入力についての係数を抽出し、ペナルティ関数について予め定められている第2の係数行列及び係数ベクトルを算出する処理を含む。

20

【0010】

本技術の第2の態様に係る制御方法は、(A)モデル予測制御の制御対象についての状態偏差の現在値についての関数ベクトルと、制御期間内における制御対象に対する線形化制御入力のベクトルとの積が関数ベクトルと等しくなるように予め算出されている第1の係数行列と、モデル予測制御の制御対象における状態制約に対応するペナルティ関数について予め算出されている第2の係数行列及び係数ベクトルと、ペナルティ関数を状態偏差について1階偏微分した第1の式及び2階偏微分した第2の式とを格納するデータ格納部から第1の式及び第2の式を読み出して、状態偏差の現在値から得られる基点の値を代入することで、第1の値及び第2の値を算出し、(B)データ格納部から第1の係数行列及び第2の係数行列を読み出して、第2の値と第2の係数行列との積と第1の係数行列との和である第1の行列を算出し、(C)データ格納部から関数ベクトルを読み出して、関数ベクトルに状態偏差の現在値を代入することで得られる第1のベクトルと、第1の値と第2の値との差に線形化制御入力の1単位時間前の値から得られる第3の値を乗じた結果と係数ベクトルとの積との差である第2のベクトルを算出し、(D)第1の行列の逆行列と第2のベクトルとの積により、線形化制御入力のベクトルを生成し、(E)線形化制御入力のベクトルの要素のうち最初の時刻のための線形化制御入力を抽出する処理を含む。

30

【発明の効果】

40

50

【0011】

一側面によれば、モデル予測制御において制御対象の状態制約をペナルティ関数で適切に取り扱うことができるようになる。

【図面の簡単な説明】

【0012】

【図1】図1は、ディーゼルエンジンの吸気システムの一例を示す図である。

【図2】図2は、エンジン制御装置のブロック線図の一例を示す図である。

【図3】図3は、オフライン処理装置の機能ブロック図である。

【図4】図4は、オフライン処理装置により実行される処理の処理フローを示す図である。

10

【図5】図5は、モデル予測制御部の機能ブロック図を示す図である。

【図6】図6は、モデル予測制御部により実行される処理の処理フローを示す図である。

【図7】図7は、従来の問題を説明するための図である。

【図8】図8は、従来の問題を説明するための図である。

【図9】図9は、ペナルティ関数の一例を示す図である。

【図10】図10は、ペナルティ関数を説明するための図である。

【図11】図11は、第2の実施の形態に係るモデル予測制御部により実行される処理の処理フローを示す図である。

【図12】図12は、第3の実施の形態におけるペナルティ関数の適用切り替えを説明するための図である。

20

【図13A】図13Aは、第3の実施の形態の効果を説明するための図である。

【図13B】図13Bは、第3の実施の形態の効果を説明するための図である。

【図14】図14は、コンピュータの機能ブロック図である。

【図15】図15は、制御装置の機能ブロック図である。

【発明を実施するための形態】

【0013】

【実施の形態1】

図1にディーゼルエンジンの吸気システムを示す。ディーゼルエンジンの吸気制御系は、吸気圧制御系と新気量制御系を含む。吸気圧制御系は、排気中の微粒子物質 (PM: Particulate Matter) を低減するために、可変ノズルターボVNTのノズル径を制御して吸気圧を吸気圧目標値に追従するようにコントロールしている。一方、新気量制御系は、排気中の窒素酸化物 (NOx) を低減するために、排気をシリンダ内に再循環させる排気循環器EGRのバルブ開度を制御して新気量を新気量目標値に追従するようにコントロールしている。

30

【0014】

すなわち、エンジン本体1には、エンジン本体1からの排ガスを供給する排気循環器EGRと、排ガスの圧力にてタービンを回して新気 (Fresh Air) を圧縮してエンジン本体1に供給する可変ノズルターボVNTとが接続されている。可変ノズルターボVNTのノズル開度を調整することによって、可変ノズルターボVNTのタービンの回転が調整され、吸気圧 (MAP) センサで測定される吸気圧 (MAP) が調整される。一方、排気循環器EGRに設けられているEGRバルブのバルブ開度を調整することによって、新気量 (MAF) センサで測定される新気量 (MAF) が調整される。

40

【0015】

本実施の形態に係るエンジン制御装置1000には、MAPセンサからの吸気圧測定値と、MAFセンサからの新気量測定値と、外部から与えられる燃料噴射量の設定値と、同じく外部から与えられるエンジン回転数の設定値とが入力されるようになっている。また、エンジン制御装置1000からは、EGRバルブのバルブ開度の操作量がEGRバルブに出力され、VNTノズルのノズル開度の操作量がVNTノズルに出力されるようになっている。

【0016】

50

以下、エンジンをプラントの一例として説明するが、本実施の形態の適用範囲は、エンジンに限られない。

【0017】

本実施の形態に係るエンジン制御装置1000のブロック線図を図2に示す。エンジン制御装置1000は、計画器210と、モデル予測制御部220と、変換部225とを有する。プラント240は、例えばモデル化されたエンジン特性に相当し、EGRバルブ及びVNTノズルの開度には物理的に制限があるため、飽和要素230も設けられている。すなわち、プラント240への操作量は、飽和要素230による制限を受ける。

【0018】

このようなエンジン制御装置1000の計画器210には、燃料噴射量の設定値とエンジン回転数の設定値とが入力されて、燃料噴射量の値及びエンジン回転数の値に対応付けてEGRバルブの状態目標値及びVNTノズルの状態目標値V_{ref}、燃料噴射量の値及びエンジン回転数の値に対応付けて吸気圧MAPの目標値及び新気量MAFの目標値X^{ref}が登録されている。本実施の形態では、計画器210は、EGRバルブの状態目標値及びVNTノズルの状態目標値V_{ref}、及び吸気圧MAPの目標値及び新気量MAFの目標値X^{ref}を出力する。

【0019】

この目標値X^{ref}とプラント240からの出力y（例えばMAFセンサ出力及びMAPセンサ出力）との差が、モデル予測制御部220に入力される。モデル予測制御部220は、以下で述べるモデル予測制御の処理を実施して、線形化制御入力uを出力する。この線形化制御入力uは、以下で説明する変換部225（g⁻¹）で、EGRバルブの状態及びVNTノズルの状態vに変換される。そして、状態vは、計画器210からの状態目標値V_{ref}とは加算されて操作量が生成され、当該操作量が飽和要素230を介してプラント240に入力される。このような処理が繰り返される。

【0020】

このようなエンジン本体1の吸気系システムは、EGRバルブ及びVNTノズルについて非線形であり、エンジンのモデルとしても、エンジン回転数及び燃料噴射量に応じて非線形に変化する。

【0021】

ここで、エンジン回転数及び燃料噴射量が固定であるとして、エンジンモデルは、以下のように定式化される。

【数1】

$$\begin{bmatrix} x_{maf}(k+i+1) \\ x_{map}(k+i+1) \end{bmatrix} = A_{\omega(k+i), \sigma(k+i)} \begin{bmatrix} x_{maf}(k+i) \\ x_{map}(k+i) \end{bmatrix} + B_{\omega(k+i), \sigma(k+i)} g \begin{bmatrix} v_{egr}(k+i) \\ v_{vnt}(k+i) \end{bmatrix} \quad (1)$$

【0022】

ここでA_(k)、B_(k)及びg_(k)は、時刻kにおける運転条件（（k））、（k）によって変化する非線形の2×2行列関数である。なお、時刻kにおける回転数を（k）と表し、燃料噴射量を（k）と表す。また、時刻kにおけるMAFの状態偏差をx_{maf}（k）と表し、MAPの状態偏差をx_{map}（k）と表す。また、時刻kにおけるMAFの測定値をy_{maf}（k）と表し、MAPの測定値をy_{map}（k）と表す。さらに、時刻kにおけるEGRバルブの状態をv_{egr}（k）と表し、VNTノズルの状態をv_{vnt}（k）と表す。また、吸気圧MAPの目標値をX_{map}^{ref}と表し、新気量MAFの目標値をX_{maf}^{ref}と表す。

【0023】

従って、状態偏差ベクトルx（k）、測定値ベクトルy（k）、目標値ベクトルX^{ref}（k）及び状態ベクトルv（k）は、以下のように表される。

【数2】

$$x(k) = \begin{bmatrix} x_{maf}(k) \\ x_{map}(k) \end{bmatrix}, y(k) = \begin{bmatrix} y_{maf}(k) \\ y_{map}(k) \end{bmatrix}$$

$$X^{ref}(k) = \begin{bmatrix} X_{maf}^{ref}(k) \\ X_{map}^{ref}(k) \end{bmatrix}, v(k) = \begin{bmatrix} v_{egr}(k) \\ v_{vnt}(k) \end{bmatrix}$$

【0024】

$x(k)$ 、 $y(k)$ 及び $X^{ref}(k)$ の関係は以下のようにになっている。

10

【数3】

$$x(k) = X^{ref}(k) - y(k) \quad (2)$$

【0025】

また、非線形関数 g を以下のように表す。

【数4】

$$g \begin{pmatrix} v_{egr}(k+i) \\ v_{vnt}(k+i) \end{pmatrix} = \begin{pmatrix} g_1(v_{egr}(k+i)) \\ g_2(v_{vnt}(k+i)) \end{pmatrix} \quad (3)$$

20

【0026】

g_1 及び g_2 は、単調増加関数である。実験データからすると、このような仮定は妥当性がある。非線形モデルでは計算速度等から取り扱いが困難なため、以下では、非線形なエンジンモデルを線型モデルに変換する。そこで、上で述べた線形化制御入力 u_{egr} 及び u_{vnt} を導入することで、線形化する。

【数5】

$$\begin{bmatrix} u_{egr}(k+i) \\ u_{vnt}(k+i) \end{bmatrix} = g \begin{pmatrix} (v_{egr}(k+i)) \\ (v_{vnt}(k+i)) \end{pmatrix} \quad (4)$$

30

【0027】

なお、線形化制御入力ベクトル $u(k)$ を以下のように表す。

【数6】

$$u(k) = \begin{bmatrix} u_{egr}(k) \\ u_{vnt}(k) \end{bmatrix}$$

【0028】

本実施の形態では、典型的な運転条件を選択して、セットポイントと呼ぶことにする。そして、モデル予測制御部220は、運転条件 $(\cdot(k), \cdot(k))$ を取得すると、対応するセットポイントについての u 及び v についての行列 A 、 B 、 C 、 D を選択し、以下で述べる処理を行って線形化制御入力 u を算出する。また、変換関数225は、線形化制御入力 u を、変換関数 g^{-1} により非線形の最適状態 v に変換する。この最適状態 v と状態目標値 V^{ref} とは加算されて操作量が生成され、エンジンのプラント240に出力される。

40

【0029】

モデル予測制御部220は、セットポイントについて最適な線形化制御入力 u を、1サンプル時間毎に、以下の線形モデルに従って算出する。

【数7】

$$\begin{bmatrix} x_{maf}(k+i+1) \\ x_{map}(k+i+1) \end{bmatrix} = A_{\omega, \sigma} \begin{bmatrix} x_{maf}(k+i) \\ x_{map}(k+i) \end{bmatrix} + B_{\omega, \sigma} \begin{bmatrix} u_{egr}(k+i) \\ u_{vnt}(k+i) \end{bmatrix} \quad (5)$$

【0030】

但し、線形化制御入力 u についての制約条件は以下のとおりである。

【数8】

$$g\begin{pmatrix} 0 \\ 0 \end{pmatrix} \leq \begin{bmatrix} u_{egr}(k+i) \\ u_{vnt}(k+i) \end{bmatrix} \leq g\begin{pmatrix} 100 \\ 100 \end{pmatrix} \quad (6)$$

10

【0031】

例えば、 $\omega = 2000 \text{ r p m}$ 、 $\sigma = 20 \text{ mm}^3 / \text{s t}$ の場合、以下のような行列が用いられる。

【数9】

$$A_{2000,20} = \begin{pmatrix} 0.928192326 & 0.000101829 \\ -1.015716036 & 0.984926428 \end{pmatrix} \quad (7)$$

【数10】

$$B_{2000,20} = \begin{pmatrix} -0.000488023 & 0.000295048 \\ -0.015300326 & -0.003700605 \end{pmatrix} \quad (8)$$

20

【0032】

各時刻 k についての線形化制御入力 u は、以下に示すような評価関数 J を最小化することで決定される。なお、予測ホライズン H_p 及び制御ホライズン H_c は、予め設定されるものとする。

【数11】

$$\begin{aligned} J(k) = & \sum_{i=1}^{H_p-1} x(k+i)^T Q x(k+i) \\ & + \sum_{i=0}^{H_c-1} \Delta u(k+i)^T R \Delta u(k+i) + x(k+H_p)^T S x(k+H_p) \quad (9) \end{aligned}$$

30

【0033】

ここで、 Q^T は行列 Q の転置行列である。また、 $Q \in M(n, R)$ 、 $R \in M(n, R)$ 及び $S \in M(n, R)$ は、重み行列であり、予め設定される。なお、 $M(n, R)$ は $n \times n$ 行列を表す。(9)式において、無限ホライズンについての最適化問題を、有限ホライズンについての最適化問題にしており、このため、Riccati代数方程式を解く行列として重み行列 S を用いている。

40

【0034】

本実施の形態では、モデル予測制御の問題は、有限ホライズンの最適制御問題を解くことに変形される。

【数12】

$$\text{Minimize } J \quad (10)$$

$$\text{Subject to } \begin{cases} x(k+i+1) = Ax(k+i) + Bu(k+i) & (11a) \\ x(k) = x & (11b) \\ u_{\min} \leq u(k+i) \leq u_{\max} & (11c) \end{cases}$$

ここで、 x は、現在の状態偏差ベクトルを表す。

10

【0035】

最適解を求める手法として、ラグランジュの未定乗数法が効率的であるが、不等式制約がある場合にはこの手法をそのまま用いることができない。従って、不等式制約の代わりにペナルティ関数を導入して、不等式制約のない評価関数に緩和する。

【0036】

すなわち、以下のような評価関数 J_b に緩和する。

【数13】

$$J_b = \sum_{i=1}^{H_p-1} x(k+i)^T Q x(k+i) + \sum_{i=0}^{H_c-1} \Delta u(k+i)^T R \Delta u(k+i) + p(u(k+i)) + x(k+H_p)^T S x(k+H_p) \quad (12)$$

20

【0037】

ここで、 $p(u(k+i))$ はペナルティ関数であり、入力値 $u(k+i)$ が、可能な値域の境界に近づくと、大きな値を評価関数 J に加えることになる。

【0038】

一般的に、不等式制約 $u \leq u^{\max}$ に対して用いられるペナルティ関数としては以下のようなものがある。

【数14】

30

$$(a) \quad p(u) = r \left(\min(0, u^{\max} - u) \right)^2$$

$$(b) \quad p(u) = -r \log(u^{\max} - u)$$

$$(c) \quad p(u) = \frac{r}{(u^{\max} - u)}$$

40

$$(d) \quad p(u) = \begin{cases} 0 & \text{if } u \leq u^{\max} \\ \infty & \text{if } u > u^{\max} \end{cases}$$

r は、ペナルティ関数の係数であり、不等式制約に対する重みを表している。

【0039】

本実施の形態では、以下の演算のためには、(c)のようなシンプルなペナルティ関数が好ましいので、(c)を用いることにする。

50

【0040】

ディーゼルエンジンの場合、線形化制御入力 u には以下のような制約が存在する。

【数15】

$$u_{egr}^{\min} \leq u_{egr}(k+i) \leq u_{egr}^{\max} \quad (13)$$

$$u_{vnt}^{\min} \leq u_{vnt}(k+i) \leq u_{vnt}^{\max} \quad (14)$$

【0041】

(c)のような形のペナルティ関数を採用する場合には、ペナルティ関数は以下のように表される。

【数16】

$$p(u) = \sum_{i=0}^{Hu-1} \frac{r_{egr}^{\max}}{u_{egr}^{\max} - u_{egr}(k+i)} + \sum_{i=0}^{Hu-1} \frac{r_{egr}^{\min}}{u_{egr}(k+i) - u_{egr}^{\min}} \\ + \sum_{i=0}^{Hu-1} \frac{r_{vnt}^{\max}}{u_{vnt}^{\max} - u_{vnt}(k+i)} + \sum_{i=0}^{Hu-1} \frac{r_{vnt}^{\min}}{u_{vnt}(k+i) - u_{vnt}^{\min}} \quad (15)$$

【0042】

(15)式を用いて制約条件を緩和する場合、以下に示すような、有限ホライズンについての最適制御問題を解くことに帰着する。

【数17】

$$\text{Minimize } Jb \quad (16)$$

$$\text{Subject to } \begin{cases} x(k+i+1) = Ax(k+i) + Bu(k+i) \\ x(k) = x \end{cases} \quad (17a) \quad (17b)$$

【0043】

このような最適制御問題を解くためには、ラグランジュの未定乗数法が用いられる。ここで、ラグランジュ乗数 $(j) = [m_{af}(j), m_{ap}(j)]^T$ を用い、ハミルトニアン H を以下のように表す。

【数18】

$$H = Jb + \chi \quad (18)$$

【0044】

ここで、 (\cdot) (の大文字) は、以下のように定義される。

【数19】

$$\chi = \sum_{j=0}^{H_p-1} \lambda(j+1) (x(k+j+1) - Ax(k+j) - Bu(k+j)) \quad (19)$$

【0045】

$x, u, H = 0$ と置けば、以下のような方程式が得られる。

【数20】

$$\frac{\partial H}{\partial x} = \frac{\partial J}{\partial x} + \frac{\partial \chi}{\partial x} = 0 \quad (20)$$

$$\frac{\partial H}{\partial u} = \frac{\partial J}{\partial u} + \frac{\partial \chi}{\partial u} + \frac{\partial p}{\partial u} = 0 \quad (21)$$

$$\frac{\partial H}{\partial \lambda} = 0 \quad (22)$$

なお、(20)式において $J b / x = J / x + p(u) / x = J / x$ 10 と変形されている。

【0046】

なお、/ x, / u, / は、以下のような演算である。

【数21】

$$\frac{\partial}{\partial x} = \left[\frac{\partial}{\partial x_{maf}(k+1)}, \frac{\partial}{\partial x_{map}(k+1)}, \frac{\partial}{\partial x_{maf}(k+2)}, \dots, \frac{\partial}{\partial x_{map}(k+H_p)} \right]^T$$

$$\frac{\partial}{\partial u} = \left[\frac{\partial}{\partial u_{egr}(k)}, \frac{\partial}{\partial u_{vnt}(k)}, \frac{\partial}{\partial u_{egr}(k+1)}, \dots, \frac{\partial}{\partial u_{vnt}(k+H_c)} \right]^T \quad 20$$

$$\frac{\partial}{\partial \lambda} = \left[\frac{\partial}{\partial \lambda_{maf}(k+1)}, \frac{\partial}{\partial \lambda_{map}(k+1)}, \frac{\partial}{\partial \lambda_{maf}(k+2)}, \dots, \frac{\partial}{\partial \lambda_{map}(k+H_p)} \right]^T$$

【0047】

なお、方程式(20)乃至(22)は、p/uの部分を除き線形方程式となっている。p/uを、直前のステップの最適線形化制御入力 $u \sim (\sim)$ は u の上に付される)でテイラー展開することで線形化すると、以下のように表される。

【数22】

$$\frac{\partial p}{\partial u}(u) = \frac{\partial p}{\partial u}(\tilde{u}) + \frac{\partial^2 p}{\partial u^2}(\tilde{u})(u - \tilde{u}). \quad (23)$$

【0048】

(23)式の第1項のEGR成分は、以下のように表される。なお、VNT成分も同様の形で表される。

【数23】

$$\begin{aligned} \frac{\partial p}{\partial u_{egr}(k)} = & \left(\frac{2r_{egr}^{\max}}{(u_{egr}^{\max} - \tilde{u}_{egr}(k))^3} - \frac{2r_{egr}^{\min}}{(u_{egr}^{\min} - \tilde{u}_{egr}(k))^3} \right) u_{egr}(k) - \frac{2r_{egr}^{\max} \tilde{u}_{egr}(k)}{(u_{egr}^{\max} - \tilde{u}_{egr}(k))^3} \\ & + \frac{2r_{egr}^{\min} \tilde{u}_{egr}(k)}{(u_{egr}^{\min} - \tilde{u}_{egr}(k))^3} + \frac{r_{egr}^{\max}}{(u_{egr}^{\max} - \tilde{u}_{egr}(k))^2} - \frac{r_{egr}^{\min}}{(u_{egr}^{\min} - \tilde{u}_{egr}(k))^2} \end{aligned} \quad 40 \quad (24)$$

【0049】

このような線形近似を行えば、(20)乃至(22)式は線形となる。そうすると、線形近似された(20)乃至(22)式は以下のように表される。

【数24】

$$MU = b \quad (25)$$

ここで M は、 $(2H_p + H_c) \times (2H_p + H_c)$ の行列であり、 U 及び b は、 $(2H_p + H_c)$ ベクトルである。ベクトル U は、以下のように表される。

【数25】

$$U = \begin{bmatrix} x_{maf}(k+1), x_{map}(k+1), x_{maf}(k+2), \dots, x_{maf}(k+H_p), x_{map}(k+H_p), \\ u_{egr}(k), u_{vnt}(k), u_{egr}(k+1), \dots, u_{egr}(k+H_c), u_{vnt}(k+H_c), \\ \lambda_{maf}(k+1), \lambda_{map}(k+1), \lambda_{maf}(k+2), \dots, \lambda_{maf}(k+H_p), \lambda_{map}(k+H_p) \end{bmatrix}^T \quad (26)$$

【0050】

また、ベクトル b は、以下のように表される。なお、 b_{maf} 、 b_{map} 及び b_{const} は、行列の変換によって算出される $(2H_p + H_c)$ 行列である。

$$b = x_{maf} b_{maf} + x_{map} b_{map} + b_{const} \quad (27)$$

【0051】

最適な線形化制御入力は、この線形方程式を解くことによって得られる。しかしながら、行列 M のサイズが大きければ計算時間は長くなってしまうので、行列 M のサイズをオフラインの前処理にて削減する。

【0052】

まず、(22)式の第1乃至第3の方程式を具体的に書き下すと以下のようになる。

【数26】

$$x_{maf}(k+1) = a_{11}x_{maf}(k) + a_{12}x_{map}(k) + b_{11}u_{egr}(k) + b_{12}u_{vnt}(k) \quad (28-1)$$

$$x_{map}(k+1) = a_{21}x_{maf}(k) + a_{22}x_{map}(k) + b_{21}u_{egr}(k) + b_{22}u_{vnt}(k) \quad (28-2)$$

$$x_{maf}(k+2) = a_{11}x_{maf}(k+1) + a_{12}x_{map}(k+1) + b_{11}u_{egr}(k+1) + b_{12}u_{vnt}(k+1) \quad (29)$$

【0053】

(28-1)式及び(28-2)式の状態偏差 $x_{maf}(k)$ 及び $x_{map}(k)$ は、現在値である。従って、(28-1)式によれば、 $x_{maf}(k+1)$ は線形化制御入力 $u(k)$ の1次式で表される。(28-2)式によれば、 $x_{map}(k+1)$ も、線形化制御入力 $u(k)$ の1次式で表される。また、(29)式によれば、 $x_{maf}(k+2)$ は、状態偏差及び線形化制御入力の1次式で表される。しかしながら、(28-1)式及び(28-2)式を(29)式に代入すると、 $x_{maf}(k+2)$ を、線形化制御入力 $u(k)$ 及び $u(k+1)$ で表すことができるようになる。これを i を増加させつつ繰り返せば、状態偏差 $x_{maf}(k+i)$ 及び $x_{map}(k+i)$ は、線形化制御入力の1次式として以下のように表される。

【数27】

$$x(k+i) = F(x(k), u(k), \dots, u(k+H_c-1)) \quad (30)$$

【0054】

一方、(20)式の $(2H_p - 1)$ 番目及び $(2H_p - 3)$ 番目の方程式を具体的に書き下すと以下のようになる。

10

20

30

40

【数28】

$$s_{11}x_{maf}(k+H_p) + \lambda_{maf}(H_p) = 0 \quad (31)$$

$$s_{22}x_{map}(k+H_p) + \lambda_{map}(H_p) = 0 \quad (32)$$

$$r_{11}x_{maf}(k+H_p-1) + \lambda_{maf}(H_p-1) - a_{11}\lambda_{maf}(H_p) - a_{22}\lambda_{map}(H_p) = 0 \quad (33)$$

ここで、 s_{11} 、 s_{22} 、 r_{11} は、重み行列 S 及び R の行列成分を表す。

【0055】

この(31)式及び(32)式は、 $_{maf}(H_p)$ 及び $_{map}(H_p)$ の一次式で表されることが分かる。そこで、(31)式及び(32)式を、(33)式に代入すると、 $_{maf}(H_p-1)$ が、目標値 X^{ref}_{maf} 、状態偏差 $x(k+H_p)$ 及び $x(k+H_p-1)$ で表すことができる。このような処理を i を減少させつつ順次行うことで、ラグランジュ乗数 $(k+i)$ は、以下のように表される。

【数29】

$$\lambda(k+i) = G_1(x(k), \dots, x(k+H_p)) \quad (34)$$

【0056】

さらに、(30)式を(34)式に代入することで、ラグランジュ乗数 $(k+i)$ は、線形化制御入力 $u(k)$ の一次式に変形される。

【数30】

$$\lambda(k+i) = G_2(x(k), u(k), \dots, u(k+H_c-1)) \quad (35)$$

【0057】

最後に、(30)式及び(35)式を、(21)式に代入すると、以下のように表される。

【数31】

$$\frac{\partial H}{\partial u} = \frac{\partial H}{\partial u}(x(k), u(k), \dots, u(k+H_c-1)) = 0 \quad (36)$$

【0058】

このように、(21)式は、状態偏差の初期値 $x(k)$ 及び線形化制御入力 u の一次式で表されるようになる。そうすると、(36)式は、以下の形式に直すことができる。

【数32】

$$M_2 U_2 = b_2 \quad (37)$$

【0059】

ここで M_2 は $H_c \times H_c$ 行列である。 U_2 及び b_2 は、以下のような形のベクトルである。

【数33】

$$U_2 = [u_{egr}(k), u_{vnt}(k), u_{egr}(k+1), \dots, u_{egr}(k+H_c-1), u_{vnt}(k+H_c-1)] \quad (38)$$

【数34】

$$b_2 = x_{maf}(k)b_{2maf} + x_{map}(k)b_{2map} + b_{2const} \quad (39)$$

なお、 b_{2maf} 、 b_{2map} 及び b_{2const} は、行列の変換によって算出される $(2H_p + H_c)$ 行列である。

【0060】

このような処理を数式処理を用いてオフラインで行っておけば、エンジン制御装置 1000 では、(37)式を解く演算を行えばよい。行列 M_2 のサイズは、行列 M よりも小さ

くなつており、計算量も削減されるため、高速に制御入力を得ることができるようになる。

【0061】

次に、定常偏差を削減するために、状態制約を導入する場合を考える。説明を簡単にするために、M A F の状態偏差について以下のような制約を導入する。M A P についてはここでは取り扱わないが、同様に設定する場合もある。

【数35】

$$x_{\min} \leq x_{maf}(i) \leq x_{\max} \quad (40)$$

【0062】

しかしながら、このままでモデル予測制御問題が不等式制約を含む問題となり、現実的には解けないので、例えば、以下に示すようなペナルティ関数を導入することで、不等式制約問題を等式制約問題に緩和する。

【数36】

$$\sum_{i=1}^{H_p} q(x(k+i)) = \sum_{i=1}^{H_p} \left(\frac{r_{\max}}{x_{\max} - x_{maf}(k+i)} + \frac{r_{\min}}{x_{maf}(k+i) - x_{\min}} \right) \quad (42)$$

【0063】

ここでは、評価関数 J_b に対して (42) 式のペナルティ関数を加える。そうすると、以下のように表される。

【数37】

$$J_c = J + p(u) + \sum_{i=1}^{H_p} q(x(k+i)) \quad (43)$$

$$H_2 = J_c + \chi \quad (44)$$

【0064】

そして、(30) 式からすると、 $x(k+i)$ は、以下のように書き下される。

【数38】

$$\begin{aligned} x_{maf}(k+i) &= \alpha_1^i u_{egr}(k) + \alpha_2^i u_{vnt}(k) + \alpha_3^i u_{egr}(k+1) + \\ &\dots + \alpha_{2H_c}^i u_{vnt}(k+H_c-1) + \beta_1^i x_{maf}(k) + \beta_2^i x_{map}(k) \end{aligned} \quad (45)$$

【0065】

そうすると、 $x(k+i)$ は、全ての i について、 $u(0)$ 乃至 $u(H_c - 1)$ の関数であるので、(42) 式のペナルティ関数についても $u(0)$ 乃至 $u(H_c - 1)$ の関数と考えることができる。よって、入力制約について (20) 式乃至 (22) 式で記述された最適化問題は、以下のように変形される。

【数39】

$$\frac{\partial H_2}{\partial x} = \frac{\partial J}{\partial x} + \frac{\partial \chi}{\partial x} = 0 \quad (46)$$

$$\frac{\partial H_2}{\partial u} = \frac{\partial J}{\partial u} + \frac{\partial \chi}{\partial u} + \frac{\partial p}{\partial u} + \sum_{i=1}^{H_p} \frac{\partial q(x(k+i))}{\partial u} = 0 \quad (47)$$

$$\frac{\partial H_2}{\partial \lambda} = 0 \quad (48)$$

【0066】

このような(46)式乃至(48)式を解くことになるが、そのためには状態についてのペナルティ関数に関する $q(x) / u$ を線形化する。

【0067】

まず、 $q(x_{maf}(k+i)) / u_{egr}(k+j)$ は以下のように表される。

【数40】

$$\frac{\partial q(x_{maf}(k+i))}{\partial u_{egr}(k+j)} = \frac{\partial q(x_{maf}(k+i))}{\partial x_{maf}(k+i)} \frac{\partial x_{maf}(k+i)}{\partial u_{egr}(k+j)} \quad (49)$$

【0068】

ここで(45)式があるので、(49)式は、以下のように表される。

【数41】

$$\frac{\partial q(x_{maf}(k+i))}{\partial u_{egr}(k+j)} = \alpha_{2^{j+1}}^i \frac{\partial q(x_{maf}(k+i))}{\partial x_{maf}(k+i)} \quad (51)$$

【0069】

次に、(51)式中の $q(x_{maf}(k+i)) / x_{maf}(k+i)$ を線形化する。その際、以下の最適な線形化制御入力 u で 1 次の泰ラー展開を行うと、 $q(x_{maf}(k+i)) / x_{maf}(k+i)$ は以下のように表される。

【数42】

$$\tilde{u} = (\tilde{u}_{egr}(k), \tilde{u}_{vnt}(k), \dots, \tilde{u}_{vnt}(k+H_c-1))$$

【数43】

$$\begin{aligned} \frac{\partial q(x_{maf}(k+i))}{\partial x_{maf}(k+i)}(u) \approx & \frac{\partial q(x_{maf}(k+i))}{\partial x_{maf}(k+i)}(\tilde{u}) + \frac{\partial^2 q(x_{maf}(k+i))}{\partial x_{maf}(k+i) \partial u_{egr}(k)}(\tilde{u})(u_{egr}(k) - \tilde{u}_{egr}(k)) + \\ & \dots + \frac{\partial^2 q(x_{maf}(k+i))}{\partial x_{maf}(k+i) \partial u_{vnt}(k+H_c-1)}(\tilde{u})(u_{vnt}(k+H_c-1) - \tilde{u}_{vnt}(k+H_c-1)) \end{aligned} \quad (52) \quad 30$$

【0070】

また、(51)式と同様に、以下のように表される。

【数44】

$$\frac{\partial^2 q(x_{maf}(k+i))}{\partial x_{maf}(k+i) \partial u_{egr}(k+j)} = \alpha_{2^{j+1}}^i \frac{\partial^2 q(x_{maf}(k+i))}{\partial x_{maf}(k+i)^2} \quad (53)$$

【0071】

従って、以下のような近似式を得られる。

【数45】

$$\begin{aligned} \frac{\partial q(x_{maf}(k+i))}{\partial x_{maf}(k+i)}(u) \approx & \frac{\partial q(x_{maf}(k+i))}{\partial x_{maf}(k+i)}(\tilde{u}) + \alpha_1^i \frac{\partial^2 q(x_{maf}(k+i))}{\partial x_{maf}(k+i)^2}(\tilde{u})(u_{egr}(k) - \tilde{u}_{egr}(k)) + \\ & \dots + \alpha_{2^{H_c}}^i \frac{\partial^2 q(x_{maf}(k+i))}{\partial x_{maf}(k+i)^2}(\tilde{u})(u_{vnt}(k+H_c-1) - \tilde{u}_{vnt}(k+H_c-1)) \end{aligned} \quad (54)$$

【0072】

10

20

40

50

よって、(51)式及び(54)式を組み合わせると、以下のような近似式が得られる。

【数46】

$$\begin{aligned}
 & \frac{\partial q(x_{maf}(k+i))}{\partial u_{egr}(k+j)} \\
 &= \frac{\partial^2 q(x_{maf}(k+i))}{\partial x_{maf}(k+i)^2}(\tilde{u}) \left(\alpha_{2j+1}^i \alpha_1^i u_{egr}(k) + \alpha_{2j+1}^i \alpha_2^i v_{vnt}(k) + \dots + \alpha_{2j+1}^i \alpha_{2H_c}^i u_{vnt}(k+H_c-1) \right) \\
 & \quad + \alpha_{2j+1}^i \left(\frac{\partial q(x_{maf}(k+i))}{\partial x_{maf}(k+i)}(\tilde{u}) - \right. \\
 & \quad \left. \frac{\partial^2 q(x_{maf}(k+i))}{\partial x_{maf}(k+i)^2}(\tilde{u}) (\alpha_1^i u_{egr}(k) + \alpha_2^i u_{vnt}(k) + \dots + \alpha_{2H_c}^i u_{vnt}(k+H_c-1)) \right) \quad (55)
 \end{aligned}$$

10

【0073】

ここで、 $x_{\sim maf}(k+i)$ （"~"は x の上に載っているものとする）を、 u_{\sim} に対応する状態偏差ベースポイントと設定する。そして、短い時間における状態偏差の値の変動は小さいので、 $x_{base maf}(k)$ は以下のように表される。

20

【数47】

$$x_{maf}^{base}(k) = (1 \ 0) \left(Ax(k) + Bu^*(k-1) \right) \quad (56)$$

そして、全ての i について $x_{\sim maf}(k+i)$ を $x_{base maf}(k)$ と近似する。 $x(k)$ は、上でも述べたように現在の状態偏差の値を表しており、 $u^*(k-1)$ は、1単位時間前の最適な線形化制御入力である。

【0074】

そうすると、(45)式から、(55)式の定数部分は、以下のように表される。

30

【数48】

$$\begin{aligned}
 & \alpha_1^i \tilde{u}_{egr}(k) + \alpha_2^i \tilde{u}_{vnt}(k) + \dots + \alpha_{2H_c}^i \tilde{u}_{vnt}(k+H_c-1) \\
 &= \tilde{x}_{maf}(k) - \beta_1^i x_{maf}(k-1) - \beta_2^i x_{map}(k-1) \quad (57)
 \end{aligned}$$

【0075】

また、 $x_{\sim maf}(k)$ を $x_{base maf}(k)$ と近似する。そうすると、(56)式から、以下のような近似式が得られる。

【数49】

$$\alpha_1^i \tilde{u}_{egr}(k) + \alpha_2^i \tilde{u}_{vnt}(k) + \dots + \alpha_{2H_c}^i \tilde{u}_{vnt}(k+H_c-1) \quad (58)$$

40

$$= \tilde{x}_{maf}(k) - \beta_1^i x_{maf}(k-1) - \beta_2^i x_{map}(k-1) \quad (59)$$

$$\approx x_{maf}^{base}(k) - (1 \ 0) Ax(k) \quad (60)$$

$$= (1 \ 0) Bu(k-1) \quad (61)$$

【0076】

そこで、

$$d(x_{base maf}(k)) = (1 \ 0) B u(k-1) \quad (1 \ 0 \ 1) \text{と置く。}$$

50

【0077】

さらに、 $x_{\sim maf}(k+i)$ を $x_{base maf}(k)$ と近似すれば、 $\{\partial^2 q(x_{maf}(k+i)) / x_{maf}(k+i)^2\} (u \sim)$ は、全ての*i*についてベース状態偏差 $x_{base maf}(k)$ に関して同じ関数になる。同様に、 $\{\partial q(x_{maf}(k+i)) / x_{maf}(k+i)\} (u \sim)$ は、全ての*i*についてベース状態偏差 $x_{base maf}(k)$ に関して同じ関数になる。

【0078】

以下、ペナルティ関数の x ((42)式であれば x_{maf})についての一階偏微分を (x) と表し、二階偏微分を $\partial^2(x)$ と表し、 $x = x_{base maf}(k)$ についての $(x_{base maf}(k))$ 及び $\partial^2(x_{base maf}(k))$ によって、(55)式は以下のように表される。

10

【数50】

$$\begin{aligned} & \frac{\partial q(x_{maf}(k+i))}{\partial u_{egr}(k+j)} \\ & \approx \partial^2(x_{maf}^{base}(k)) \left(\alpha_{2j+1}^i \alpha_1^j u_{egr}(k) + \alpha_{2j+1}^j \alpha_2^i u_{vnt}(k) + \dots + \alpha_{2j+1}^i \alpha_{2H_c}^i u_{vnt}(k+H_c-1) \right) \\ & \quad + \alpha_{2j+1}^i \left(\partial(x_{maf}^{base}(k)) - \partial^2(x_{maf}^{base}(k)) d(x_{maf}^{base}(k)) \right) \end{aligned} \quad (62)$$

【0079】

20

この(62)式から、以下のような近似も行われる。

【数51】

$$\begin{aligned} & \frac{\partial q(x_{maf})}{\partial u_{egr}(k+i)} = \frac{\partial \sum_{i=1}^{H_p} q(x_{maf})(k+i)}{\partial u_{egr}(k+i)} \\ & \approx \partial^2(x_{maf}^{base}(k)) \left(\sum_{i=1}^{H_p} \alpha_{2j+1}^i \alpha_1^j u_{egr}(k) + \sum_{i=1}^{H_p} \alpha_{2j+1}^j \alpha_2^i u_{vnt}(k) + \dots + \sum_{i=1}^{H_p} \alpha_{2j+1}^i \alpha_{2H_c}^i u_{vnt}(k+H_c-1) \right) \\ & \quad + \sum_{i=1}^{H_p} \alpha_{2j+1}^i \left(\partial(x_{maf}^{base}(k)) - \partial^2(x_{maf}^{base}(k)) d(x_{maf}^{base}(k)) \right) \end{aligned} \quad (63)$$

30

【0080】

さらに、この(63)式から、以下のような近似も行われる。

【数52】

$$\begin{aligned} & \frac{\partial q(x_{maf})}{\partial u} = \left[\frac{\partial q(x_{maf})}{\partial u_{egr}(k)}, \frac{\partial q(x_{maf})}{\partial u_{vnt}(k)}, \dots, \frac{\partial q(x_{maf})}{\partial u_{vnt}(k+H_c-1)} \right]^T \\ & \approx \partial^2(x_{maf}^{base}(k)) \Lambda U_2 + \left(\partial(x_{maf}^{base}(k)) - \partial^2(x_{maf}^{base}(k)) d(x_{maf}^{base}(k)) \right) v \end{aligned} \quad (64)$$

40

ここで、 Λ 及び U_2 は以下のように表される。

【数53】

$$\Lambda = \left(\sum_{i=1}^{H_p} \alpha_p^i \alpha_q^i \right)_{p,q}$$

【数54】

$$V = \left[\sum_{i=1}^{H_p} \alpha_1^i, \dots, \sum_{i=1}^{H_p} \alpha_{2H_c}^i \right]^T$$

なお、 p における p は $2j+1$ で固定で、 q については、1から H_c まで増加する。すなわち、 V は、1行 H_c 列の行列である。

【0081】

従って、(64)式及び(37)式から、状態についてのペナルティ関数を伴う最適化問題は、以下のように緩和される。

【数55】

$$\left(M_2 + \partial^2 \left(x_{maf}^{base}(k) \right) \Lambda \right) U_2 = b_2 - \left(\partial \left(x_{maf}^{base}(k) \right) - \partial^2 \left(x_{maf}^{base}(k) \right) d \left(x_{maf}^{base}(k) \right) \right) V \quad (65)$$

【0082】

(65)式をエンジン制御装置1000で解くためには、 M_2 、 b_2 、 Λ 及び V を予めオフラインで用意しておき、ペナルティ関数の x_{maf} についての一階偏微分 (x) と、二階偏微分 (x^2) を予めオフラインで算出しておく。そして、エンジン制御装置1000では、 x_{maf}^{base} を(56)式で算出するとともに、 (x_{maf}^{base}) 及び $(x_{maf}^{base})^2$ を算出し、さらに $d(x_{maf}^{base}) = (1 \ 0) B u(k-1)$ を算出する。そして(65)式を解くことになる。

【0083】

なお、上では x_{maf} について状態制約を設ける例を示したが、 x_{map} について状態制約を設けるようにしてもよいし、それら両方に状態制約を設けるようにしてもよい。基本的な処理は上で述べたとおりである。

【0084】

次に、本実施の形態に係るオフライン処理装置100の機能ブロック図を図3に示す。本実施の形態では、オフライン処理装置100は、入力部101と、入力データ格納部102と、第1変形処理部103と、第1データ格納部104と、第2変形処理部105と、第2データ格納部106と、第3変形処理部107と、第3データ格納部108と、第1行列生成部109と、第4データ格納部110と、第2行列生成部111と、微分処理部112とを有する。

【0085】

次に、オフライン処理装置100の処理内容を図4を用いて説明する。まず、入力部101は、ユーザからの入力データとして、行列A、B、S及びR、予測ホライズン H_p 及び制御ホライズン H_c などの値の入力を受け付け、入力データ格納部102に格納する(図4:ステップS1)。なお、入力データ格納部102に格納されていない場合には、(46)乃至(48)式のデータも入力部101を介して入力データ格納部102に格納されるものとする。同様に、入力制約に係る項の式のデータ((24)式)についても同様に、入力データ格納部102に格納されており、格納されていない場合には入力部101を介して入力される。さらに、状態制約に係る式のデータについても同様に、入力データ格納部102に格納されており、格納されていない場合には入力部101を介して入力される。

【0086】

次に、第1変形処理部103は、入力データ格納部102に格納されているデータを用いて、状態偏差 x についての第1変形処理を実行し、処理結果を第1データ格納部104に格納する(ステップS3)。(22)式に由来する(28-1)及び(28-2)式を(29)式に代入する処理を i を増加させながら繰り返して実行し、(30)式を生成する処理を実行する。なお、(30)式については、(45)式のように書き下される。

【0087】

また、第2変形処理部105は、入力データ格納部102に格納されているデータを用

10

20

30

40

50

いて、ラグランジュ乗数についての第2変形処理を実行し、処理結果を第2データ格納部106に格納する(ステップS5)。(20)式に由来する(31)及び(32)式を(33)式に代入する処理をiを減少させながら繰り返して実行し、(34)式を生成する処理を実行する。

【0088】

さらに、第3変形処理部107は、第1データ格納部104及び第2データ格納部106に格納されているデータを用いて、ラグランジュ乗数についての第3変形処理を実行し、処理結果を第3データ格納部108に格納する(ステップS7)。(30)式を(34)式に代入する処理を実行して、(35)式を生成する。

【0089】

そして、第1行列生成部109は、(30)式(第1データ格納部104に格納されている)及び(35)式(第3データ格納部108に格納されている)を(21)式(入力データ格納部102に格納されている)に代入して連立方程式(36)式を生成し、これを(37)式の形に変形することで、行列M₂及びb₂を算出し、第4データ格納部110に格納する(ステップS9)。ここまで処理は、特開2012-194960号公報に記載の技術とほぼ同様である。

【0090】

その後、第2行列生成部111は、第1データ格納部104に格納されている、xについての第1変形処理の結果((30)式)を用いて、入力データ格納部102に格納されている式に従って、係数行列及び係数ベクトルを生成し、第4データ格納部110に格納する(ステップS11)。

【0091】

さらに、微分処理部112は、第1データ格納部102に格納されている、状態制約についてのペナルティ関数のxについての1階偏微分関数(x)と、2階偏微分関数²(x)を生成し、第4データ格納部110に格納する(ステップS13)。1階偏微分関数及び2階偏微分関数の生成については、周知であるからここでは説明を省略する。

【0092】

以上のような処理を実行することで、エンジン制御装置1000の処理負荷を大幅に削減でき、処理速度を高速化できる。

【0093】

次に、図5にエンジン制御装置1000におけるモデル予測制御部220の構成例を示す。モデル予測制御部220は、微分値算出部2201と、行列算出部2202と、出力生成部2203と、データ格納部2205とを有する。

【0094】

データ格納部2205には、第4データ格納部110に格納されているデータが格納される。

【0095】

次に、図6を用いて、エンジン制御装置1000におけるモデル予測制御部220の処理内容について説明する。

【0096】

まず、モデル予測制御部220は、x(k)(=X_{ref}(k)-y(k))を取得する(図6:ステップS21)。

【0097】

そうすると、モデル予測制御部220は、関数ベクトルb₂にx(k)を代入して関数ベクトルb₂を数値ベクトル化する(ステップS23)。

【0098】

また、微分値算出部2201は、x^{base}(k)(x^{base_{maf}}(k)であれば(56)式)を算出し、データ格納部2205に格納されているペナルティ関数の1階偏微分関数(x)と、2階偏微分関数²(x)にx^{base}(k)を代入し、(x^{base}(k))及び²(x^{base}(k))を求める(ステップS25)。

10

20

30

40

50

【0099】

そして、行列算出部2202は、 M_3 及び b_3 を算出する（ステップS27）。

【0100】

(65)式の左辺におけるベクトル U_2 以外の部分である($M_2 + \frac{1}{2}(x^{base}(k))$)が M_3 である。また、(65)式の右辺が b_3 である。すなわち、 $b_3 = b_2 - \frac{1}{2}(x^{base}(k)) - \frac{1}{2}(x^{base}(k))d(x^{base}(k))$ である。 b_3 を算出するためには、 $d(x^{base}(k))$ を、 $B_u(k-1)$ で算出する。

【0101】

そして、出力生成部2203は、 $U = M^{-1}b_3$ によってベクトル U を算出する（ステップS29）。さらに、出力生成部2203は、ベクトル U から線形化制御入力 $u(k)$ ($i = 0$)を抽出し、出力する（ステップS31）。

【0102】

その後、モデル予測制御部220は、処理が終了するのかを判断し（ステップS33）、処理終了でなければ処理はステップS21に戻り、処理終了であれば処理は終了する。

【0103】

以上のような処理で分かるように、オフライン処理によって算出されているデータを活用でき、 M_3 のサイズも削減されているので、状態制約についてのペナルティ関数を導入することでステップS25及びS27が追加されても、高速に線形化制御入力を得ることができるようになる。さらに、ペナルティ関数の近似も適切に行われてあり、高い精度で適切な制御がなされるようになる。

【0104】

なお、ディーゼルエンジンの吸気系制御では、図7（縦軸はMAFを表し、横軸は時間を表す）に示すようにMAFの状態 y_{maf} が目標値 X^{ref}_{maf} を下回ると大量の微粒子物質が発生し、吸気系のつまりの原因となることがある。そのため、上で述べたようにMAFに対して下限値を設定して制御を行うこととする。

【0105】

例えば、以下のようなモデルを採用するものとする。

【数56】

$$\begin{bmatrix} x_{maf}(k+1) \\ x_{map}(k+1) \end{bmatrix} = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 1 \end{bmatrix} \begin{bmatrix} x_{maf}(k) \\ x_{map}(k) \end{bmatrix} + \begin{bmatrix} -2 & 10 \\ -30 & -20 \end{bmatrix} \begin{bmatrix} u_{egr}(k) \\ u_{vnt}(k) \end{bmatrix}$$

【0106】

この時に、例えば $x_{maf} \leq 10$ という状態制約を設定する。そして、予測ホライズン $H_p = 4$ ステップ、制御ホライズン $H_c = 2$ ステップとする。この際、状態制約についてのペナルティ関数は、以下のような1階偏微分関数及び2階偏微分関数が用いられる。

【数57】

$$\partial(x) = -\frac{1}{(x+10)^2}$$

$$\partial^2(x) = -\frac{2}{(x+10)^3}$$

【0107】

また、以下に示すような ∂ 及び ∂^2 が得られる。

10

20

30

40

【数58】

$$\Lambda = \begin{bmatrix} 174.6 & -106.5 & 196.3 & -379.8 \\ -106.5 & 172.9 & -54.0 & 190.4 \\ 196.3 & -54.0 & 250.4 & -438.9 \\ -379.8 & 190.4 & -438.9 & 847.0 \end{bmatrix}$$

$$v = \begin{bmatrix} -23.8 \\ 23.5 \\ -22.9 \\ 48.4 \end{bmatrix}$$

10

【0108】

このようにすれば、大量の微粒子物質の発生を抑制させることができる。

【0109】

なお、エンジンの制御だけではなく、例えば自動車の運転支援のための制御アルゴリズムにおいて、ハンドル（ステアリング）の位置とアクセル開度から、自動車の位置と速度を制御する場合にも本実施の形態は適用可能である。例えば、自動車は白線を越えてはいけないため、白線の位置が状態制約となる。自動車のモデルは非線形であるため、区分的線形なモデルに近似することによって、本実施の形態を適用すれば、安全な運転支援が可能となる。

20

【0110】

さらに、空調機で室内の温度を使用電力が最小になるように制御する場合にも、本実施の形態を適用可能である。室内で、ある一定の温度区間外の室温となった時、室内にいる住人は不快に感じるため、それらの境界の温度が状態制約となる。本実施の形態を適用することで、不快温度にすることなく室内の温度の制御ができる。

【0111】

[実施の形態2]

30

ディーゼルエンジンは、EGRバルブとVNTノズルを操作することでMAFとMAPを最適に制御することで、最適な燃焼状態を保ち、有害物質（例えばNOx、PM）の発生を抑制する。

【0112】

EGR及びVNTによってMAF及びMAPを制御する際、過渡運転状態においてMAF状態が最適な状態（目標値）を上回るオーバーシュートを起こすことがあり、NOx発生の原因となっている。図8は、縦軸がMAFを表し、横軸は時間を表し、点線が目標値を表し、実線が実際のMAFの値 x_{maf} を表す。この例でも、オーバーシュートが発生している。また、図8のように、目標値が下がっていく時でも、MAFの状態が目標値を上回った状態を続けることが多くなると、この点にてもNOx発生の要因となる。また、図7に示したように目標値を大きく下回ると、PMが過剰に発生する。

40

【0113】

第1の実施の形態のように状態制約を取り扱うモデル予測制御を行えば、改善が見込まれるが、以下のような問題もある。

【0114】

(a) 目標値が上がっていく運転状態（上りの状態）と下がっていく状態（下りの状態）、変わらない状態（定常状態）で状態制約の意味合いが変わってしまう。

すなわち、上りの状態の時は上限の設定が難しく、下りの時は目標値を上回る状態が続くことが多いため、現在の状態が制約を満たさない場合がある。

(b) 運転状態は運転手に依存するため、状態制約を切り替えて制御することが難しい。

50

(c) 外乱やノイズの影響で状態制約を超えてしまう可能性があり、最適解を求めることができなくなることがある。

【0115】

そこで、本実施の形態では、以下のような状態制約を導入してオーバーシュートやアンダーシュートを抑制する。

【数59】

$$q(x(i)) = \begin{cases} \gamma_1 \left(e^{\gamma_2(x_{\min} - x(i))} - 1 \right) & \text{if } (x(i) < x_{\min}) \\ 0 & \text{if } (x_{\min} \leq x(i) \leq x_{\max}) \\ \gamma_3 \left(e^{\gamma_4(x(i) - x_{\max})} - 1 \right) & \text{if } (x_{\max} < x(i)) \end{cases} \quad 10$$

γ_1 乃至 γ_4 についてはチューニングパラメータである。また、本実施の形態では MAF についての状態制約として採用する。

【0116】

このペナルティ関数は、図9に示すような関数である。 $x(i)$ 、 x_{\min} 及び x_{\max} は、目標値を基準にした値、すなわち $x(i)$ は状態偏差であり、 x_{\min} 及び x_{\max} はそれぞれ目標値から下又は上にどこまでずれても対処しないかを表しており、その範囲を出た場合には、その範囲内に戻ろうとする。図10に示すように、目標値曲線の上側の上限曲線と、下側の下限曲線との間であればペナルティ関数の値は0であるが、それを外れると値が急激に大きくなるペナルティ関数を採用する。 20

【0117】

なお、上で述べたペナルティ関数は一例であり、 $(x - x_{\max})$ や $(x_{\min} - x)$ を e のべき数としていたが、これらを基数として n 乗 (n は 3 以上) するような関数 $\max(x - x_{\max})^n$ 、 $\min(x_{\min} - x)^n$ を採用してもよい。

【0118】

このような状態制約を導入したとしても、基本的に第1の実施の形態で説明したオンライン処理は同じである。但し、上限値を超えた場合の関数と、下限値を超えた場合の関数とでは形が異なるので、それについて x_{\max} についての 1 階偏微分関数及び 2 階偏微分関数を算出する。すなわち、ステップ S13 では、下限側の 1 階偏微分関数 $\min(x)$ 及び 2 階偏微分関数 $\min^2(x)$ と、上限側の 1 階偏微分関数 $\max(x)$ 及び 2 階偏微分関数 $\max^2(x)$ とを生成する。 30

【0119】

そして、 $x_{\max}(k)$ の値に応じて、下限側の 1 階偏微分関数 $\min(x)$ 及び 2 階偏微分関数 $\min^2(x)$ と、上限側の 1 階偏微分関数 $\max(x)$ 及び 2 階偏微分関数 $\max^2(x)$ と、制約なしとを切り替えることになる。

【0120】

具体的に、図11を用いてモデル予測制御部220の処理内容を説明する。

【0121】

まず、モデル予測制御部220は、 $x(k) (= x_{\text{ref}}(k) - y(k))$ を取得する (図11:ステップ S41)。

【0122】

そうすると、モデル予測制御部220は、関数ベクトル b_2 に $x(k)$ を代入して関数ベクトル b_2 を数値ベクトル化する (ステップ S43)。

【0123】

また、微分値算出部2201は、 $x(k)$ がペナルティ関数の適用範囲のうち下限値未満についての適用範囲 0 であるか、上限値を超える範囲についての適用範囲 1 であるか、それ以外の範囲である適用範囲 2 であるかを判断する (ステップ S45)。

【0124】

適用範囲 1 であれば、微分値算出部 2201 は、 $x^{base}(k)$ ($x^{base_{maf}}(k)$) であれば (56) 式) を算出し、データ格納部 2205 に格納されているペナルティ関数の 1 階偏微分関数 $\min(x)$ と、2 階偏微分関数 $^2\min(x)$ に $x^{base}(k)$ を代入し、 $\min(x^{base}(k))$ 及び $^2\min(x^{base}(k))$ を求める (ステップ S 47)。

【0125】

そして、行列算出部 2202 は、範囲 1 についての M_3 及び b_3 を算出する (ステップ S 49)。

【0126】

(65) 式の左辺におけるベクトル U_2 以外の部分である ($M_2 + ^2\min(x^{base}(k))$) が M_3 である。また、(65) 式の右辺が b_3 である。すなわち、 $b_3 = b_2 - ((\min(x^{base}(k)) - ^2\min(x^{base}(k)) d(x^{base}(k)))$ である。 b_3 を算出するためには、 $d(x^{base}(k))$ を、 $Bu(k-1)$ で算出する。そして処理はステップ S 57 に移行する。 10

【0127】

一方、適用範囲 0 であれば、微分値算出部 2201 は、 $x^{base}(k)$ ($x^{base_{maf}}(k)$) であれば (56) 式) を算出し、データ格納部 2205 に格納されているペナルティ関数の 1 階偏微分関数 $\max(x)$ と、2 階偏微分関数 $^2\max(x)$ に $x^{base}(k)$ を代入し、 $\max(x^{base}(k))$ 及び $^2\max(x^{base}(k))$ を求める (ステップ S 51)。

【0128】

そして、行列算出部 2202 は、範囲 0 についての M_3 及び b_3 を算出する (ステップ S 53)。 20

【0129】

(65) 式の左辺におけるベクトル U_2 以外の部分である ($M_2 + ^2\max(x^{base}(k))$) が M_3 である。また、(65) 式の右辺が b_3 である。すなわち、 $b_3 = b_2 - ((\max(x^{base}(k)) - ^2\max(x^{base}(k)) d(x^{base}(k)))$ である。 b_3 を算出するためには、 $d(x^{base}(k))$ を、 $Bu(k-1)$ で算出する。そして処理はステップ S 57 に移行する。

【0130】

さらに、適用範囲 2 であれば、微分値算出部 2201 は、 $M_3 = M_2$ 及び $b_3 = b_2$ と設定する (ステップ S 55)。状態制約についてのペナルティ関数は適用されない。その後処理はステップ S 57 に移行する。 30

【0131】

そして、出力生成部 2203 は、 $U = M^{-1} b_3$ によってベクトル U を算出する (ステップ S 57)。さらに、出力生成部 2203 は、ベクトル U から線形化制御入力 $u(k)$ ($i = 0$) を抽出し、出力する (ステップ S 59)。

【0132】

その後、モデル予測制御部 220 は、処理が終了するのかを判断し (ステップ S 61)、処理終了でなければ処理はステップ S 41 に戻り、処理終了であれば処理は終了する。

【0133】

このような処理を実行すれば、適切なペナルティ関数が適用されるようになり、オーバーシュートやアンダーシュートを効果的に抑制できるようになる。 40

【0134】

[実施の形態 3]

一般に MAF と MAP の追従性にはトレードオフの関係があるため、MAF の追従性を上げるために MAP は悪化するという現象が発生する。そこで、以下のような制御を行うと効果的である。

【0135】

(A) MAP がある程度目標値に近い場合は、できるだけ MAF を近づけたいので、MAF のペナルティ関数を加える。

(B) MAF はある程度目標値に近いが MAP が目標値から大きく離れている場合、MA 50

F にペナルティ関数を加えて得られる効果より M A P の悪化が大きくなることが多いので、ペナルティ関数を加えない。

(C) M A F 及び M A P ともに目標値から離れている場合には、M A F 重視であっても M A P もある程度追従していかなければ N O x 及び P M は減らないし、M A F 及び M A P ともに大きく変化させようとする場合、入力のバルブが全開もしくは全閉になることが多く、ペナルティ関数を入れたとしても入力制約の影響で効果が出ないことが多い。従って、計算量の観点もあり、ペナルティ関数を加えない。

【 0 1 3 6 】

そこで、図 1 2 に示すような領域分割を行って、M A F 及び M A P がいずれの領域に入っているか否かを判断することで、制御を切り替えるものとする。図 1 2 の例では、縦軸は M A P の偏差 x_{map} を表し、横軸は M A F の偏差 x_{maf} を表す。そして、M A P の基準値 A 及び B を設け、 x_{max} と M A P の基準値 A 及び B で囲まれる範囲を領域 0 と設定して M A F の状態制約についての上限側のペナルティ関数を適用する。また、 x_{min} と M A P の基準値 A 及び B で囲まれる範囲を領域 1 として設定して M A F の状態制約についての下限側のペナルティ関数を適用する。それ以外の領域については、領域 2 として設定し、ペナルティ関数を適用しない。

【 0 1 3 7 】

このような領域設定を行う場合には、第 2 の実施の形態における処理フローのステップ S 4 5 において、M A F 及び M A P の状態偏差 $x(k)$ に基づき、いずれの領域に $x(k)$ が入っているかを判定するようとする。後の処理については図 1 1 と同様である。

【 0 1 3 8 】

このような制御の切り替えを行った結果の効果を、図 1 3 A 及び図 1 3 B に示す。図 1 3 A は、M A F の値の時間変化を表しており、図 1 3 B は、M A P の値の時間変化を表している。M A P についてはあまり変化していないが、M A F については、部分 P ではオーバーシュートが低減されており、部分 Q では目標値への追従性が向上している。これによって、N O x 及び P M 排出量のさらなる低減がなされる。

【 0 1 3 9 】

以上本技術の実施の形態を説明したが、本技術はこれに限定されるものではない。例えば、上で述べた機能ブロック構成については、プログラムモジュール構成とは一致しない場合もある。また、処理結果が変わらない限り、処理フローについても、ステップの順番を入れ替えたり、複数のステップを並列実行するようにしてもよい。

【 0 1 4 0 】

なお、上で述べたオフライン処理装置 1 0 0 は、コンピュータ装置であって、図 1 4 に示すように、メモリ 2 5 0 1 と C P U 2 5 0 3 とハードディスク・ドライブ (H D D) 2 5 0 5 と表示装置 2 5 0 9 に接続される表示制御部 2 5 0 7 とリムーバブル・ディスク 2 5 1 1 用のドライブ装置 2 5 1 3 と入力装置 2 5 1 5 とネットワークに接続するための通信制御部 2 5 1 7 とがバス 2 5 1 9 で接続されている。オペレーティング・システム (O S : Operating System) 及び本実施例における処理を実施するためのアプリケーション・プログラムは、H D D 2 5 0 5 に格納されており、C P U 2 5 0 3 により実行される際には H D D 2 5 0 5 からメモリ 2 5 0 1 に読み出される。C P U 2 5 0 3 は、アプリケーション・プログラムの処理内容に応じて表示制御部 2 5 0 7 、通信制御部 2 5 1 7 、ドライブ装置 2 5 1 3 を制御して、所定の動作を行わせる。また、処理途中のデータについては、主としてメモリ 2 5 0 1 に格納されるが、H D D 2 5 0 5 に格納されるようにしてもよい。本技術の実施例では、上で述べた処理を実施するためのアプリケーション・プログラムはコンピュータ読み取り可能リムーバブル・ディスク 2 5 1 1 に格納されて頒布され、ドライブ装置 2 5 1 3 から H D D 2 5 0 5 にインストールされる。インターネットなどのネットワーク及び通信制御部 2 5 1 7 を経由して、H D D 2 5 0 5 にインストールされる場合もある。このようなコンピュータ装置は、上で述べた C P U 2 5 0 3 、メモリ 2 5 0 1 などのハードウェアと O S 及びアプリケーション・プログラムなどのプログラムとが有機的に協働することにより、上で述べたような各種機能を実現する。

10

20

30

40

50

【0141】

また、上で述べたエンジン制御装置1000は、コンピュータ装置であって、図15に示すように、RAM(Random Access Memory)4501とプロセッサ4503とROM(Read Only Memory)4507とセンサ群4515とがバス4519で接続されている。本実施の形態における処理を実施するための制御プログラム及び存在している場合にはオペレーティング・システム(OS:Operating System)は、ROM4507に格納されており、プロセッサ4503により実行される際にはROM4507からRAM4501に読み出される。プロセッサ4503は、センサ群4515(例えば、吸気圧センサ及び新気量センサ。場合によっては燃料噴射量測定部及びエンジン回転数測定部など。)を制御して、測定値を取得する。また、処理途中のデータについては、RAM4501に格納される。なお、プロセッサ4503は、ROM4507を含む場合もあり、さらに、RAM4501を含む場合もある。本技術の実施の形態では、上で述べた処理を実施するための制御プログラムは、コンピュータ読み取り可能リムーバブル・ディスクに格納されて頒布され、ROMライタによってROM4507に書き込まれる場合もある。このようなコンピュータ装置は、上で述べたプロセッサ4503、RAM4501、ROM4507などのハードウェアと制御プログラム(場合によってはOSも)とが有機的に協働することにより、上で述べたような各種機能を実現する。

【0142】

以上述べた本実施の形態をまとめると以下のようになる。

【0143】

本実施の形態の第1の態様に係るオフライン処理方法は、(A)(a)モデル予測制御の評価関数とラグランジュ乗数を含み且つ制御対象の状態方程式に関する第1の関数との和を含む第2の関数をラグランジュ乗数について偏微分することで得られ且つ制御対象の状態偏差及び制御対象に対する線形化制御入力を変数として含む第1の一次式群と、(b)第2の関数を制御対象の状態偏差について偏微分することで得られ且つ制御対象の状態偏差及びラグランジュ乗数を変数として含む第2の一次式群と、(c)第2の関数を制御対象の線形化制御入力について偏微分することで得られ且つラグランジュ乗数及び制御対象の状態偏差を変数として含む第3の一次式群とを格納するデータ格納部から第1の一次式群を読み出して、状態偏差の現在値を除き状態偏差の変数を消し込むように順次代入を行って、第1の一次式群を、状態偏差の現在値と線形化制御入力とを変数として含む第4の一次式群に変形し、(B)データ格納部から第2の一次式群を読み出して、ラグランジュ乗数の変数を消し込むように順次代入を行って、第2の一次式群を、状態偏差を変数として含む第5の一次式群に変形し、(C)データ格納部から第3の一次式群を読み出して、第4の一次式群及び第5の一次式群を用いて、第3の一次式群を、状態偏差の現在値と線形化制御入力とを変数として含む第6の一次式群に変形し、(D)第6の一次式群を、第1の係数行列と線形化制御入力についてのベクトルとの積が、状態偏差の現在値の関数ベクトルと等式で表されるように変形し、(E)モデル予測制御の制御対象における状態制約に対応するペナルティ関数を状態偏差について1階偏微分した第1の微分関数及び2階偏微分した第2の微分関数を生成し、(F)第4の一次式群から線形化制御入力についての係数を抽出し、ペナルティ関数について予め定められている第2の係数行列及び係数ベクトルを算出する処理を含む。

【0144】

このような処理をオフラインで実行しておくことにより、状態制約を導入したモデル予測制御を行うことができるようになる。

【0145】

上で述べたペナルティ関数が、状態偏差が第1の値以上第2の値以下で0となり、状態偏差が第1の値より小さい場合及び第2の値より大きい場合状態偏差の値に応じて指數関数的に増加する関数である場合もある。制御対象の特性に応じた適切な関数が用いられる。

【0146】

10

20

30

40

50

また、上で述べたペナルティ関数が、状態偏差が第1の値以上第2の値以下で0となり、状態偏差が第1の値より小さい場合に状態偏差の値に応じて指数関数的に増加する第1の関数と、第2の値より大きい場合状態偏差の値に応じて指数関数的に増加する第2の関数とで表される場合もある。この場合、第1及び第2の微分関数を、第1の関数及び第2の関数の各々について生成する場合もある。このようにすれば、制御実行時に第1の関数と第2の関数とを切り替えて用いることができるようになる。

【0147】

本実施の形態の第2の態様に係る制御方法は、(A)モデル予測制御の制御対象についての状態偏差の現在値についての関数ベクトルと、制御期間内における制御対象に対する線形化制御入力のベクトルとの積が関数ベクトルと等しくなるように予め算出されている第1の係数行列と、モデル予測制御の制御対象における状態制約に対応するペナルティ関数について予め算出されている第2の係数行列及び係数ベクトルと、ペナルティ関数を状態偏差について1階偏微分した第1の式及び2階偏微分した第2の式とを格納するデータ格納部から第1の式及び第2の式を読み出して、状態偏差の現在値から得られる基点の値を代入することで、第1の値及び第2の値を算出し、(B)データ格納部から第1の係数行列及び第2の係数行列を読み出して、第2の値と第2の係数行列との積と第1の係数行列との和である第1の行列を算出し、(C)データ格納部から関数ベクトルを読み出して、関数ベクトルに状態偏差の現在値を代入することで得られる第1のベクトルと、第1の値と第2の値との差に線形化制御入力の1単位時間前の値から得られる第3の値を乗じた結果と係数ベクトルとの積との差である第2のベクトルを算出し、(D)第1の行列の逆行列と第2のベクトルとの積により、線形化制御入力のベクトルを生成し、(E)線形化制御入力のベクトルの要素のうち最初の時刻のための線形化制御入力を抽出する処理を含む。

10

【0148】

このような処理を実行することで、高速に状態制約を伴うモデル予測制御を実行できるようになる。

【0149】

上で述べた制御方法は、(F)状態偏差の現在値が所定範囲内であるか判断する処理をさらに含むようにしてもよい。この場合、状態偏差の現在値が所定範囲内であれば、第1の値及び第2の値を算出する処理と、第1の行列を算出する処理と、第2のベクトルを算出する処理と、線形化制御入力のベクトルを生成する処理と、線形化制御入力を抽出する処理とを実行するようにしてもよい。制御対象によっては、状態制約に基づく処理を行うことが好ましくない場合もあるためである。

30

【0150】

さらに、上で述べたデータ格納部が、状態偏差の現在値の値域に応じた第1の式及び第2の式を格納している場合もある。この場合、(a1)上記第1の値及び第2の値を算出する処理が、状態偏差の現在値が属する値域を特定し、当該特定された値域に応じた第1の式及び第2の式を読み出す処理を含むようにしてもよい。適切な式を状況に応じて切り分けて用いることで、適切な制御を行うことができるようになる。

【0151】

また、本制御方法が、上で述べた状態偏差の現在値が所定範囲外であれば、(G)データ格納部から関数ベクトル及び第1の係数行列を読み出して、関数ベクトルに状態偏差の現在値を代入することで得られる第1のベクトルと、第1の係数行列の逆行列との積により、線形化制御入力の第2のベクトルを生成し、(H)線形化制御入力の第2のベクトルの要素のうち最初の時刻のための線形化制御入力を抽出する処理をさらに含むようにしてもよい。状態制約を考慮しないモデル予測制御を高速に実行できる。

40

【0152】

なお、上記方法による処理をプロセッサ(又はコンピュータ)に行わせるためのプログラムを作成することができ、当該プログラムは、例えばフレキシブルディスク、CD-ROM、光磁気ディスク、半導体メモリ、ハードディスク等のコンピュータ読み取り可能な

50

記憶媒体又は記憶装置に格納される。尚、中間的な処理結果はメインメモリ等の記憶装置に一時保管される。

【0153】

以上の実施例を含む実施形態に関し、さらに以下の付記を開示する。

【0154】

(付記1)

(a) モデル予測制御の評価関数とラグランジュ乗数を含み且つ前記制御対象の状態方程式に関する第1の関数との和を含む第2の関数を前記ラグランジュ乗数について偏微分することで得られ且つ前記制御対象の状態偏差及び前記制御対象に対する線形化制御入力を変数として含む第1の一次式群と、(b) 前記第2の関数を前記制御対象の状態偏差について偏微分することで得られ且つ前記制御対象の状態偏差及び前記ラグランジュ乗数を変数として含む第2の一次式群と、(c) 前記第2の関数を前記制御対象の線形化制御入力について偏微分することで得られ且つ前記ラグランジュ乗数及び前記制御対象の状態偏差を変数として含む第3の一次式群とを格納するデータ格納部から前記第1の一次式群を読み出して、前記状態偏差の現在値を除き前記状態偏差の変数を消し込むように順次代入を行って、前記第1の一次式群を、前記状態偏差の現在値と前記線形化制御入力を変数として含む第4の一次式群に変形し、

前記データ格納部から前記第2の一次式群を読み出して、前記ラグランジュ乗数の変数を消し込むように順次代入を行って、前記第2の一次式群を、前記状態偏差を変数として含む第5の一次式群に変形し、

前記データ格納部から前記第3の一次式群を読み出して、前記第4の一次式群及び前記第5の一次式群を代入することで、前記第3の一次式群を、前記状態偏差の現在値と前記線形化制御入力を変数として含む第6の一次式群に変形し、

前記第6の一次式群を、第1の係数行列と前記線形化制御入力についてのベクトルとの積が、前記状態偏差の現在値の関数ベクトルと等式で表されるように変形し、

前記モデル予測制御の制御対象についての制約条件に対応するペナルティ関数を前記状態偏差について1階偏微分した第1の微分関数及び2階偏微分した第2の微分関数を生成し、

前記第4の一次式群から前記線形化制御入力についての係数を抽出し、前記ペナルティ関数について予め定められている第2の係数行列及び係数ベクトルを算出する

処理を、コンピュータに実行させるためのプログラム。

【0155】

(付記2)

前記ペナルティ関数が、前記状態偏差が第1の値以上第2の値以下で0となり、前記状態偏差が前記第1の値より小さい場合及び前記第2の値より大きい場合前記状態偏差の値に応じて指數関数的に増加する関数である

付記1記載のプログラム。

【0156】

(付記3)

前記ペナルティ関数が、前記状態偏差が第1の値以上第2の値以下で0となり、前記状態偏差が前記第1の値より小さい場合に前記状態偏差の値に応じて指數関数的に増加する第1の関数と、前記第2の値より大きい場合前記状態偏差の値に応じて指數関数的に増加する第2の関数とで表され、

前記第1及び第2の微分関数を、前記第1の関数及び前記第2の関数の各々について生成する

付記1記載のプログラム。

【0157】

(付記4)

モデル予測制御の制御対象についての状態偏差の現在値についての関数ベクトルと、制御期間内における前記制御対象に対する線形化制御入力のベクトルとの積が前記関数ベク

10

20

30

40

50

トルと等しくなるように予め算出されている第1の係数行列と、前記モデル予測制御の制御対象における状態制約に対応するペナルティ関数について予め算出されている第2の係数行列及び係数ベクトルと、前記ペナルティ関数を前記状態偏差について1階偏微分した第1の式及び2階偏微分した第2の式とを格納するデータ格納部から前記第1の式及び第2の式を読み出して、前記状態偏差の現在値から得られる基点の値を代入することで、第1の値及び第2の値を算出し、

前記データ格納部から前記第1の係数行列及び前記第2の係数行列を読み出して、前記第2の値と前記第2の係数行列との積と前記第1の係数行列との和である第1の行列を算出し、

前記データ格納部から前記関数ベクトルを読み出して、前記関数ベクトルに前記状態偏差の現在値を代入することで得られる第1のベクトルと、前記第1の値と前記第2の値との差に前記線形化制御入力の1単位時間前の値から得られる第3の値を乗じた結果と前記係数ベクトルとの積との和である第2のベクトルを算出し、

前記第1の行列の逆行列と前記第2のベクトルとの積により、前記線形化制御入力のベクトルを生成し、

前記線形化制御入力のベクトルの要素のうち最初の時刻のための線形化制御入力を抽出する

処理をプロセッサに実行させるためのプログラム。

【0158】

(付記5)

前記状態偏差の現在値が所定範囲内であるか判断する処理をさらに含み、

前記状態偏差の現在値が前記所定範囲内であれば、前記第1の値及び第2の値を算出する処理と、前記第1の行列を算出する処理と、前記第2のベクトルを算出する処理と、前記線形化制御入力のベクトルを生成する処理と、前記線形化制御入力を抽出する処理とを実行する

付記4記載のプログラム。

【0159】

(付記6)

前記データ格納部が、

前記状態偏差の現在値の値域に応じた前記第1の式及び第2の式を格納しており、

前記第1の値及び前記第2の値を算出する処理が、

前記状態偏差の現在値が属する値域を特定し、当該特定された値域に応じた前記第1の式及び前記第2の式を読み出す処理

を含む付記4又は5記載のプログラム。

【0160】

(付記7)

前記状態偏差の現在値が前記所定範囲外であれば、

前記データ格納部から前記関数ベクトル及び前記第1の係数行列を読み出して、前記関数ベクトルに前記状態偏差の現在値を代入することで得られる第1のベクトルと、前記第1の係数行列の逆行列との積により、前記線形化制御入力の第2のベクトルを生成し、

前記線形化制御入力の第2のベクトルの要素のうち最初の時刻のための線形化制御入力を抽出する

処理をさらに含む付記3乃至5のいずれか1つ記載のプログラム。

【0161】

(付記8)

(a) モデル予測制御の評価関数とラグランジュ乗数を含み且つ前記制御対象の状態方程式に関する第1の関数との和を含む第2の関数を前記ラグランジュ乗数について偏微分することで得られ且つ前記制御対象の状態偏差及び前記制御対象に対する線形化制御入力を変数として含む第1の一次式群と、(b) 前記第2の関数を前記制御対象の状態偏差について偏微分することで得られ且つ前記制御対象の状態偏差及び前記ラグランジュ乗数を

10

20

30

40

50

変数として含む第2の一次式群と、(c)前記第2の関数を前記制御対象の線形化制御入力について偏微分することで得られ且つ前記ラグランジュ乗数及び前記制御対象の状態偏差を変数として含む第3の一次式群とを格納するデータ格納部から前記第1の一次式群を読み出して、前記状態偏差の現在値を除き前記状態偏差の変数を消し込むように順次代入を行って、前記第1の一次式群を、前記状態偏差の現在値と前記線形化制御入力とを変数として含む第4の一次式群に変形し、

前記データ格納部から前記第2の一次式群を読み出して、前記ラグランジュ乗数の変数を消し込むように順次代入を行って、前記第2の一次式群を、前記状態偏差を変数として含む第5の一次式群に変形し、

前記データ格納部から前記第3の一次式群を読み出して、前記第4の一次式群及び前記第5の一次式群を代入することで、前記第3の一次式群を、前記状態偏差の現在値と前記線形化制御入力とを変数として含む第6の一次式群に変形し、

前記第6の一次式群を、第1の係数行列と前記線形化制御入力についてのベクトルとの積が、前記状態偏差の現在値の関数ベクトルと等式で表されるように変形し、

前記モデル予測制御の制御対象についての制約条件に対応するペナルティ関数を前記状態偏差について1階偏微分した第1の微分関数及び2階偏微分した第2の微分関数を生成し、

前記第4の一次式群から前記線形化制御入力についての係数を抽出し、前記ペナルティ関数について予め定められている第2の係数行列及び係数ベクトルを算出する

処理を含み、コンピュータにより実行されるオフライン処理方法。

【0162】

(付記9)

モデル予測制御の制御対象についての状態偏差の現在値についての関数ベクトルと、制御期間内における前記制御対象に対する線形化制御入力のベクトルとの積が前記関数ベクトルと等しくなるように予め算出されている第1の係数行列と、前記モデル予測制御の制御対象における状態制約に対応するペナルティ関数について予め算出されている第2の係数行列及び係数ベクトルと、前記ペナルティ関数を前記状態偏差について1階偏微分した第1の式及び2階偏微分した第2の式とを格納するデータ格納部から前記第1の式及び第2の式を読み出して、前記状態偏差の現在値から得られる基点の値を代入することで、第1の値及び第2の値を算出し、

前記データ格納部から前記第1の係数行列及び前記第2の係数行列を読み出して、前記第2の値と前記第2の係数行列との積と前記第1の係数行列との和である第1の行列を算出し、

前記データ格納部から前記関数ベクトルを読み出して、前記関数ベクトルに前記状態偏差の現在値を代入することで得られる第1のベクトルと、前記第1の値と前記第2の値との差に前記線形化制御入力の1単位時間前の値から得られる第3の値を乗じた結果と前記係数ベクトルとの積との和である第2のベクトルを算出し、

前記第1の行列の逆行列と前記第2のベクトルとの積により、前記線形化制御入力のベクトルを生成し、

前記線形化制御入力のベクトルの要素のうち最初の時刻のための線形化制御入力を抽出する

処理を含み、プロセッサにより実行される制御方法。

【0163】

(付記10)

(a)モデル予測制御の評価関数とラグランジュ乗数を含み且つ前記制御対象の状態方程式に関する第1の関数との和を含む第2の関数を前記ラグランジュ乗数について偏微分することで得られ且つ前記制御対象の状態偏差及び前記制御対象に対する線形化制御入力を変数として含む第1の一次式群と、(b)前記第2の関数を前記制御対象の状態偏差について偏微分することで得られ且つ前記制御対象の状態偏差及び前記ラグランジュ乗数を変数として含む第2の一次式群と、(c)前記第2の関数を前記制御対象の線形化制御入

10

20

30

40

50

力について偏微分することで得られ且つ前記ラグランジュ乗数及び前記制御対象の状態偏差を変数として含む第3の一次式群とを格納するデータ格納部から前記第1の一次式群を読み出して、前記状態偏差の現在値を除き前記状態偏差の変数を消し込むように順次代入を行って、前記第1の一次式群を、前記状態偏差の現在値と前記線形化制御入力とを変数として含む第4の一次式群に変形する手段と、

前記データ格納部から前記第2の一次式群を読み出して、前記ラグランジュ乗数の変数を消し込むように順次代入を行って、前記第2の一次式群を、前記状態偏差を変数として含む第5の一次式群に変形する手段と、

前記データ格納部から前記第3の一次式群を読み出して、前記第4の一次式群及び前記第5の一次式群を代入することで、前記第3の一次式群を、前記状態偏差の現在値と前記線形化制御入力とを変数として含む第6の一次式群に変形する手段と、

前記第6の一次式群を、第1の係数行列と前記線形化制御入力についてのベクトルとの積が、前記状態偏差の現在値の関数ベクトルと等式で表されるように変形する手段と、

前記モデル予測制御の制御対象についての制約条件に対応するペナルティ関数を前記状態偏差について1階偏微分した第1の微分関数及び2階偏微分した第2の微分関数を生成する手段と、

前記第4の一次式群から前記線形化制御入力についての係数を抽出し、前記ペナルティ関数について予め定められている第2の係数行列及び係数ベクトルを算出する手段と、

を有するオフライン処理装置。

【0164】

(付記11)

モデル予測制御の制御対象についての状態偏差の現在値についての関数ベクトルと、制御期間内における前記制御対象に対する線形化制御入力のベクトルとの積が前記関数ベクトルと等しくなるように予め算出されている第1の係数行列と、前記モデル予測制御の制御対象における状態制約に対応するペナルティ関数について予め算出されている第2の係数行列及び係数ベクトルと、前記ペナルティ関数を前記状態偏差について1階偏微分した第1の式及び2階偏微分した第2の式とを格納するデータ格納部から前記第1の式及び第2の式を読み出して、前記状態偏差の現在値から得られる基点の値を代入することで、第1の値及び第2の値を算出する第1の算出部と、

前記データ格納部から前記第1の係数行列及び前記第2の係数行列を読み出して、前記第2の値と前記第2の係数行列との積と前記第1の係数行列との和である第1の行列を算出し、前記データ格納部から前記関数ベクトルを読み出して、前記関数ベクトルに前記状態偏差の現在値を代入することで得られる第1のベクトルと、前記第1の値と前記第2の値との差に前記線形化制御入力の1単位時間前の値から得られる第3の値を乗じた結果と前記係数ベクトルとの積との和である第2のベクトルを算出する第2の算出部と、

前記第1の行列の逆行列と前記第2のベクトルとの積により、前記線形化制御入力のベクトルを生成する生成部と、

前記線形化制御入力のベクトルの要素のうち最初の時刻のための線形化制御入力を出力する出力部と、

を有する制御装置。

【符号の説明】

【0165】

101 入力部

102 入力データ格納部

103 第1変形処理部

104 第1データ格納部

105 第2変形処理部

106 第2データ格納部

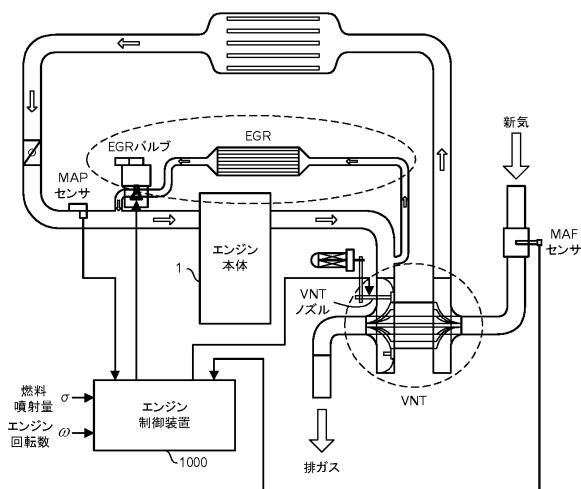
107 第3変形処理部

108 第3データ格納部

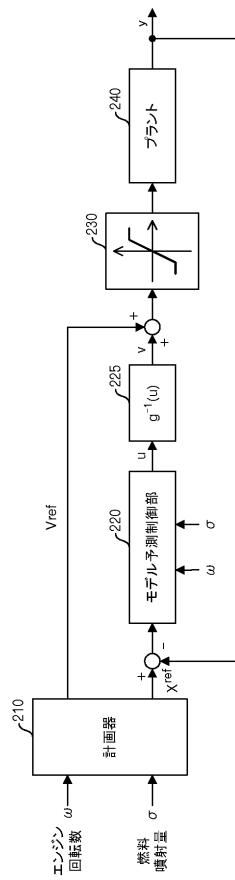
JP 6136494 B2 2017.5.31

- 1 0 9 第1行列生成部
- 1 1 0 第4データ格納部
- 1 1 1 第2行列生成部
- 1 1 2 微分処理部

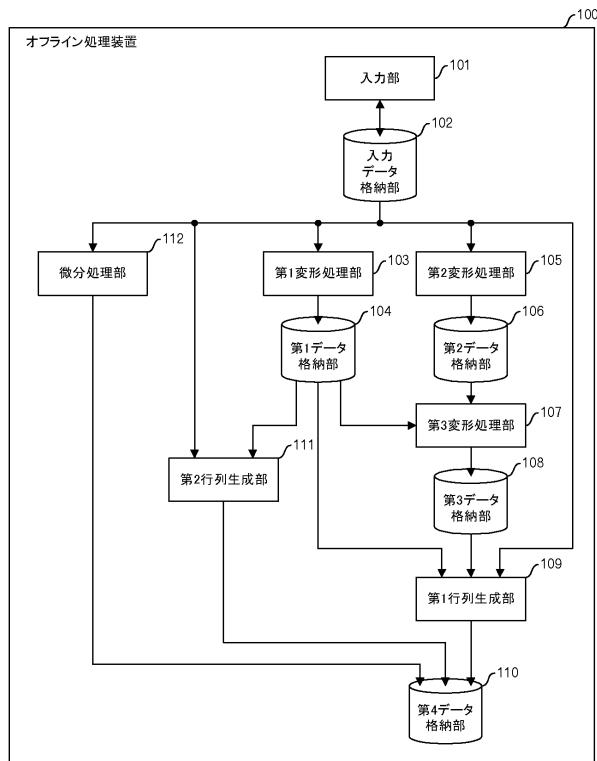
【図1】



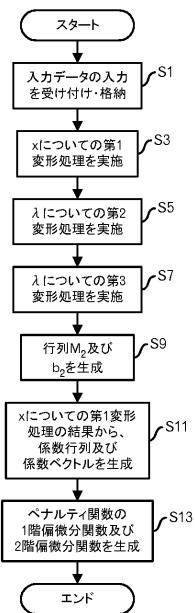
【図2】



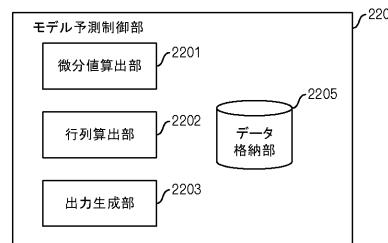
【図3】



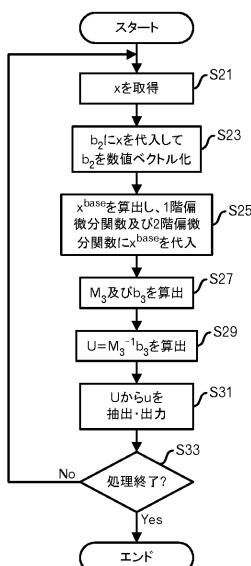
【図4】



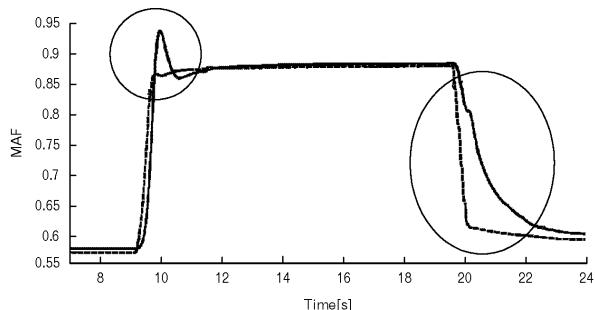
【図5】



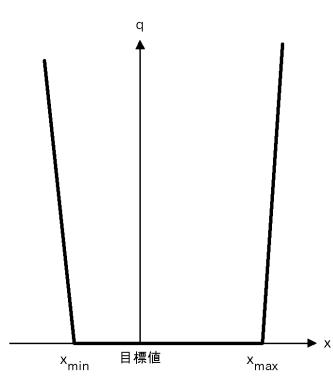
【図6】



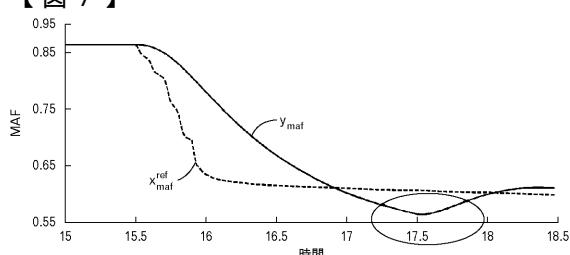
【図8】



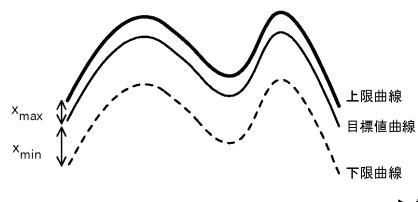
【図9】



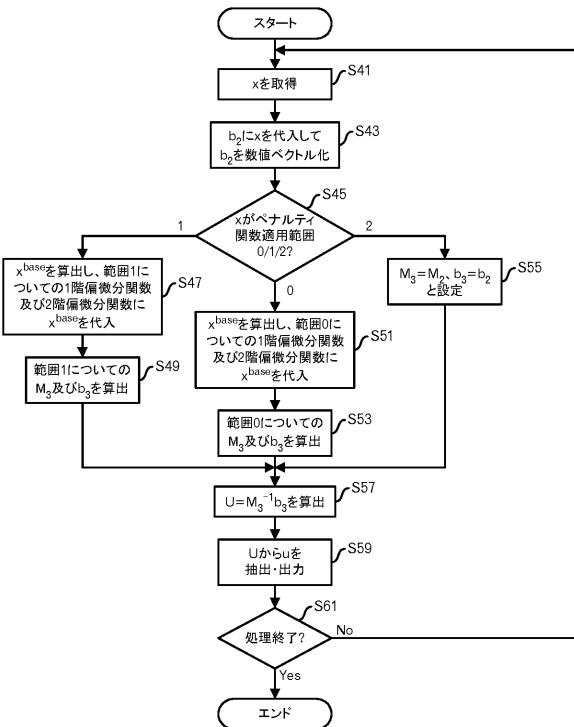
【図7】



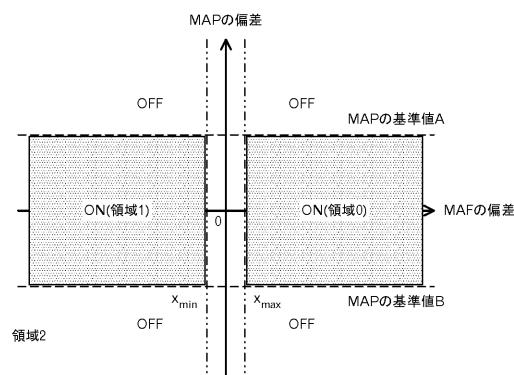
【図10】



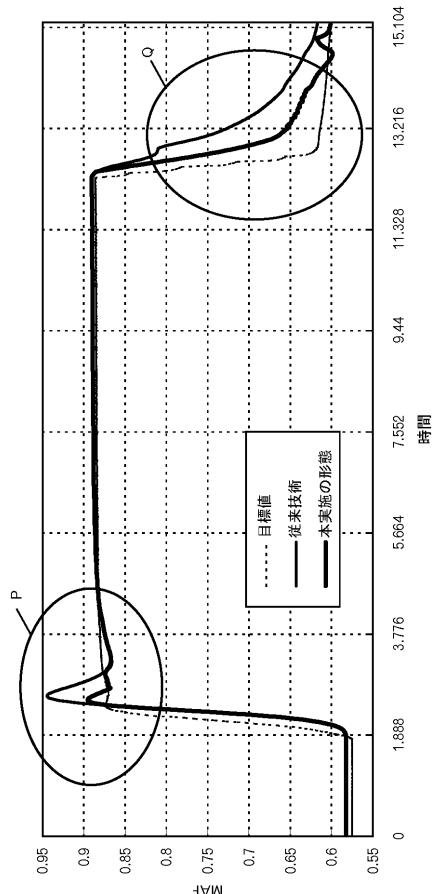
【図11】



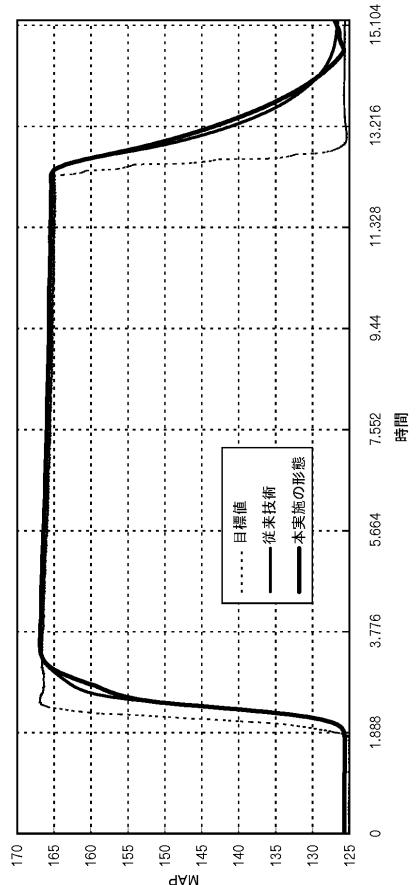
【図12】



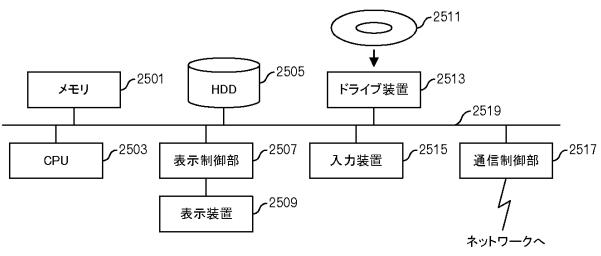
【図13A】



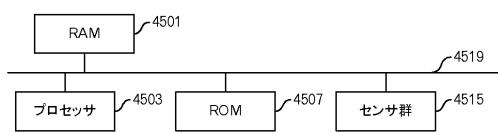
【図13B】



【図14】



【図15】



フロントページの続き

(56)参考文献 特開2012-194960(JP, A)
特開2006-072700(JP, A)
特開2001-067103(JP, A)
特開平09-029057(JP, A)

(58)調査した分野(Int.Cl. , DB名)

G 05 B 13 / 04
F 02 D 41 / 18
F 02 D 45 / 00