

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7577385号
(P7577385)

(45)発行日 令和6年11月5日(2024.11.5)

(24)登録日 令和6年10月25日(2024.10.25)

(51)国際特許分類		F I		
G 0 6 N	20/00 (2019.01)	G 0 6 N	20/00	
B 2 5 J	9/22 (2006.01)	B 2 5 J	9/22	Z
G 0 6 F	16/906 (2019.01)	G 0 6 F	16/906	
G 0 6 F	18/23 (2023.01)	G 0 6 F	18/23	
G 0 6 F	123/02 (2023.01)	G 0 6 F	123:02	
請求項の数 11 (全41頁)				
(21)出願番号	特願2023-542213(P2023-542213)	(73)特許権者	517107634	
(86)(22)出願日	令和4年3月24日(2022.3.24)		S O I N N株式会社	
(86)国際出願番号	PCT/JP2022/014070		東京都町田市鶴間八丁目4番30号	
(87)国際公開番号	WO2023/021776	(74)代理人	100103894	
(87)国際公開日	令和5年2月23日(2023.2.23)		弁理士 家入 健	
審査請求日	令和5年11月20日(2023.11.20)	(72)発明者	長谷川 修	
(31)優先権主張番号	特願2021-133415(P2021-133415)		東京都町田市鶴間八丁目4番30号 S	
(32)優先日	令和3年8月18日(2021.8.18)		O I N N株式会社内	
(33)優先権主張国・地域又は機関	日本国(JP)	(72)発明者	井加田 洸輔	
(出願人による申告) 2018年度、国立研究開発法人新エネルギー・産業技術総合開発機構、「高効率・高速処理を可能とするAIチップ・次世代コンピューティングの技術開発／革新的AIエッジコンピューティング技術の開発／動的再構成技術を活用した組み込みAIシ			東京都町田市鶴間八丁目4番30号 S	
	最終頁に続く	(72)発明者	O I N N株式会社内	
		(72)発明者	津田 直純	
			東京都町田市鶴間八丁目4番30号 S	
			O I N N株式会社内	
		審査官	山本 俊介	
			最終頁に続く	

(54)【発明の名称】 情報処理装置、情報処理方法及びプログラム

(57)【特許請求の範囲】

【請求項1】

学習対象から予め取得した第1の状態量と前記学習対象の動作を制御するために与えた前記第1の状態量に対応する第1の指令値と、を含む、多次元ベクトルで表されたデータ要素を複数含む時系列データに基づいた学習用データのそれぞれを第2の状態量及び第2の指令値を含む多次元ベクトルとして表されるノードとして時系列クラスタリングすることで、前記学習用データの分布構造を、複数のクラスタに分類した前記ノードの集合として学習したモデルを保持するモデル保持部と、

制御対象から取得した第3の状態量が入力データとして入力され、前記モデルに含まれる前記学習用データのデータ要素数よりも少数の前記ノードとして、前記入力データが属するクラスタと前記入力データが属するクラスタに時間的に直後のクラスタとに属するノードを選択し、選択した2つの前記クラスタに属する前記ノードから前記入力データに対して最短距離のノードを、前記入力データに一致又は近似するノードとして探索する探索部と、

前記探索したノードの前記第2の指令値に基づく値を、前記制御対象を操作するために与える出力指令値として前記制御対象へ出力する出力部と、を備える、

情報処理装置。

【請求項2】

前記探索部は、前記探索したノードが属するクラスタの一部又は全部のノードの指令値から算出される統計量に基づいて、前記出力指令値を決定する、

請求項 1 に記載の情報処理装置。

【請求項 3】

前記探索部は、前記探索したノードの指令値を、前記出力指令値として出力する、
請求項 1 に記載の情報処理装置。

【請求項 4】

前記探索部は、前記探索したノードとの指令値と、前記探索したノードに近似する 1 つ以上のノードの指令値と、から算出される統計量に基づいて、前記出力指令値を決定する、
請求項 1 に記載の情報処理装置。

【請求項 5】

前記探索したノードに近似する 1 以上のノードは、前記探索したノードから所定の距離以内のノード、又は、前記探索したノードから近い順に選択された所定の個数のノードである、

請求項 4 に記載の情報処理装置。

【請求項 6】

前記統計量は、平均値、中央値、最大値、最小値及び最頻値のいずれかである、
請求項 4 又は 5 に記載の情報処理装置。

【請求項 7】

前記第 2 の状態量は、前記学習対象から予め取得した前記第 1 の状態量であり、かつ、前記第 2 の指令値は、前記予め取得した前記第 1 の状態量に応じて前記学習対象に与えられた前記第 1 の指令値である、

請求項 1 乃至 6 のいずれか一項に記載の情報処理装置。

【請求項 8】

前記第 1 の指令値は、前記学習対象から予め取得した前記第 1 の状態量に基づいて前記学習対象の操作者が前記学習対象に与えた指令値の実績値である、

請求項 7 に記載の情報処理装置。

【請求項 9】

前記学習用データのデータ要素に含まれる前記第 1 の指令値は、前記学習対象から予め取得した前記第 1 の状態量に対して、所定の時間だけ後に前記制御対象に与えられた指令値である、

請求項 7 に記載の情報処理装置。

【請求項 10】

学習対象から予め取得した第 1 の状態量と前記学習対象の動作を制御するために与えた前記第 1 の状態量に対応する第 1 の指令値と、を含む、多次元ベクトルで表されたデータ要素を複数含む時系列データに基づいた学習用データのそれぞれを第 2 の状態量及び第 2 の指令値を含む多次元ベクトルとして表されるノードとして時系列クラスタリングすることで、前記学習用データの分布構造を、複数のクラスタに分類した前記ノードの集合として学習したモデルを保持し、

制御対象から取得した第 3 の状態量が入力データとして入力され、前記モデルに含まれる前記学習用データのデータ要素数よりも少数の前記ノードとして、前記入力データが属するクラスタと前記入力データが属するクラスタに時間的に直後のクラスタとに属するノードを選択し、選択した 2 つの前記クラスタに属する前記ノードから前記入力データに対して最短距離のノードを、前記入力データに一致又は近似するノードとして探索し、

前記探索したノードの前記第 2 の指令値に基づく値を、前記制御対象を操作するために与える出力指令値として前記制御対象へ出力する、

情報処理方法。

【請求項 11】

学習対象から予め取得した第 1 の状態量と前記学習対象の動作を制御するために与えた前記第 1 の状態量に対応する第 1 の指令値と、を含む、多次元ベクトルで表されたデータ要素を複数含む時系列データに基づいた学習用データのそれぞれを第 2 の状態量及び第 2 の指令値を含む多次元ベクトルとして表されるノードとして時系列クラスタリングするこ

10

20

30

40

50

とで、前記学習用データの分布構造を、複数のクラスタに分類した前記ノードの集合として学習したモデルを保持する処理と、

制御対象から取得した第3の状態量が入力データとして入力され、前記モデルに含まれる前記学習用データのデータ要素数よりも少数の前記ノードとして、前記入力データが属するクラスタと前記入力データが属するクラスタに時間的に直後のクラスタとに属するノードを選択し、選択した2つの前記クラスタに属する前記ノードから前記入力データに対して最短距離のノードを、前記入力データに一致又は近似するノードとして探索する処理と、

前記探索したノードの前記第2の指令値に基づく値を、前記制御対象を操作するために与える出力指令値として前記制御対象へ出力する処理と、をコンピュータに実行させる、プログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、情報処理装置、情報処理方法及びプログラムに関する。

【背景技術】

【0002】

今日、様々なデータ処理手法が用いられており、処理対象となるデータには様々なものがあり、例えば値が系統立って配列されているものが有る。こうした系統立ったデータの代表例として、ある現象を連続的又は間欠的に観測することで得られる一連の値の集合、すなわち現象の時間的に変化を示す値の集合である時系列データが知られている。

【0003】

こうした時系列データの分析手法として、時系列クラスタリングが知られている。時系列クラスタリングとしては、概ねWhole time-series clustering、Subsequence clustering及びTime point clusteringの3つの手法が知られている（非特許文献1及び2）。Whole time-series clusteringでは、複数の時系列データ（時系列集合）ごとに類似性を測定してクラスタリングが行われる。Subsequence clusteringでは、1つの時系列データ（時系列集合）が複数のセグメントに分けられ、セグメントごとにクラスタリングが行われる。Time point clusteringでは、1つの時系列データ（時系列集合）が点ごとに分けられ、各点同士で類似性を測定してクラスタリングが行われる。

【先行技術文献】

【特許文献】

【0004】

【文献】特開2008-217246号公報

【文献】特開2014-164396号公報

【非特許文献】

【0005】

【文献】@ZoneTsuyoshi、「Toeplitz Inverse Covariance-based Clustering ～時系列クラスタリング～」、2018年12月6日、Qiita、2021年2月9日検索、〈URL: <https://qiita.com/ZoneTsuyoshi/items/53dad1d0f0d43512a913>〉

【文献】Hiroshi、「時系列クラスタリングの研究サーベイ論文を読んだ」、2019年7月20日、10001 IDEAS、2021年2月9日検索、〈URL: <https://10001ideas.com/2019/07/20/time-series-survey-paper/>〉

【発明の概要】

【発明が解決しようとする課題】

【0006】

例えば、あるシステムの状態量を取得して、状態量に応じた指令値を与えることで、システムの挙動を制御する手法が広く用いられている。このような制御方式において、制御処理を自動的に行う場合には、任意の時点で取得したシステムの状態量に基づいて好適な指令値を推定し、推定した指令値をシステムに与えることとなる。この場合、予め観測さ

10

20

30

40

50

れた、例えば人が行なった操作履歴（指令値）とシステムの状態量とからなる時系列データである学習用データを機械学習によって学習し、その学習結果に基づいて指令値を推定することが行われる。

【0007】

このとき、例えば、入力データの状態量に対応する指令値を推定するにあたって、k近傍法が知られている。k近傍法は学習データをそのまま使用し、入力データに近いk個の履歴データから操作値を出力させる。出力値は、例えば入力データに近いk個の履歴データの各々における所定時刻経過後の指令値の平均値などにすることができる。k近傍法は、他の教師あり学習に比べると学習を行わないので、膨大なデータを必要とせず、また提供したデータの範囲内の操作となるため、様々な分野で使用されている。しかし、k近傍法は、学習用データのデータ要素の全てと入力データとの間で距離を計算することとなるため、計算量が多くなり、計算に要する時間が長くなってしまいう問題と有する。

10

【0008】

これは、制御対象となるシステムの状態量を取得してから指令値を与えるまでの時間に余裕が有るシステムでは問題にならないものの、例えば車両など、状態量の取得に応じて迅速に指令値を与えなければならないシステムでは、計算時間の長さが迅速な指令値の付与の妨げとなってしまう。

【0009】

そこで、制御対象のシステムの運転状況に応じて、システムの状態量に応じた指令値をより高速に出力することのできる手法の確立が求められている。

20

【0010】

本発明は、上記の事情に鑑みて成されたものであり、制御対象のシステムの運転状況及び状態量に応じた指令値を迅速にシステムに与えることを目的とする。

【課題を解決するための手段】

【0011】

本発明の一態様である情報処理装置は、学習対象から予め取得した第1の状態量と前記学習対象の動作を制御するために与えた前記第1の状態量に対応する第1の指令値と、を含む、多次元ベクトルで表されたデータ要素を複数含む時系列データに基づいた学習用データを学習することで、前記学習用データの分布構造を学習結果に基づいた第2の状態量及び第2の指令値を含む多次元ベクトルとして表されるノードの集合として作成されたモデルを保持するモデル保持部と、制御対象から取得した第3の状態量が入力データとして入力され、前記モデルに含まれる前記学習用データのデータ要素数よりも少数の前記ノードから前記入力データに一致又は近似するノードを探索する探索部と、前記探索したノードの前記第2の指令値に基づく値を、前記制御対象を操作するために与える出力指令値として前記制御対象へ出力する出力部と、を有するものである。これにより、制御対象の運転状況及び状態量に応じた指令値を迅速に制御対象に与えることができる。

30

【0012】

本発明の一態様である情報処理装置は、上記の情報処理装置であって、前記モデルは、前記学習用データのそれぞれをノードとして学習することで作成され、前記探索部は、前記ノードから、前記少数の前記ノードとして、前記入力データに対して時間的に比較的近いノードを選択し、前記選択したノードから前記入力データに対して最短距離のノードを探索することが望ましい。これにより、制御対象の運転状況及び状態量に応じた指令値を迅速に制御対象に与えるにあたり、指令値の推定に用いるモデルを作成し、かつ、最短距離のノードの特定を行うことができる。

40

【0013】

本発明の一態様である情報処理装置は、上記の情報処理装置であって、前記モデルは、前記学習用データを時系列クラスタリングして、前記ノードを複数のクラスタに分類することで作成され、前記探索部は、前記入力データが属するクラスタと、前記入力データが属するクラスタに時間的に直後のクラスタと、を探索し、探索した2つのクラスタに属するノードから前記入力データに対して最短距離のノードを探索することが望ましい。これ

50

により、制御対象の運転状況及び状態量に応じた指令値を迅速に制御対象に与えるにあたり、指令値の推定に用いるモデルを作成し、かつ、最短距離のノードの特定を具体的に行うことができる。

【0014】

本発明の一態様である情報処理装置は、上記の情報処理装置であって、前記探索部は、前記探索したノードが属するクラスタの一部又は全部のノードの指令値から算出される統計量に基づいて、前記出力指令値を決定することが望ましい。これにより、必要に応じて適切な出力指令値を決定することができる。

【0015】

本発明の一態様である情報処理装置は、上記の情報処理装置であって、前記学習用データのデータ要素よりも少数のノードによって前記学習用データのデータ要素の分布構造を近似的に学習することで前記モデルを作成し、作成した前記モデルを前記モデル保持部へ出力するモデル作成部をさらに有し、前記探索部は、前記モデルに含まれる前記ノードから前記入力データに近似するノードを探索することが望ましい。これにより、制御対象の運転状況及び状態量に応じた指令値を迅速に制御対象に与えるにあたり、指令値の推定に用いるモデルを作成し、かつ、最短距離のノードの特定を行うことができる。

【0016】

本発明の一態様である情報処理装置は、上記の情報処理装置であって、前記探索部は、前記探索したノードの指令値を、前記出力指令値として出力することが望ましい。これにより、制御対象の運転状況及び状態量に応じた指令値を迅速に制御対象に与えるにあたり、指令値の推定に用いるモデルを作成し、かつ、最短距離のノードの特定を行うことができる。

【0017】

本発明の一態様である情報処理装置は、上記の情報処理装置であって、前記探索部は、前記探索したノードとの指令値と、前記探索したノードに近似する1つ以上のノードの指令値と、から算出される統計量に基づいて、前記出力指令値を決定することが望ましい。これにより、出力指令値を適切に決定することができる。

【0018】

本発明の一態様である情報処理装置は、上記の情報処理装置であって、前記探索したノードに近似する1以上のノードは、前記探索したノードから所定の距離以内のノード、又は、前記探索したノードから近い順に選択された所定の個数のノードであることが望ましい。これにより、必要に応じた出力指令値を決定することができる。

【0019】

本発明の一態様である情報処理装置は、上記の情報処理装置であって、前記統計量は、平均値、中央値、最大値、最小値及び最頻値のいずれかであることが望ましい。これにより、必要に応じた出力指令値を決定することができる。

【0020】

本発明の一態様である情報処理装置は、上記の情報処理装置であって、前記第2の状態量は、前記学習対象から予め取得した前記第1の状態量であり、かつ、前記第2の指令値は、前記予め取得した前記第1の状態量に応じて前記学習対象に与えられた前記第1の指令値であることが望ましい。これにより、適切なデータに基づいてモデルを作成することができる。

【0021】

本発明の一態様である情報処理装置は、上記の情報処理装置であって、前記第1の指令値は、前記学習対象から予め取得した前記第1の状態量に基づいて前記学習対象の操作者が前記学習対象に与えた指令値の実績値であることが望ましい。これにより、適切なデータに基づいてモデルを作成することができる。

【0022】

本発明の一態様である情報処理装置は、上記の情報処理装置であって、前記学習用データのデータ要素に含まれる前記第1の指令値は、前記学習対象から予め取得した前記第1

10

20

30

40

50

の状態量に対して、所定の時間だけ後に前記制御対象に与えられた指令値であることが望ましい。これにより、状態量の取得と指令値の出力との間のタイムラグを加味したモデルを作成することができる。

【0023】

本発明の一態様である情報処理方法は、学習対象から予め取得した第1の状態量と前記学習対象の動作を制御するために与えた前記第1の状態量に対応する第1の指令値と、を含む、多次元ベクトルで表されたデータ要素を複数含む時系列データに基づいた学習用データを学習することで、前記学習用データの分布構造を学習結果に基づいた第2の状態量及び第2の指令値を含む多次元ベクトルとして表されるノードの集合として作成されたモデルを保持し、制御対象から取得した第3の状態量が入力データとして入力され、前記モデルに含まれる前記学習用データのデータ要素数よりも少数の前記ノードから前記入力データに一致又は近似するノードを探索し、前記探索したノードの前記第2の指令値に基づく値を、前記制御対象を操作するために与える出力指令値として前記制御対象へ出力するものである。これにより、制御対象の運転状況及び状態量に応じた指令値を迅速に制御対象に与えることができる。

10

【0024】

本発明の一態様であるプログラムは、学習対象から予め取得した第1の状態量と前記学習対象の動作を制御するために与えた前記第1の状態量に対応する第1の指令値と、を含む、多次元ベクトルで表されたデータ要素を複数含む時系列データに基づいた学習用データを学習することで、前記学習用データの分布構造を学習結果に基づいた第2の状態量及び第2の指令値を含む多次元ベクトルとして表されるノードの集合として作成されたモデルを保持する処理と、制御対象から取得した第3の状態量が入力データとして入力され、前記モデルに含まれる前記学習用データのデータ要素数よりも少数の前記ノードから前記入力データに一致又は近似するノードを探索する処理と、前記探索したノードの前記第2の指令値に基づく値を、前記制御対象を操作するために与える出力指令値として前記制御対象へ出力する処理と、をコンピュータに実行させるものである。これにより、制御対象の運転状況及び状態量に応じた指令値を迅速に制御対象に与えることができる。

20

【発明の効果】

【0025】

本発明によれば、制御対象のシステムの運転状況及び状態量に応じた指令値を迅速にシステムに与えることができる。

30

【図面の簡単な説明】

【0026】

【図1】実施の形態1にかかる情報処理装置を実現するためのシステム構成の一例を示す図である。

【図2】実施の形態1にかかる情報処理装置の外観構成を示す図である。

【図3】実施の形態1にかかる情報処理装置の構成を模式的に示す図である。

【図4】実施の形態1にかかる情報処理装置が用いられる状況を模式的に示す図である。

【図5】実験に用いるロボットの概要を示す図である。

【図6】ロボットアームが学習する動作の例を示す図である。

40

【図7】ロボットアームが障害物を回避するときの各状態において作用する力を模式的に示す図である。

【図8】時系列データの形式を示す図である。

【図9】ロボットアームの動作を学習させた例を示す図である。

【図10】モデル作成部が作成するノードを示す図である。

【図11】実施の形態1にかかる情報処理装置における処理のフローチャートである。

【図12】実施の形態1にかかる情報処理装置における処理の変形例のフローチャートである。

【図13】実施の形態1にかかる情報処理装置での処理とk近傍法での処理とにおける距離計算の対象となるノードを示す図である。

50

【図 1 4】実施の形態 2 にかかる情報処理装置の構成を模式的に示す図である。
【図 1 5】実施の形態 2 にかかる情報処理装置が用いられる状況を模式的に示す図である。
【図 1 6】実施の形態 2 にかかる情報処理装置における動作を示すフローチャートである。
【図 1 7】実施の形態 2 にかかるモデル作成部の構成を模式的に示す図である。
【図 1 8】S O I N N 法による学習処理のフローチャートである。
【図 1 9】実施の形態 2 にかかる情報処理装置において実施の形態 1 で使用した学習用データを S O I N N 法で学習することで得られたノードの分布の例を示す図である。

【発明を実施するための形態】

【0027】

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。各図面においては、同一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る。

【0028】

実施の形態 1

図 1 は、実施の形態 1 にかかる情報処理装置を実現するためのシステム構成の一例を示す図である。情報処理装置 100 は、専用コンピュータ、パーソナルコンピュータ（PC）などのコンピュータ 1000 により実現可能である。但し、コンピュータは、物理的に単一である必要はなく、分散処理を実行する場合には、複数であってもよい。図 1 に示すように、コンピュータ 1000 は、CPU（Central Processing Unit）1001、ROM（Read Only Memory）1002 及び RAM（Random Access Memory）1003 を有し、これらがバス 1004 を介して相互に接続されている。尚、コンピュータを動作させるための OS ソフトなどは、説明を省略するが、この情報処理装置を構築するコンピュータも当然有しているものとする。

【0029】

バス 1004 には、入出力インターフェイス 1005 も接続されている。入出力インターフェイス 1005 には、例えば、キーボード、マウス、センサなどよりなる入力部 1006、CRT、LCD などよりなるディスプレイ、並びにヘッドフォンやスピーカなどよりなる出力部 1007、ハードディスクなどより構成される記憶部 1008、モデム、ターミナルアダプタなどより構成される通信部 1009 などが接続されている。

【0030】

CPU 1001 は、ROM 1002 に記憶されている各種プログラム、又は記憶部 1008 から RAM 1003 にロードされた各種プログラムに従って各種の処理、本実施の形態においては、例えば後述する情報処理装置 100 の各部の処理を実行する。CPU 1001 とは別に GPU（Graphics Processing Unit）を設け、CPU 1001 と同様の処理を行わせてもよい。なお、GPU は、定型的な処理を並列的に行う用途に適しており、後述する学習処理などに適用することで、CPU 1001 に比べて処理速度を向上させることも可能である。RAM 1003 には又、CPU 1001 及び GPU が各種の処理を実行する上において必要なデータなども適宜記憶される。

【0031】

通信部 1009 は、例えば図示しないインターネットを介しての通信処理を行ったり、CPU 1001 から提供されたデータを送信したり、通信相手から受信したデータを CPU 1001、RAM 1003、記憶部 1008 に出力したりする。記憶部 1008 は CPU 1001 との間でやり取りし、情報の保存・消去を行う。通信部 1009 は又、他の装置との間で、アナログ信号又はデジタル信号の通信処理を行う。

【0032】

入出力インターフェイス 1005 はまた、必要に応じてドライブ 1010 が接続され、例えば、磁気ディスク 1011、光ディスク 1012、フレキシブルディスク 1013、又は半導体メモリ 1014 などが適宜装着され、それらから読み出されたコンピュータプログラムが必要に応じて記憶部 1008 にインストールされる。

【0033】

情報処理装置 100 の外観構成について説明する。図 2 に、実施の形態 1 にかかる情報

10

20

30

40

50

処理装置 100 の外観構成を示す。情報処理装置 100 は、処理部 110、表示部 120 及び入力部 130 を有する。処理部 110 は、上述の CPU 1001、ROM 1002、RAM 1003、バス 1004、入出力インターフェイス 1005、記憶部 1008、通信部 1009、ドライブ 1010 などを有するハードウェアとして構成される。表示部 120 は、上述の出力部 1007 に対応し、オペレータが視認可能な形式で画像を表示する、LCD などの表示装置として構成される。入力部 130 は上述の入力部 1006 に対応し、マウスやキーボードなどの各種の入力手段により構成される。

【0034】

次いで、情報処理装置 100 の学習対象の入力データである時系列データ TSD について説明する。時系列データ TSD は、データ要素を表す多次元ベクトルの集合として与えられる。i をデータ要素の順番を表すパラメータとすると、データ要素を表す多次元ベクトル a_i は、時刻 t_i 及び m 個の成分 p を含むベクトルとして、以下の式のように定義される。ここで、i は 2 の整数であり、m は 1 以上の整数である。

【数 1】

$$a_i = \{t_i, p1_i, p2_i, \dots, pm_i\} \quad [1]$$

よって、データ要素の個数を n とすると、時系列データ TSD は、以下の式で表される。

【数 2】

$$TSD = \{a_1, a_2, \dots, a_n\} \quad [2]$$

【0035】

時系列データ TSD は、例えば情報処理装置 100 に設けられた記憶部（例えば RAM 1003 や記憶部 1008）に格納される。

【0036】

以下で説明する情報処理装置 100 においては機械学習が適用されるが、その前提として、機械学習の概要について説明する。以下で説明する一般的な機械学習は、あくまで以下の実施の形態で説明する制御システムの理解の前提に過ぎず、制御システムに適用される機械学習がこれに限定されるものではない。

【0037】

機械学習は、教師あり学習と教師なし学習とに大別される。以下、それぞれの手法の概要について説明する。

【0038】

一般に、教師あり学習では、与えられた変数（説明変数）からある変数（目的変数）を予測するための学習が行われる。より具体的には、教師あり学習は、入力データ（説明変数）に対して正解データ（目的変数）を与えて、入力データと正解データとの関係性を学習する手法である。

【0039】

例えば、正解データが連続値である場合には回帰分析による学習が行われる。なお、連続データの学習の手法は、回帰分析（例えば、線形回帰）に限られるものではない。回帰分析においては、入力データを各種の関数によってフィッティングすることで、入力データに対応する出力を予測することが可能となる。

【0040】

また、入力データに対する正解データがラベル情報である場合には、分類による学習が行われる。分類による学習においては、例えば、回帰（ロジスティクス回帰、サポートベクターマシン）、木（決定木、ランダムフォレスト）、ニューラルネットワーク、クラス

10

20

30

40

50

タリング（ k 近傍法など）等の手法が用いられる。

【0041】

教師なし学習では、正解データが与えられない状態で、入力データの特徴を学習するものである。教師なし学習の手法としては、 k -means法やSOINN法に代表されるクラスタリング、PCA法などの次元削減、ホテリングの T^2 法などの異常検知などがある。例えば、クラスタリングでは、入力データから近似し特徴を有するものを抽出してグループ化することが可能である。

【0042】

ここで、SOINN法について説明する。SOINN法とは、学習中に必要に応じてニューロンを増殖させる学習手法であり、自己組織化ニューラルネットワーク（SOINN：Self-Organizing Incremental Neural Network）と呼ばれる手法である。SOINNでは、ノード数を自律的に管理することにより非定常的な入力を学習することができ、複雑な分布形状を有するクラスに対しても適切なクラス数及び位相構造を抽出できるなど多くの利点を有する。SOINNの応用例として、例えばパターン認識においては、ひらがな文字のクラスを学習させた後に、カタカナ文字のクラスなどを追加的に学習させることができる。

【0043】

このようなSOINNの一例として、E-SOINN（Enhanced SOINN、特許文献1）と称される手法が提案されている。E-SOINNでは、学習を随時追加するオンライン追加学習が可能であり、バッチ学習ではなく学習効率が良いという利点を有している。このため、E-SOINNでは、学習環境が新しい環境に変化した場合においても追加学習が可能である。また、E-SOINNでは、入力データに対するノイズ耐性が高いという利点をも有している。他にも、LB-SOINN（Load Balance Self-Organizing Incremental Neural Network、特許文献2）などの各種のSOINN法も提案されている。

【0044】

SOINN法では、複数のノードを有するニューラルネットワークが適用される。具体的には、情報処理装置100は、 n （ n は、1以上の整数）次元ベクトルで記述されるノードが配置される非階層構造のニューラルネットワークが入力される。ニューラルネットワークは、例えばRAM1003などの記憶部に格納されている。

【0045】

SOINN法におけるニューラルネットワークは、入力ベクトルをニューラルネットワークに入力し、入力される入力ベクトルに基づいて、ニューラルネットワークに配置されるノード及び辺を自動的に増加させる自己増殖型ニューラルネットワークであり、自己増殖型ニューラルネットワークを用いることで、ノードを自動的に増加させることができる。

【0046】

SOINN法におけるニューラルネットワークは、非階層構造を有するものである。非階層構造を採用することで、他の層での学習を開始するタイミングを指定せずに追加学習を実施することができる。すなわち、オンラインでの追加学習を実施することができる。

【0047】

SOINN法におけるクラスタリングは、ノード及び辺を参照し、クラス分類を行うものである。ネットワークを構成するノードのクラス分類は、各種のクラス分類手法を適用可能であり、例えば特許文献2のLB-SOINNと同様の処理を行ってもよい。

【0048】

以下、実施の形態1にかかる情報処理装置100の構成及び動作について説明する。図3に、実施の形態1にかかる情報処理装置100の構成を模式的に示す。情報処理装置100は、モデル保持部1、探索部2及び出力部3を有する。

【0049】

モデル保持部1は、時系列データである学習用データDATから作成されたモデルMDを読み込んで保持する。モデル保持部1は、例えば、図1のRAM1003及び記憶部1

10

20

30

40

50

008など各種の記憶手段が適用可能である。

【0050】

探索部2は、適宜、モデル保持部1からモデルMDを読み込み可能に構成される。探索部2は、別途入力された推定の対象となるデータ（以下、入力データと称する）の状態量を参照し、読み込んだモデルMDに含まれるノードの中から近似するノードを探索する。そして、探索結果を、出力部3へ出力する。

【0051】

出力部3は、探索結果に基づいて決定した、入力データの状態量に対応する出力指令値を出力する。

【0052】

次いで、情報処理装置100が用いられる状況について説明する。図4に、実施の形態1にかかる情報処理装置100が用いられる状況を模式的に示す。情報処理装置100は、操作者10が操作対象装置20（例えば、後述するロボット）に対して行った操作を学習フェーズにおいて学習し、推定フェーズにおいて学習結果に基づいて推定した指令値を操作対象装置20に出力することで、操作対象装置20に適切な動作を行わせるものである。なお、以下では、操作対象装置20などの情報処理装置100からの制御を受ける対象物を、単に制御対象とも称する。

【0053】

なお、以下では、学習フェーズで取り扱うデータと推定フェーズで取り扱うデータとを区別するため、学習フェーズで取り扱うデータは大文字で表し、推定フェーズで取り扱うデータを小文字で表す。学習フェーズで取り扱う状態量をP、指令値をQと表記する。学習フェーズで取り扱う状態量、すなわち入力データをp、指令値、すなわち出力値をqと表記する。

【0054】

学習フェーズにおいて、操作者10は、適切な動作を指令するための指令値を操作対象装置20に与えるため、指令装置11を操作して指示INSを入力する。指令装置11は、入力された指示INSに応じた指令値Qを、操作対象装置20と、モデル作成部12とへ出力する。操作対象装置20は指令値Qに応じた動作を行うとともに、操作対象装置20はモデル作成部12へ状態量Pを出力する。

【0055】

これにより、モデル作成部12は、ある時刻における、操作対象装置20へ与えられた指令値Qと操作対象装置20の状態量Pとを取得する。指令値Q及び状態量Pの取得は逐次行われるため、モデル作成部12は、指令値Q及び状態量Pのセットと各セットを取得したタイミングを含むデータ要素を複数含む時系列データTSDを得ることができる。なお、以下では、モデル作成のために予め取得される、操作対象装置20へ与えられた指令値Q及び操作対象装置20の状態量Pを、それぞれ第1の指令値及び第1の状態量とも称する。

【0056】

モデル作成部12は、時系列データを適宜加工して学習用データを作成し、学習用データを学習してモデルMDを生成し、モデル保持部1へ出力する。これにより、モデルMDはモデル保持部1によって保持される。すなわち、学習用データは、予め取得された第1の指令値及び第1の状態量からなるデータ要素が複数含まれることとなる。モデル作成部12で作成されるモデルMDは、時系列データの分布構造を、少なくとも状態量及び指令値を要素として含む多次元ベクトルで記述されるノードの集合として学習したモデルである。なお、以下では、モデルMDのノードを記述する多次元ベクトルに含まれる状態量及び指令値を、それぞれ第2の状態量及び第2の指令値とも称する。

【0057】

推定フェーズにおいては、探索部2は、モデル保持部1からモデルMDを適宜読み込む。探索部2は、モデルMDに基づくモデルに入力データ、すなわち、操作対象装置20から取得した状態量pに基づいて、モデルMDに含まれるノードから1つのノードNNEAR

10

20

30

40

50

を決定する。なお、以下では、推定フェーズにおいて操作対象装置 20 から取得される状態量を、第 3 の状態量とも称する。出力部 3 は、ノード N_{NEAR} に対応する指令値を読み込み、読み込んだ指令値を指令値 Q として、操作対象装置 20 へ出力する。本実施の形態では、モデル MD に含まれるノードから 1 つのノード N_{NEAR} を決定したが、モデル MD に含まれるノードから複数のノードを決定してもよい。決定された複数のノードに対応する指令値から、例えば統計的な手法を用いて、出力する指令値を算出してもよい。

【0058】

次に、情報処理装置 100 が行う処理の前提として、モデル作成部 12 が取得した状態量 P 及び指令値 Q で構成される時系列データ TSD と、時系列データ TSD に基づいて作成された学習用データ DAT をモデル作成部 12 が学習することで生成されるモデル MD について説明する。

【0059】

モデル作成部 12 は、学習用データ DAT を読み込んでクラスタリングを行い、学習用データ DAT にクラスタリング情報を追加したものをノードとして学習する。なお、以下で説明するように、学習用データ DAT は状態量 P 、指令値 Q 及びタイミングを示す情報を含む時系列データを適宜加工した多次元ベクトルの集合体の形式になっており、各ノードは同じ多次元ベクトルで構成される。

【0060】

具体例として、ロボットを使用した実験について学習する例について説明する。図 5 に、実験に用いるロボットの概要を示す。ロボット 30 は、ロボットアーム 31 を有しており、ロボットアーム 31 の先端からは円柱ポール 33 が延在しており、ロボットアーム 31 と円柱ポール 33 との接続部には、円柱ポール 33 に作用する力を検出する力センサ 32 が設けられている。これにより、円柱ポール 33 に外力が掛かると、力センサ 32 によってその外力を検出することができる。ロボットアーム 31 は、水平方向に延在するビーム 34 が図示しない多関節駆動機構などによって駆動されることで、水平面内（ x 方向及び y 方向）及び高さ方向（ z 方向）の 3 次元方向に移動することができる。

【0061】

この例では、操作者が指令装置 11 を使用してロボットアーム 31 を移動させることで、ロボットアーム 31 の動作を学習した。但し、ロボットアーム 31 の動作の学習はこの例に限られず、操作者がロボットアーム 31 を手で保持しながらその操作を学習してもよいし、遠隔操作システムなどを導入して、遠隔操作にてロボットアーム 31 を操作して、その操作を学習してもよい。

【0062】

図 6 に、ロボットアーム 31 が学習する動作の例を示す。この例では、ロボットアーム 31 が障害物 40 に接触しながら、水平面（ $x-y$ 平面）内で障害物 40 を回避しながら移動するものとする。

【0063】

図 7 に、ロボットアーム 31 が障害物 40 を回避するときの各状態において作用する力を模式的に示す。以下、各状態における動作と作用する力について説明する。

【0064】

[状態 1]

ロボットアーム 31 は、初期位置から $+y$ 方向に直進して障害物 40 に接触する。この場合、状態 1 の最後で障害物 40 に接触するまでは、ロボットアーム 31 に作用する外力はほぼ 0 となる。

【0065】

[状態 2]

ロボットアーム 31 は、障害物 40 に接触すると、ロボットアーム 31 が障害物 40 に押しつけられた状態のまま、移動方向を $-x$ 方向に変えて、移動を継続する。この場合、ロボットアーム 31 には、障害物 40 に押しつけられたことによる $-y$ 方向の反力 $f_y 2$ と、 $+x$ 方向の摩擦力 $f_x 2$ が作用する。

10

20

30

40

50

【0066】

〔状態3〕

ロボットアーム31は、障害物40の角（-x側かつ-y側の角）に到達すると、移動方向を+y方向に変え、かつ、障害物40に対して+x方向に押しつけられた状態で、移動を継続する。この場合、ロボットアーム31には、障害物40に押しつけられたことによる-x方向の反力 f_x3 と、-y方向の摩擦力 f_y3 が作用する。

【0067】

〔状態4〕

ロボットアーム31は、障害物40の角（-x側かつ+y側の角）に到達すると、移動方向を+x方向に変え、かつ、障害物40に対して-y方向に押しつけられた状態で、移動を継続する。この場合、ロボットアーム31には、障害物40に押しつけられたことによる+y方向の反力 f_y4 と、-x方向の摩擦力 f_x4 が作用する。

【0068】

次いで、本実験における時系列データの形式について説明する。図8に、時系列データTSDの形式を示す。縦方向には時系列データTSDの各データ要素が配列されており、各行がサンプリングされた1つのデータ要素に対応する。各データ要素は、4つの領域DZ1～DZ4を含んでいる。

【0069】

第1の領域DZ1は、データ要素のサンプリングの番号*i*を示している。なお、ここではサンプリングの個数を*n*（*n*は1以上の整数）とし、よって、*i*は1以上*n*以下の整数である。第2の領域DZ2は、各データ要素をサンプリングした時刻（タイミング）*t*（*i*）を示している。

【0070】

第3の領域DZ3は、ロボットアーム31に作用する外力（ $f_x(i)$ 、 $f_y(i)$ ）、速度（ $v_x(i)$ 、 $v_y(i)$ ）、及び、位置（ $x(i)$ 、 $y(i)$ ）などであり、この領域DZ3に含まれる値は、制御対象の情報に関する値を示している。

【0071】

第4の領域DZ4は、ロボットアーム31に与える指令値として、操作によってロボットアーム31に加えた力（ $F_x(i)$ 、 $F_y(i)$ ）であり、この領域DZ4に含まれる量が指令値に対応する。なお、ここでは、力を指令値としているが、これは例示に過ぎず、速度及び位置などの他の値を指令値としてもよい。

【0072】

ここでは、力、速度及び位置がいずれのデータ要素に属するかを示すためにサンプリング番号*i*を用いているが、これは例示に過ぎず、サンプリング番号*i*に変えて時刻*t*（*i*）を用いてもよい。

【0073】

この例では、以下の式に示すように、上記の第3の領域DZ3に含まれる力及び速度を、学習用データDATに含まれる状態量*P*（*i*）として用いる。

〔数3〕

$$P(i) = \{f_x(i), f_y(i), v_x(i), v_y(i)\} \quad [3]$$

なお、状態量*P*はこれに限られるものではなく、必要に応じて、*i*よりも前の領域DZ3に含まれる値を含めてもよいし、領域DZ1、DZ2及びD4の一部又は全部の値を含めてもよい。

【0074】

また、以下の式に示すように、上記の第4の領域DZ4に含まれる指令値に対して所定の時間*h*だけ将来の指令値を指令値*Q*として用いる。

〔数4〕

$$Q(i+h) = \{F_x(i+h), F_y(i+h)\} \quad [4]$$

以上より、学習用データ DAT のデータ要素は以下の式で表される。

【数 5】

$$D_i = \{P(i), Q(i+h)\} \quad [5]$$

ここで、指令値 Q の時刻として h だけ将来の $i+h$ での値を用いている点について説明する。本実施の形態では、操作対象装置から状態量を取得し、探索部が取得した状態量に基づいて推定した指令値を操作対象装置に与え得ることで、操作対象装置を適切に操作するものである。そこで、状態量を取得してから指令値を与えるまでの間で、ある程度のタイムラグが存在することが考え得る。この場合、ある時刻に取得した状態量に対しては、このタイムラグ分だけ将来の指令値を対応させることが望ましい。そのため、本実施の形態では、このタイムラグを反映するため、 h だけ将来の指令値を状態量に対応させている。ただし、タイムラグが無視しうるほど小さいことも有るため、 h は 0 以上の任意の値、すなわち、0 以上の整数としてもよい。

【0075】

学習用データ DAT は、以下の式で表される。

【数 6】

$$DAT = \{D_1, D_2, D_3, \dots, D_n\} \quad [6]$$

【0076】

なお、本実施の形態では、式 [4] 及び [5] において、時間的に最後にサンプリングした指令値 $Q(n)$ よりも将来の指令値 $Q(n+1) \sim Q(n+h)$ については、指令値 $Q(n)$ を用いるものとした。但し、将来の指令値 $Q(n+1) \sim Q(n+h)$ はこれに限られるものではなく、動作が好適なものとなるように、適宜他の値を用いてもよい。また、将来の指令値 $Q(n+1) \sim Q(n+h)$ のように時系列データ TSD には本来存在しない指令値を有するデータ要素 D_i については、学習用データ DAT に含めないようにしてもよい。

【0077】

本実施の形態では、学習用データ DAT のデータ要素数をサンプリング数 n と同数としたが、これに限らない。例えば、 $\{D_1, D_3, D_5, \dots\}$ のように要素の一部のみを学習データ DAT としてもよいし、例えば D_i と D_{i+1} の間を補完するように新たなデータを作成して学習用データ DAT に加えてもよい。

【0078】

次いで、この例におけるモデル作成について説明する。モデル作成部 12 は、学習用データ DAT に対して教師なし学習であるクラスタリングを行い、クラスタリング結果をモデルとして生成する。ここでは、クラスタリング手法としては、 k -means 法を用いる。なお、クラスタリング手法はこれに限られるものではなく、DBSCAN (Density-based spatial clustering of applications with noise) などの他の各種のクラスタリング手法を用いてもよい。また、この例では距離指標としてユークリッド距離を用いるが、他の距離指標を適宜用いてもよい。

【0079】

図 9 に、ロボットアーム 31 の動作を学習させた例を示す。図 9 では、可視化しやすいように、学習データの状態量の中から f_x , f_y の 2 次元平面を抜き出してマッピングし

10

20

30

40

50

ている。破線で分けられている領域が各々のクラスタリング領域であり、この例では4つの領域に分けられている。図6及び7に示すように、領域CL1が障害物への衝突前の状態（状態1）、領域CL2が衝突後に左方向へ移動している状態（状態2）、CL3が障害物の左側を上方向へ移動している状態（状態3）、状態CL4が障害物を右方向へ移動している状態（状態4）に対応している。

【0080】

モデル作成部12は、学習用データDATに対してクラスタリング結果を付与したものをノードとして作成する。図10に、モデル作成部が作成するノードを示す。図10に示すように、学習用データDATの各データ要素にクラスタリング結果を示すCL1～CL4のいずれかを付加したものがノードとして作成される。すなわち、i番目のノードは、以下の式で表される。

【数7】

$$node_i = \{P(i), Q(i+h), cl_i\} \quad [7]$$

なお、 cl_i はクラスタ番号であり、CL1～CL4のいずれかの値をとる。

【0081】

モデル作成部12は、上式で表される学習用データDATの要素数（本実施の形態ではn個）のノードの集合であるモデルMDを出力する。モデルMDは、以下の式で表される。

【数8】

$$MD = \{node_1, node_2, \dots, node_n\} \quad [8]$$

【0082】

学習用データは時系列データであるので、当然ながら、学習により得られるノードも時系列に順序を有している。具体的には、ノードは時間に関する量、ここではサンプリング順序と紐付けることができるので、ノードを時系列に並べることが可能である。なお、ノードに時系列の順序を持たせる方法はこれに限られるものではなく、ノードを表現する多次元ベクトルに時間に関する量（すなわち、時刻）を要素として含めることで表現してもよい。

【0083】

この例では、4つのクラスタCL1～CL4は、時系列的にCL1、CL2、CL3、CL4の順であることが得られる。

【0084】

次に、情報処理装置100が行う処理について説明する。図11に、実施の形態1にかかる情報処理装置100における処理のフローチャートを示す。情報処理装置100の動作は、以下のステップSA1～SA5で構成される。

【0085】

ステップSA1

探索部2は、モデル保持部1からモデルMDを読み込む。

【0086】

ステップSA2

探索部2は、入力データである状態量pを取得する。

【0087】

ステップSA3

探索部2は、入力データである状態量pがモデルMDのクラスタのいずれに属するかを取得する。以下では、入力データである状態量pが属するクラスタを対象クラスタCTRG

10

20

30

40

50

と称する。ここでは、入力データである状態量 p と各クラスタの重心との距離を計算し、最短距離にある重心を有するクラスタを対象クラスタ C_{TRG} とする。なお、対象クラスタ C_{TRG} の決定方法はこれに限られず、他の決定方法を適宜用いてもよい。

【0088】

ステップ S A 4

探索部 2 は、対象クラスタ C_{TRG} と、対象クラスタ C_{TRG} に対して時間的に直後のクラスタ C_{NEXT} に含まれるノードから、入力データである状態量 p に最も近い状態量を有するノード N_{NEAR} を 1 つ探索し、探索結果を出力部 3 へ出力する。ここでは、ノードを探索するための距離指標として、ユークリッド距離を用いる。なお、以下では、入力データである状態量 p に最も近い状態量を有するノード N_{NEAR} を、単に、最も近いノード N_{NEAR} と称する。

【0089】

ここでは、最も近いノードを 1 つだけ探索したが、入力データである状態量 p から近い複数のノードを探索してもよい。この場合、状態量 p から近い順に所定個数のノードを探索してもよいし、状態量 p から所定の距離以内の複数のノードを探索してもよい。

【0090】

探索の対象となるクラスタは、対象クラスタ C_{TRG} 及び直後のクラスタ C_{NEXT} であるものとしたが、例えば、時間的に更に次のクラスタを対象にしてもよいし、直後のクラスタ C_{NEXT} のうちで時間的に早い半数のノードのみを対象としてもよい。

【0091】

また、ノードを探索するための距離指標としてユークリッド距離を用いたが、他の距離指標を適宜用いてもよい。

【0092】

ステップ S A 5

出力部 3 は、最も近いノード N_{NEAR} に基づいて決定した出力指令値 q_{OUT} を出力する。例えば、出力部 3 は、最も近いノード N_{NEAR} が保持する指令値 q_{NEAR} を、出力指令値 q_{OUT} として出力してもよい（第 1 の出力指令値決定方法）。

【0093】

なお、出力部 3 が出力指令値 q_{OUT} を決定する方法はこれに限られるものではなく、例えば以下のように決定してもよい。例えば、出力部 3 は、最も近いノード N_{NEAR} が属するクラスタの全ノードの指令値の平均値を、出力指令値 q_{OUT} として出力してもよい（第 2 の出力指令値決定方法）。

【0094】

また、出力部 3 は、最も近いノード N_{NEAR} が属するクラスタのノードの一部のノードの指令値の平均値を、出力指令値 q_{OUT} として出力してもよい（第 3 の出力指令値決定方法）。このとき、最も近いノード N_{NEAR} から所定の距離以内のノード、最も近いノード N_{NEAR} から距離が近い順位に選択した所定個数のノードなど、種々の選択方法によって選択したノードの指令値の平均値を出力指令値 q_{OUT} として出力してもよい。

【0095】

さらに、ステップ S A 4 において複数のノードを探索した場合には、複数のノードの指令値の平均値を出力指令値 q_{OUT} として出力してもよい（第 4 の出力指令値決定方法）。

【0096】

なお、第 2 ～第 4 の出力指令値決定方法においては平均値を用いたが、これは例示に過ぎず、中央値、最大値及び最小値など統計的な手法により算出可能な各種の値、換言すれば統計量に基づいて決定した値を指令値として算出してもよい。

【0097】

以上説明したように、本構成によれば、入力データ p に対応する出力値を求めるときに、入力データ p に対する距離を計算するノードを、全ノードではなく、限定されたクラスタ群に含まれるノードに限定することができる。これにより、一般的な k 近傍法などの手法と比べて、距離計算の対象となるノードの数を大幅に削減することができ、入力データ

10

20

30

40

50

pに対応する出力値を求める処理を高速化することが可能となる。

【0098】

情報処理装置100における基本的な動作は以上で説明した通りであるが、入力データxが複数個連続して入力される場合には、以下で説明する動作を行ってもよい。図12に、実施の形態1にかかる情報処理装置100における処理の変形例のフローチャートを示す。この変形例では、2個目以降の入力データpに対しては、1つ前に入力された入力データpに基づいて決定した対象クラスタC_{TRG}を用いて、最も近いノードN_{NEAR}を探索するものである。図12に示す動作では、図11のステップSA1～SA5の他に、ステップSA11～SA14が追加されている。SA1～SA5については図11と同様であるので、追加したステップSA11～SA14について説明する。

10

【0099】

ステップSA11

ステップSA11は、ステップSAの前段に挿入されるステップである。探索部2は、まず初めに、動作が初期状態、すなわち、入力データxの入力が初めてであることを示すため、初期フラグFGの値として0を設定する。なお、初期フラグはこの例に限られず、動作が初期状態であることを示せる限り、いかなる形式のデータでもよい。

【0100】

ステップSA12

ステップSA12は、ステップSA2とステップSA3との間に挿入されるステップである。探索部2は、動作が初期状態であるか、すなわち初期フラグFGの値が0であるかを判定する。初期フラグFGの値が0である（初期状態）場合には処理をステップSA3へ進め、初期フラグFGの値が0ではない（初期状態ではない）場合には処理をステップSA4へ進める。

20

【0101】

ステップSA13

ステップSA13は、ステップSA3とステップSA4との間に挿入されるステップである。探索部2は、動作が初期状態ではないことを示すため、初期フラグFGの値を1に切り替える。

【0102】

ステップSA14

探索部2は、ステップSA4で取得された最も近いノードN_{NEAR}が属するクラスタに基づいて、次の処理における対象クラスタC_{TRG}を決定する。ここでは、今回の処理で検出した最も近いノードN_{NEAR}のクラスタを対象クラスタC_{TRG}と決定する。なお、次の処理における対象クラスタC_{TRG}の決定方法はこれに限られず、例えば、過去の所定回数の処理で検出された最も近いノードN_{NEAR}が属するクラスタのうちで、最も出現頻度が多いクラスタを次の処理における対象クラスタC_{TRG}としてもよい。さらに、ステップSA4において複数のノードを探索した場合には、その複数のノードが属するクラスタのうちで頻度の多いクラスタを次の処理における対象クラスタC_{trg}としてもよい。例えば、ステップSA4で5つのノードが探索され、そのクラスタが{3, 3, 4, 4, 4}だった場合、対象クラスタを4としてもよい。

30

40

【0103】

以上、説明した変形例によれば、前回入力された入力データpに対応する最も近いノードN_{NEAR}が属するクラスタを、次回入力される入力データpに対して用いる対象クラスタC_{TRG}とすることができる。一般に時系列データは、隣接する入力データ間では急峻な変化は稀であり、比較的緩やかに変化することが期待されるため、隣接する入力データ間の距離も短くなることが期待される。

【0104】

そのため、次の処理で入力される入力データpについても、前回の処理の最も近いノードN_{NEAR}からの距離は短いと考えられる。よって、次の処理で用いるべき対象クラスタC_{TRG}も、結果として前回の処理の対象クラスタC_{TRG}と同じクラスタになる場合

50

が多いと考えられる。そこで、本変形例においては、前回の処理の最も近いノード N_{NEAR} が属するクラスタを次回の処理の対象クラスタ C_{TRG} として用いることで、2回目以降の処理の対象クラスタ C_{TRG} を簡便な処理で決定することができる。具体的には、1回目の対象クラスタ C_{TRG} の決定は、ステップ $SA3$ (図11) のように各クラスタの重心と入力データとの間の距離を算出しているため、各クラスタの重心を算出する必要がある。これに対し、本変形例においては、2回目以降の対象クラスタ C_{TRG} の決定を簡便な処理で行い、各クラスタの重心を計算せずともよいので、計算量を削減して処理時間の短縮を実現できる。なお、本実施の形態では、1回目の対象クラスタ C_{trg} の決定は、各クラスタの重心と入力データとの間の距離を算出して決定したが、これに限らず、時系列的に最初のクラスタ (本実施の形態では $CL1$) を1回目の対象クラスタ C_{trg} としてもよい。

10

【0105】

次いで、実施の形態1にかかる情報処理装置100での処理の具体例について説明する。図13に、実施の形態1にかかる情報処理装置100での処理と k 近傍法での処理とにおける距離計算の対象となるノードを示す。情報処理装置100では、上述の通り、2つのクラスタ $CL2$ 及び $CL3$ に属するノードのみが、入力データ p に対する距離の計算の対象となる。これに対し、 k 近傍法では、モデル MD に含まれる全てのノードが、入力データ p に対する距離の計算の対象となってしまう。なお、本実施の形態では、説明の簡略化のため、数十個のノードが属するクラスタを4つ含むモデルを例として説明したが、これは例示に過ぎない。より多数のクラスタを含み、かつ、1つのクラスタに属するノード

20

【0106】

よって、本構成によれば、入力データ p に対応する出力値を求めるときに、入力データ p に対する距離を計算するノードを制限して、処理速度を向上させることができる。

【0107】

実施の形態2

実施の形態2にかかる情報処理装置200について説明する。図14に、実施の形態2にかかる情報処理装置200の構成を模式的に示す。また、図15に、実施の形態2にかかる情報処理装置200が用いられる状況を模式的に示す。情報処理装置200は、情報処理装置100にモデル作成部4を追加し、かつ、探索部2を探索部6に置換した構成を有する。

30

【0108】

モデル作成部4は、時系列データである学習用データ DAT を読み込んで $SOINN$ 法による学習を行い、モデル MD を作成する。作成されたモデル MD は、モデル保持部1へ出力される。モデル作成部4における学習については、後述する。

【0109】

探索部5は、モデル MD を用いて、別途入力された推定の対象となるデータ (以下、入力データと称する) の状態量を参照し、モデル MD に含まれるノードの中から近似するノードを探索する。そして、探索結果を、出力部3へ出力する。

40

【0110】

データ保持部1及び出力部3については、実施の形態1と同様であるので、説明を省略する。

【0111】

次に、情報処理装置200が行う処理について説明する。図16に、実施の形態2にかかる情報処理装置200における動作のフローチャートを示す。

【0112】

ステップ $SB1$

50

モデル作成部 4 は、S O I N N 法により、状態量 P 及び指令値 Q を受け取って保持及び加工して得られる学習用データ D A T を学習してモデル M D を作成する。S O I N N 法は、多次元ベクトルで記述された入力データを入力することで、入力データを表現するノードが生成され、この生成されたノードによって構成されるネットワークとしてモデル M D が得られるものである。以下、S O I N N 法における学習処理について説明する。

【0113】

モデル作成部 4 は、S O I N N 法を用いることで、n 次元ベクトルで記述されるノードが配置される少なくとも 1 層以上の構造のニューラルネットワークからなるモデルを作成する。モデル作成部 4 が作成するモデルを構成するニューラルネットワークは、入力ベクトルをニューラルネットワークに入力し、入力される入力ベクトルに基づいて、ニューラルネットワークに配置されるノードを自動的に増加させる自己増殖型ニューラルネットワークであり、1 層構造を有するものである。

【0114】

これにより、自己増殖型ニューラルネットワークを用いてノードを自動的に増加させることができるため、順次入力ベクトルを入力することで、オンラインでの追加学習を行うことが可能である。

【0115】

以下、モデル作成部 4 の構成について説明する。図 17 に、実施の形態 2 にかかるモデル作成部 4 の構成を模式的に示す。モデル作成部 4 は、入力情報取得手段 41、勝者ノード探索手段 42、類似度閾値算出手段 43、類似度閾値判定手段 44、ノード挿入手段 45、重みベクトル更新手段 46、ノード密度算出手段 47、分布重なり領域検出手段 48、辺接続判定手段 49、辺接続手段 50、辺削除手段 51、ノイズノード削除手段 52 及び出力情報表示手段 53 を有する。

【0116】

なお、以下の S O I N N 法の説明において、i、j、k、x などの各種のアルファベットを用いるが、これは S O I N N 法の説明のために便宜上用いるものであり、実施の形態 1 において用いられるアルファベットと重複して使用されている場合でも、異なる値を示すものであるものとする。

【0117】

入力情報取得手段 41 は、モデル作成部 4 に入力として与えられる情報として、n 次元の入力ベクトルを取得する。そして、取得された入力ベクトルを一時記憶部（例えば R A M 1003）に格納し、一時記憶部に格納されたニューラルネットワークに対して順次入力する。

【0118】

勝者ノード探索手段 42 は、一時記憶部に格納される入力ベクトル及びノードについて、入力ベクトルに最も近い重みベクトルを持つノードを第 1 勝者ノードとし、2 番目に近い重みベクトルを持つノードを第 2 勝者ノードとして探索し、その結果を一時記憶部に格納する。すなわち、n 次元の入力ベクトル ξ に対して、一時記憶部に格納される以下の式を満足するノードを、それぞれ第 1 勝者ノード a_1 及び第 2 勝者ノード a_2 として探索し、その結果を一時記憶部に格納する。

【数 9】

$$a_1 = \operatorname{argmin}_{a \in A} \|\xi - W_a\| \quad [9]$$

【数 10】

$$a_2 = \operatorname{argmin}_{a \in A \setminus \{a_1\}} \|\xi - W_a\| \quad [10]$$

10

20

30

40

50

ここで、 a は一時記憶部に格納されたノード集合 A に含まれるノードであり、 W_a は一時記憶部に格納されたノード a の重みベクトルである。

【0119】

類似度閾値算出手段 43 は、一時記憶部に格納されたノード及びノードの類似度閾値について、注目するノードについて、注目するノードと辺によって直接的に接続されるノード（以下、隣接ノードという。）が存在する場合には、隣接ノードのうち注目するノードからの距離が最大であるノードまでの距離を類似度閾値として算出し、その結果を一時記憶部に格納し、隣接ノードが存在しない場合には、注目するノードからの距離が最小であるノードまでの距離を類似度閾値として算出し、その結果を一時記憶部に格納する。具体的には、例えば以下のようにして注目するノードの類似度閾値を算出し、その結果を一時記憶部に格納する。

10

〔手順 M__A1〕

類似度閾値算出手段 43 は、新しく挿入され一時記憶部に格納されたノード i の類似度閾値 T_i を $+\infty$ （十分大きな値）に設定し、その結果を一時記憶部に格納する。

〔手順 M__A2〕

一時記憶部に格納されたノードについて、ノード i が入力ベクトルから最も近いノードまたは 2 番目に近いノードとなった場合に、ノード i が隣接ノードを持つか否かを判定し、その結果を一時記憶部に格納する。

〔手順 M__A3〕

一時記憶部に格納された判定の結果、隣接ノードを持つ場合には、一時記憶部に格納された類似度閾値及びノードについて、類似度閾値 T_i を隣接ノードへの最大距離とし、その結果を一時記憶部に格納する。

20

即ち、ノード i について、一時記憶部に格納される以下の式に基づいて類似度閾値 T_i を算出し、その結果を一時記憶部に格納する。

〔数 11〕

$$T_i = \max_{c \in N_i} \|W_i - W_c\| \quad [11]$$

ここで、 c は一時記憶部に格納されたノード i の隣接ノード集合 N_i に含まれるノードであり、 W_c は一時記憶部に格納されたノード c の重みベクトルである。

30

〔手順 M__A4〕

判定の結果、隣接ノードを持たない場合には、ノード i からノード i を除いた他の各ノードへの距離を算出し、算出された距離のうち最小の距離を類似度閾値 T_i とする。

即ち、ノード i について、一時記憶部に格納される以下の式に基づいて類似度閾値 T_i を算出し、その結果を一時記憶部に格納する。

〔数 12〕

$$T_i = \min_{c \in A \setminus \{i\}} \|W_i - W_c\| \quad [12]$$

40

【0120】

類似度閾値判定手段 44 は、一時記憶部に格納された入力ベクトル、ノード及び、ノードの類似度閾値について、入力ベクトルと第 1 勝者ノードとの間の距離が第 1 勝者ノードの類似度閾値より大きいかな否か、及び、入力ベクトルと第 2 勝者ノード間の距離が第 2 勝者ノードの類似度閾値より大きいかな否かを判定し、その結果を一時記憶部に格納する。すなわち、一時記憶部に格納される以下の式に示すように、入力ベクトル ξ と第 1 勝者ノード a_1 の間の距離が類似度閾値 T_{a_1} よりも大きいかな否かを判定しその結果を一時記憶部に格納すると共に、入力ベクトル ξ と第 2 勝者ノード a_2 の間の距離が類似度閾値 T_{a_2} よりも大きいかな否かを判定しその結果を一時記憶部に格納する。

50

【数 1 3】

$$\|\xi - W_{a_1}\| > T_{a_1} \quad or \quad \|\xi - W_{a_2}\| > T_{a_2} \quad [13]$$

【0 1 2 1】

ノード挿入手段 4 5 は、一時記憶部に格納された類似度閾値判定手段 4 4 による判定結果に基づいて、一時記憶部に格納された入力ベクトルについて、入力ベクトルを新たなノードとして、入力ベクトルと同じ位置に挿入し、その結果を一時記憶部に格納する。

10

【0 1 2 2】

重みベクトル更新手段 4 6 は、一時記憶部に格納されたノードの重みベクトルについて、第 1 勝者ノードの重みベクトル及び第 1 勝者ノードの隣接ノードの重みベクトルをそれぞれ入力ベクトルに更に近づけるように更新し、その結果を一時記憶部に格納する。

第 1 勝者ノード a_1 の重みベクトルの更新量 ΔW_{a_1} 、及び第 1 勝者ノード a_1 の隣接ノード i の重みベクトルの更新量 ΔW_{a_i} は、例えば一時記憶部に格納される以下の式に基づいて算出し、その結果を一時記憶部に格納する。

【数 1 4】

$$\Delta W_{a_1} = \varepsilon_1(t)(\xi - W_{a_1}) \quad [14]$$

20

【数 1 5】

$$\Delta W_i = \varepsilon_2(t)(\xi - W_i) \quad [15]$$

ここで、 $\varepsilon_1(t)$ 及び $\varepsilon_2(t)$ はそれぞれ一時記憶部に格納される以下の式に基づいて算出し、その結果を一時記憶部に格納する。

30

【数 1 6】

$$\varepsilon_1(t) = 1/t \quad [16]$$

【数 1 7】

$$\varepsilon_2(t) = 1/100t \quad [17]$$

40

なお、本実施の形態においては、追加学習に対応するため、入力ベクトルの入力回数 t に代えて、一時記憶部に格納される第 1 勝者ノード a_1 が第 1 勝者ノードとなった累積回数 M_{a_1} を用いる。

【0 1 2 3】

ノード密度算出手段 4 7 は、一時記憶部に格納されたノード及びノード密度について、注目するノードについて、その隣接ノード間の平均距離に基づいて、注目するノードのノード密度を算出し、その結果を一時記憶部に格納する。さらに、ノード密度算出手段 4 7 は、単位ノード密度算出部を有し、単位ノード密度算出部は、追加学習に対応するため、一時記憶部に格納された第 1 勝者ノード及びノード密度について、第 1 勝者ノードとその隣接ノード間の平均距離に基づいて、第 1 勝者ノードのノード密度を単位入力数あたりの

50

割合として算出し、その結果を一時記憶部に格納する。さらにまた、ノード密度算出手段 47 は、一時記憶部に格納されたノード及びノード密度ポイントについて、第 1 勝者ノード及びその隣接ノード間の平均距離に基づいて、第 1 勝者ノードのノード密度のポイント値を算出するノード密度ポイント算出部と、入力ベクトルの入力数が所定の単位入力数となるまでノード密度ポイントを一時記憶部に格納して累積し、入力ベクトルの入力数が所定の単位入力数になった場合に、一時記憶部に格納して累積されたノード密度ポイントを単位入力数あたりの割合として算出し、単位入力数あたりのノードのノード密度を算出し、その結果を一時記憶部に格納する単位ノード密度ポイント算出部を有する。

【0124】

具体的には、ノード密度ポイント算出部は、例えば一時記憶部に格納される以下の式に基づいてノード i に与えられるノード密度のポイント値 pt_i を算出し、その結果を一時記憶部に格納する。尚、ノード i に与えられるポイント値 pt_i は、ノード i が第 1 勝者ノードとなった場合には一時記憶部に格納される以下の式に基づいて算出されるポイント値が与えられるが、ノード i が第 1 勝者ノードでない場合にはノード i にはポイントは与えられないものとする。

【数 18】

$$pt_i = \frac{1}{(1 + e_i)^2} \quad [18]$$

ここで、 e_i はノード i からその隣接ノードまでの平均距離を示し、一時記憶部に格納される以下の式に基づいて算出し、その結果を一時記憶部に格納する。

【数 19】

$$e_i = \frac{1}{m} \sum_{j=1}^m \|w_i - w_j\|^2 \quad [19]$$

尚、 m は一時記憶部に格納されたノード i の隣接ノードの個数を示し、 w_i は一時記憶部に格納されたノード i の重みベクトルを示す。

【0125】

ここで、隣接ノードへの平均距離が大きくなる場合には、ノードを含むその領域にはノードが少ないものと考えられ、逆に平均距離が小さくなる場合には、その領域にはノードが多いものと考えられる。

従って、ノードの多い領域で第 1 勝者ノードとなった場合には高いポイントが与えられ、ノードの少ない領域で第 1 勝者ノードとなった場合には低いポイントが与えられるようにノードの密度のポイント値の算出方法を上述のように構成する。

これにより、ノードを含むある程度の範囲の領域におけるノードの密集具合を推定することができるため、ノードの分布が高密度の領域に位置するノードであっても、ノードが第 1 勝者回数となった回数をノードの密度とする従来の場合に比べて、入力ベクトルの入力分布密度により近似した密度となるノード密度ポイントを算出することができる。

【0126】

単位ノード密度ポイント算出部は、例えば一時記憶部に格納される以下の式に基づいてノード i の単位入力数あたりのノード密度 $density_i$ を算出し、その結果を一時記憶部に格納する。

【数 20】

10

20

30

40

50

$$density_i = \overline{s_i} = \frac{1}{N} s_i \quad [20]$$

ここで、連続して与えられる入力ベクトルの入力回数を予め設定され一時記憶部に格納される一定の入力回数 λ ごとの区間に分け、各区間においてノード i に与えられたポイントについてその合計を累積ポイント s_i と定める。なお、入力ベクトルの総入力回数を予め設定され一時記憶部に格納される LT とする場合に、 LT/λ を区間の総数 n としその結果を一時記憶部に格納し、 n のうち、ノードに与えられたポイントの合計が 0 以上であった区間の数を N として算出し、その結果を一時記憶部に格納する（ N と n は必ずしも同じにならない点に注意する）。

累積ポイント s_i は、例えば一時記憶部に格納される以下の式に基づいて算出し、その結果を一時記憶部に格納する。

【数 2 1】

$$s_i = \sum_{j=1}^n \left(\sum_{k=1}^{\lambda} pt_i^{(j,k)} \right) \quad [21]$$

ここで、 $pt_i^{(j,k)}$ は j 番目の区間における k 番目の入力によってノード i に与えられたポイントを示し、上述のノード密度ポイント算出部により算出され、その結果を一時記憶部に格納する。

このように、単位ノード密度ポイント算出部は、一時記憶部に格納されたノード i の密度 $density_i$ を累積ポイント s_i の平均として算出し、その結果を一時記憶部に格納する。

【0 1 2 7】

なお、本実施の形態においては追加学習に対応するため、 n に代えて N を用いる。これは、追加学習において、以前の学習で生成されたノードにはポイントが与えられないことが多く、 n を用いて密度を算出すると、以前学習したノードの密度が次第に低くなってしまいうという問題を回避するためである。即ち、 n に代えて N を用いてノード密度を算出することで、追加学習を長時間行った場合であっても、追加されるデータが以前学習したノードの近くに入力されない限りは、そのノードの密度を変化させずに保持することができる。

これにより、追加学習を長時間実施する場合であっても、ノードのノード密度が相対的に小さくなってしまいうことを防ぐことができ、従来の手法に比べて、入力ベクトルの入力分布密度により近似したノード密度を変化させずに保持して算出することができる。

【0 1 2 8】

分布重なり領域検出手段 4 8 は、一時記憶部に格納されたノード、ノード間を接続する辺、及びノードの密度について、辺によって接続されるノードの集合であるクラスタを、ノード密度算出手段 4 7 によって算出されるノード密度に基づいてクラスタの部分集合であるサブクラスタに分割し、その結果を一時記憶部に格納し、サブクラスタの境界である分布の重なり領域を検出し、その結果を一時記憶部に格納する。

【0 1 2 9】

さらに、分布重なり領域検出手段 4 8 は、一時記憶部に格納されたノード、ノード間を接続する辺、及びノードの密度について、ノード密度算出手段 4 7 により算出されたノード密度に基づいて、ノード密度が局所的に最大であるノードを探索するノード探索部と、探索したノードに対して、既に他のノードに付与済みのラベルとは異なるラベルを付与する第 1 のラベル付与部と、第 1 のラベル付与部によりラベルが付与されなかったノードの

うち、そのノードと辺によって接続されるノードについて、第1のラベル付与部によりラベルが付与されたノードのラベルと同じラベルを付与する第2のラベル付与部と、それぞれ異なるラベルが付与されたノード間に辺によって直接的に接続がある場合に、その辺によって接続されるノードの集合であるクラスタをクラスタの部分集合であるサブクラスタに分割するクラスタ分割部と、注目するノード及びその隣接ノードがそれぞれ異なるサブクラスタに属する場合に、その注目するノード及びその隣接ノードを含む領域を、サブクラスタの境界である分布の重なり領域として検出する分布重なり領域検出部を有する。

【0130】

具体的には、一時記憶部に格納されたノード、ノード間を接続する辺、及びノードの密度について、例えば以下のようにしてサブクラスタの境界である分布の重なり領域を検出し、その結果を一時記憶部に格納する。

10

〔手順M__B1〕

ノード探索部は、一時記憶部に格納されたノード及びノードの密度について、ノード密度算出手段47により算出されたノード密度に基づいて、ノード密度が局所的に最大であるノードを探索し、その結果を一時記憶部に格納する。

〔手順M__B2〕

第1のラベル付与部は、一時記憶部に格納されたノード、及びノードのラベルについて、手順M__B1において探索したノードに対して、既に他のノードに付与済みのラベルとは異なるラベルを付与し、その結果を一時記憶部に格納する。

20

〔手順M__B3〕

第2のラベル付与部は、一時記憶部に格納されたノード、ノード間を接続する辺、及びノードのラベルについて、手順M__B2において第1のラベル付与部によりラベルが付与されなかったノードについて、第1のラベル付与部にラベルが付与されたノードと辺によって接続されるノードについて、第1のラベル付与部によりラベルが付与されたノードのラベルと同じラベルを付与し、その結果を一時記憶部に格納する。即ち、密度が局所的に最大の隣接ノードと同じラベルを付与する。

〔手順M__B4〕

クラスタ分割部は、一時記憶部に格納されたノード、ノード間を接続する辺、及びノードのラベルについて、一時記憶部に格納された辺によって接続されるノードの集合であるクラスタを、同じラベルが付与されたノードからなるクラスタの部分集合であるサブクラスタに分割し、その結果を一時記憶部に格納する。

30

〔手順M__B5〕

分布重なり領域検出部は、一時記憶部に格納されたノード、ノード間を接続する辺、及びノードのラベルについて、注目するノードとその隣接ノードが異なるサブクラスタにそれぞれ属する場合に、その注目するノード及びその隣接ノードを含む領域を、サブクラスタの境界である分布の重なり領域として検出し、その結果を一時記憶部に格納する。

【0131】

辺接続判定手段49は、一時記憶部に格納されたノード、ノード密度、及び分布重なり領域について、第1勝者ノード及び第2勝者ノードが分布重なり領域に位置するノードである場合に、第1勝者ノード及び第2勝者ノードのノード密度に基づいて第1勝者ノード及び第2勝者ノード間に辺を接続するか否かを判定し、その結果を一時記憶部に格納する。

40

【0132】

さらに辺接続判定手段49は、一時記憶部に格納されたノード、ノード密度、ノードのサブクラスタについて、ノードが属しているサブクラスタを判定する所属サブクラスタ判定部と、ノードが属するサブクラスタの頂点の密度及びノードの密度に基づいて、第1勝者ノード及び第2勝者ノード間に辺を接続するか否かを判定する辺接続判定部を有する。

【0133】

辺接続手段50は、一時記憶部に格納された辺接続判定手段49の判定結果に基づいて、一時記憶部に格納されたノード及びノード間の辺について、第1勝者ノード及び第2勝者ノード間に辺を接続し、その結果を一時記憶部に格納する。

50

【0134】

辺削除手段51は、一時記憶部に格納された辺接続判定手段49の判定結果に基づいて、一時記憶部に格納されたノード及びノード間の辺について、第1勝者ノード及び第2勝者ノード間の辺を削除し、その結果を一時記憶部に格納する。

【0135】

具体的には、一時記憶部に格納されたノード、ノード密度、ノードのサブクラスタ、及びノード間の辺について、例えば以下のようにして辺接続判定手段49は辺を接続するか否かを判定し、辺接続手段50及び辺削除手段51は辺の生成及び削除処理を実施し、その結果を一時記憶部に格納する。

[手順M__C1]

所属サブクラスタ判定部は、一時記憶部に格納されたノード、ノードのサブクラスタについて、第1勝者ノード及び第2勝者ノードが属するサブクラスタをそれぞれ判定し、その結果を一時記憶部に格納する。

[手順M__C2]

一時記憶部に格納された手順M__C1における判定の結果、第1勝者ノード及び第2勝者ノードがどのサブクラスタにも属していない場合、又は、第1勝者ノード及び第2勝者ノードが同じサブクラスタに属している場合には、辺接続手段50は、一時記憶部に格納されたノード及びノード間の辺について、第1勝者ノード及び第2勝者ノード間に辺を生成することによりノード間を接続し、その結果を一時記憶部に格納する。

[手順M__C3]

一時記憶部に格納された手順M__C1における判定の結果、第1勝者ノード及び第2勝者ノードが互いに異なるサブクラスタに属す場合には、辺接続判定部は、一時記憶部に格納されたノード、ノード密度、及びノード間の辺について、ノードが属するサブクラスタの頂点の密度及びノードの密度に基づいて、第1勝者ノード及び第2勝者ノード間に辺を接続するか否かを判定し、その結果を一時記憶部に格納する。

[手順M__C4]

一時記憶部に格納された手順M__C3における辺接続判定部による判定の結果、辺を接続する必要がないと判定した場合には、一時記憶部に格納されたノード及びノード間の辺について、第1勝者ノード及び第2勝者ノード間を辺によって接続せず、既にノード間が辺によって接続されていた場合には、辺削除手段51は、一時記憶部に格納されたノード及びノード間の辺について、一時記憶部に格納された第1勝者ノード及び第2勝者ノード間の辺を削除し、その結果を一時記憶部に格納する。

[手順M__C5]

一時記憶部に格納された手順M__C3における辺接続判定部による判定の結果、辺を接続する必要があると判定した場合には、辺接続手段50は、一時記憶部に格納されたノード及びノード間の辺について、第1勝者ノード及び第2勝者ノード間に辺を生成しノード間を接続する。

【0136】

ここで、辺接続判定部による判定処理について詳細に説明する。

まず、辺接続判定部は、一時記憶部に格納されたノード及びノード密度について、第1勝者ノードのノード密度 $density_{win}$ 及び第2勝者ノード密度 $density_{sec-win}$ のうち、最小のノード密度 m を例えば一時記憶部に格納される以下の式に基いて算出し、その結果を一時記憶部に格納する。

【数22】

$$m = \min(density_{win}, density_{sec-win}) \quad [22]$$

次に、一時記憶部に格納されたノード、ノードのノード密度、及びノードのサブクラスについて、第1勝者ノード及び第2勝者ノードがそれぞれ属するサブクラスタA及びサブ

クラスタ B について、サブクラスタ A の頂点の密度 A_{max} 及びサブクラスタ B の頂点の密度 B_{max} を算出し、その結果を一時記憶部に格納する。

尚、サブクラスタに含まれるノードのうち、ノード密度が最大であるノード密度をサブクラスタの頂点の密度とする。

そして、一時記憶部に格納されたノードが属するサブクラスタの頂点の密度 A_{max} 及び B_{max} 、及びノードの密度 m について、 m が $\alpha_A A_{max}$ より小さく、かつ、 m が $\alpha_B B_{max}$ より小さいか否かを判定し、その結果を一時記憶部に格納する。即ち、一時記憶部に格納される以下の不等式を満足するか否かを判定し、その結果を一時記憶部に格納する。

【数 2 3】

$$\alpha_A A_{max} \text{ and } \alpha_B B_{max} \quad [23]$$

10

判定の結果、 m が $\alpha_A A_{max}$ より小さく、かつ、 m が $\alpha_B B_{max}$ より小さい場合には、一時記憶部に格納されたノード及びノード間の辺について、第 1 勝者ノード及び第 2 勝者ノード間には辺は不要であると判定し、その結果を一時記憶部に格納する。

一方、判定の結果、 m が $\alpha_A A_{max}$ 以上、または、 m が $\alpha_B B_{max}$ 以上である場合には、一時記憶部に格納されたノード及びノード間の辺について、第 1 勝者ノード及び第 2 勝者ノード間には辺は必要であると判定し、その結果を一時記憶部に格納する。

【0137】

20

このように、第 1 勝者ノード及び第 2 勝者ノードの最小ノード密度 m を、第 1 勝者ノード及び第 2 勝者ノードをそれぞれ含むサブクラスタの平均的なノード密度と比較することで、第 1 勝者ノード及び第 2 勝者ノードを含む領域におけるノード密度の凹凸の大きさを判定することができる。即ち、サブクラスタ A 及びサブクラスタ B の間に存在する分布の谷間のノード密度 m が、閾値 $\alpha_A A_{max}$ 又は $\alpha_B B_{max}$ より大きな場合には、ノード密度の形状は小さな凹凸であると判定することができる。

【0138】

ここで、 α_A 及び α_B は一時記憶部に格納される以下の式に基づいて算出し、その結果を一時記憶部に格納する。尚、 α_B についても α_A と同様にして算出することができるためここでは説明を省略する。

30

i) $A_{max} / \text{mean}_A - 1 \leq 1$ の場合には、 $\alpha_A = 0.0$ とする。

ii) $1 < A_{max} / \text{mean}_A - 1 \leq 2$ の場合には、 $\alpha_A = 0.5$ とする。

iii) $2 < A_{max} / \text{mean}_A - 1$ の場合には、 $\alpha_A = 1.0$ とする。

A_{max} / mean_A の値が 1 以下となる i) の場合には、 A_{max} と mean_A の値は同程度であり、密度の凹凸はノイズの影響によるものと判断する。そして、 α の値を 0.0 とすることで、サブクラスタが統合されるようにする。

また、 A_{max} / mean_A の値が 2 を超える iii) の場合には、 A_{max} は mean_A に比べて十分大きく、明らかな密度の凹凸が存在するものと判断する。そして、 α の値を 1.0 とすることで、サブクラスタが分離されるようにする。

そして、 A_{max} / mean_A の値が上述した場合以外となる ii) の場合には、 α の値を 0.5 とすることで、密度の凹凸の大きさに応じてサブクラスタが統合又は分離されるようにする。

40

尚、 mean_A はサブクラスタ A に属すノード i のノード密度 density_i の平均値を示し、 N_A をサブクラスタ A に属するノードの数として、一時記憶部に格納される以下の式に基づいて算出し、その結果を一時記憶部に格納する。

【数 2 4】

50

$$mean_A = \frac{1}{N_A} \sum_{i \in A} density_i \quad [24]$$

【0139】

このように、サブクラスタへの分離を行う際に、サブクラスタに含まれるノード密度の凹凸の程度を判定し、ある基準を満たした2つのサブクラスタを1つに統合することで、分布の重なり領域の検出におけるサブクラスタの分けすぎによる不安定化を防止することができる。

10

【0140】

ノイズノード削除手段52は、一時記憶部に格納されたノード、ノード密度、ノード間の辺、隣接ノードの個数について、注目するノードについて、ノード密度算出手段47により算出されるノード密度及び注目するノードの隣接ノードの個数に基づいて、注目するノードを削除し、その結果を一時記憶部に格納する。

【0141】

さらにノイズノード削除手段52は、一時記憶部に格納されたノード、ノード密度、ノード間の辺、隣接ノードの個数について、注目するノードのノード密度を所定の閾値と比較するノード密度比較部と、注目するノードの隣接ノードの個数を算出する隣接ノード数算出部と、注目するノードをノイズノードとみなして削除するノイズノード削除部を有する。

20

具体的には、例えば以下のようにして一時記憶部に格納されたノード、ノード密度、ノード間の辺、隣接ノードの個数について、ノード密度及び注目するノードの隣接ノードの個数に基づいて、注目するノードを削除し、その結果を一時記憶部に格納する。

【0142】

ノイズノード削除手段52は、一時記憶部に格納されたノード、ノード間の辺、隣接ノードの個数について、注目するノード*i*について、隣接ノード数算出部によりその隣接ノードの個数を算出し、その結果を一時記憶部に格納する。そして、一時記憶部に格納された隣接ノードの個数に応じて、以下の処理を実施する。

30

i) 一時記憶部に格納された隣接ノード数が2の場合、ノード密度比較部はノード*i*のノード密度 $density_i$ を例えば一時記憶部に格納される以下の式に基づいて算出する閾値と比較し、その結果を一時記憶部に格納する。

【数25】

$$c_1 \sum_{j=1}^{N_A} density_i \quad [25]$$

40

一時記憶部に格納された比較結果について、ノード密度 $density_i$ が閾値より小さい場合には、ノイズノード削除部は、一時記憶部に格納されたノードについて、ノードを削除し、その結果を一時記憶部に格納する。

ii) 一時記憶部に格納された隣接ノード数が1の場合、ノード密度比較部はノード*i*のノード密度 $density_i$ を例えば一時記憶部に格納される以下の式に基づいて算出する閾値と比較し、その結果を一時記憶部に格納する。

【数26】

50

$$c_2 \sum_{j=1}^{N_A} density_i \quad [26]$$

一時記憶部に格納された比較の結果について、ノード密度 $density_i$ が閾値より小さい場合には、ノイズノード削除部は、一時記憶部に格納されたノードについて、ノードを削除し、その結果を一時記憶部に格納する。

i i i) 一時記憶部に格納された隣接ノード数について、隣接ノードを持たない場合、ノイズノード削除部は、一時記憶部に格納されたノードについて、ノードを削除し、その結果を一時記憶部に格納する。

ここで、予め設定され一時記憶部に格納される所定のパラメータ c_1 及び c_2 を調整することで、ノイズノード削除手段 5 2 によるノイズノードの削除の振る舞いを調整することができる。

【0143】

出力情報表示手段 5 3 は、一時記憶部に格納されたノードを含む情報、すなわちモデル MD を出力する。

【0144】

続いて、SOINN 法による学習処理のフローについて説明する。図 18 に、SOINN 法による学習処理のフローチャートを示す。

【0145】

ステップ M 1

入力情報取得手段 4 1 は、ランダムに 2 つの入力ベクトルを取得し、ノード集合 A をそれらに対応する 2 つのノードのみを含む集合として初期化し、その結果を一時記憶部に格納する。また、辺集合 $C \subset A \times A$ を空集合として初期化し、その結果を一時記憶部に格納する。

【0146】

ステップ M 2

入力情報取得手段 4 1 は、学習用データ DAT からランダムに選択された新しい入力ベクトル ξ を入力し、その結果を一時記憶部に格納する。ただし、一度選択された入力ベクトルが再び選択されることはない点については、言うまでもない。

【0147】

ステップ M 3

勝者ノード探索手段 4 2 は、一時記憶部に格納された入力ベクトル及びノードについて、入力ベクトル ξ に最も近い重みベクトルを持つ第 1 勝者ノード a_1 及び 2 番目に近い重みベクトルを持つ第 2 勝者ノード a_2 を探索し、その結果を一時記憶部に格納する。

【0148】

ステップ M 4

類似度閾値判定手段 4 4 は、一時記憶部に格納された入力ベクトル、ノード、ノードの類似度閾値について、入力ベクトル ξ と第 1 勝者ノード a_1 間の距離が第 1 勝者ノード a_1 の類似度閾値 T_1 より大きいかな否か、及び、入力ベクトル ξ と第 2 勝者ノード a_2 間の距離が第 2 勝者ノード a_2 の類似度閾値 T_2 より大きいかな否かを判定し、その結果を一時記憶部に格納する。

ここで、一時記憶部に格納された第 1 勝者ノード a_1 の類似度閾値 T_1 及び第 2 勝者ノード a_2 の類似度閾値 T_2 は、上述の手順 M_A 1 ~ 手順 M_A 4 において示したように類似度閾値算出手段 4 3 により算出され、その結果が一時記憶部に格納される。

【0149】

ステップ M 5

一時記憶部に格納されたステップ M 4 における判定の結果、入力ベクトル ξ と第 1 勝者

10

20

30

40

50

ノード a_1 間の距離が第 1 勝者ノード a_1 の類似度閾値 T_1 より大きい、又は、入力ベクトル ξ と第 2 勝者ノード a_2 間の距離が第 2 勝者ノード a_2 の類似度閾値 T_2 より大きい場合には、ノード挿入手段 45 は、一時記憶部に格納された入力ベクトル及びノードについて、入力ベクトル ξ を新たなノード i として、入力ベクトル ξ と同じ位置に挿入し、その結果を一時記憶部に格納する。

【0150】

ステップ M6

一方、一時記憶部に格納されたステップ M4 における判定の結果、入力ベクトル ξ と第 1 勝者ノード a_1 間の距離が第 1 勝者ノード a_1 の類似度閾値 T_1 以下であり、かつ、入力ベクトル ξ と第 2 勝者ノード a_2 間の距離が第 2 勝者ノード a_2 の類似度閾値 T_2 以下である場合には、辺接続判定手段 49 は、一時記憶部に格納されたノード、ノード密度、ノード間の辺について、第 1 勝者ノード a_1 及び第 2 勝者ノード a_2 のノード密度に基づいて、第 1 勝者ノード a_1 及び第 2 勝者ノード a_2 間に辺を接続するか否かを判定し、その結果を一時記憶部に格納する。

【0151】

ステップ M7

一時記憶部に格納されたステップ M6 における判定の結果、第 1 勝者ノード a_1 及び第 2 勝者ノード a_2 間に辺を生成して接続する場合には、辺接続手段 50 は、一時記憶部に格納されたノード及びノード間の辺について、第 1 勝者ノード及び第 2 勝者ノード間に辺を接続し、その結果を一時記憶部に格納する。

そして、情報処理装置は、一時記憶部に格納された辺及び辺の年齢について、新しく生成された辺、及び、既にノード間に辺が生成されていた場合にはその辺について、辺の年齢を 0 に設定しその結果を一時記憶部に格納し、第 1 勝者ノード a_1 と直接的に接続される辺の年齢をインクリメントし（1 増やす）、その結果を一時記憶部に格納する。

一方、一時記憶部に格納されたステップ M6 における判定の結果、第 1 勝者ノード a_1 及び第 2 勝者ノード a_2 間に辺を接続しない場合には、ステップ M8 へと処理を進めるが、既にノード間に辺が生成されていた場合には、辺削除手段 51 は、一時記憶部に格納されたノード及びノード間の辺について、第 1 勝者ノード a_1 及び第 2 勝者ノード a_2 間の辺を削除し、その結果を一時記憶部に格納する。尚、上述の手順 M_C1 ~ M_C5 において示したようにして、辺接続判定手段 49、辺接続手段 50、及び辺削除手段 51 は処理を実施する。

次いで、一時記憶部に格納されたノード及びノード密度のポイント値について、第 1 勝者ノード a_1 について、ノード密度算出手段 47 は、一時記憶部に格納された第 1 勝者ノード a_1 のノード密度のポイント値を算出しその結果を一時記憶部に格納し、算出され一時記憶部に格納されたノード密度のポイント値を以前までに算出され一時記憶部に格納されたポイント値に加算することで、ノード密度ポイントとして累積し、その結果を一時記憶部に格納する。

次いで、情報処理装置は、一時記憶部に格納された第 1 勝者ノード a_1 が第 1 勝者ノードとなった累積回数 M_{a_1} をインクリメントし（1 増やす）、その結果を一時記憶部に格納する。

【0152】

ステップ M8

重みベクトル更新手段 46 は、一時記憶部に格納されたノード及びノードの重みベクトルについて、第 1 勝者ノード a_1 の重みベクトル及び第 1 勝者ノード a_1 の隣接ノードの重みベクトルをそれぞれ入力ベクトル ξ に更に近づけるように更新し、その結果を一時記憶部に格納する。

【0153】

ステップ M9

情報処理装置は、一時記憶部に格納された辺について、予め設定され一時記憶部に格納された閾値 age_t を超えた年齢を持つ辺を削除し、その結果を一時記憶部に格納する。

尚、 $aget$ はノイズなどの影響により誤って生成される辺を削除するために使用する。 $aget$ に小さな値を設定することにより、辺が削除されやすくなりノイズによる影響を防ぐことができるものの、値を極端に小さくすると、頻繁に辺が削除されるようになり学習結果が不安定になる。一方、極端に大きな値を $aget$ に設定すると、ノイズの影響で生成された辺を適切に取り除くことができない。これらを考慮して、パラメータ $aget$ は実験により予め算出し一時記憶部に格納される。

【0154】

ステップM10

情報処理装置は、一時記憶部に格納された与えられた入力ベクトル ξ の総数について、与えられた入力ベクトル ξ の総数が予め設定され一時記憶部に格納された λ の倍数であるか否かを判定し、その結果を一時記憶部に格納する。一時記憶部に格納された判定の結果、入力ベクトルの総数が λ の倍数でない場合にはステップM2へと戻り、次の入力ベクトル ξ を処理する。

10

一方、入力ベクトル ξ の総数が λ の倍数となった場合には以下の処理を実行する。

尚、 λ はノイズと見なされるノードを削除する周期である。 λ に小さな値を設定することにより、頻繁にノイズ処理を実施することができるものの、値を極端に小さくすると、実際にはノイズではないノードを誤って削除してしまう。一方、極端に大きな値を λ に設定すると、ノイズの影響で生成されたノードを適切に取り除くことができない。これらを考慮して、パラメータ λ は実験により予め算出し一時記憶部に格納される。

【0155】

20

ステップM11

分布重なり領域検出手段48は、一時記憶部に格納されたサブクラスタ及び分布の重なり領域について、上述の手順M__B1～M__B5において示したようにしてサブクラスタの境界である分布の重なり領域を検出し、その結果を一時記憶部に格納する。

【0156】

ステップM12

ノード密度算出手段47は、一時記憶部に格納されて累積されたノード密度ポイントを単位入力数あたりの割合として算出しその結果を一時記憶部に格納し、単位入力数あたりのノードのノード密度を算出し、その結果を一時記憶部に格納する。

【0157】

30

ステップM13

ノイズノード削除手段52は、一時記憶部に格納されたノードについて、ノイズノードと見なしたノードを削除し、その結果を一時記憶部に格納する。尚、ステップM13においてノイズノード削除手段52が使用するパラメータ c_1 及び c_2 はノードをノイズと見なすか否かの判定に使用する。通常、隣接ノード数が2であるノードはノイズではないことが多いため、 c_1 は0に近い値を使用する。また、隣接ノード数が1であるノードはノイズであることが多いため、 c_2 は1に近い値を使用するものとし、これらのパラメータは予め設定され一時記憶部に格納される。

【0158】

ステップM14

40

情報処理装置は、一時記憶部に格納された与えられた入力ベクトル ξ の総数について、与えられた入力ベクトル ξ の総数が予め設定され一時記憶部に格納された LT であるか否かを判定し、その結果を一時記憶部に格納する。一時記憶部に格納された判定の結果、入力ベクトルの総数が LT の倍数でない場合にはステップM2へと戻り、次の入力ベクトル ξ を処理する。

一方、入力ベクトル ξ の総数が LT となった場合には、学習を停止する。

【0159】

以上のステップM1～M14を実行することで、SOINN法によって作成されたノードのネットワークとして構成されるモデルMDが得られる。図19に、実施の形態2にかかる情報処理装置200において実施の形態1で使用した学習用データDATをSOIN

50

N法で学習することで得られたノードの分布の例を示す。図19では、可視化しやすくするため、学習データの状態量の中から f_x , f_y の2次元平面を抜き出してマッピングしている。

【0160】

この例では、SOINN法で学習を行うことで、学習用データに含まれるデータ要素の学習結果は、SOINN法によって生成された代表的なノードによって表現されている。つまり、学習用データDATに含まれるデータ要素の数に比べて、モデルMDに含まれるノードの数は少なくなる。

【0161】

SOINN法は、例えば特許文献1及び2におけるSOINN法が適用できる。また、学習用データに含まれるデータ要素の分布構造を、データ要素よりも少数のノードによって表現された分布構造として学習する各種のSOINN法を適用してもよい。

【0162】

引き続き、図16に戻り、情報処理装置200でのステップSB2以降の処理について説明する。

【0163】

ステップSB2

探索部5は、図11のステップSA1と同様に、モデル保持部1からモデルMDを読み込む。

【0164】

ステップSB3

探索部5は、図11のステップSA2と同様に、入力データである状態量pを取得する。

【0165】

ステップSB4

探索部5は、図11のステップSA4と同様に、モデルMDから、入力データpに最も近いノードN_{NEAR}を探索する。

【0166】

ステップSB5

出力部3は、図11のステップSA5と同様に、実施の形態1と同様である。つまり、出力部3は、最も近いノードN_{NEAR}に基づいて決定した出力指令値q_{OUT}を出力する。なお、実施の形態1と同様に、第1～第4の出力指令値決定方法を含む各種の方法を用いて出力指令値q_{OUT}を決定してもよい。

【0167】

以上説明したように、本構成によれば、入力データpに対応する出力値を求めるときに、入力データpに対する距離を計算するノードを、学習用データのデータ要素の数よりも少ない数のノードに制限することができる。これにより、学習用データ又は学習用データのデータ要素と同じ数のノードを用いる一般的なk近傍法などの手法と比べて、距離計算の対象となるノードの数を大幅に削減することができ、入力データpに対応する指令値を求める処理を高速化することが可能となる。

【0168】

よって、本構成によれば、実施の形態1と同様に、入力データpに対応する出力値を求めるときに、入力データpに対する距離を計算するノードを制限して、処理速度を向上させることができる。

【0169】

その他の実施の形態

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、上述の実施の形態において、2つの値の大小判定について説明したが、これは例示に過ぎず、2つの値の大小判定において2つの値が等しい場合については、必要に応じて取り扱ってもよい。すなわち、第1の値が第2の値以上であるか又は第2の値よりも小さいかの判定と、第1の値が第2の値よりも大きいのか又は第

10

20

30

40

50

2の値以下であるかの判定とについては、必要に応じていずれを採用してもよい。第1の値が第2の値以下であるか又は第2の値よりも大きいかの判定と、第1の値が第2の値よりも小さいか又は第2の値以上であるかの判定については、いずれを採用してもよい。換言すれば、2つの値の大小判定を行って2つの判定結果を得る場合、2つの値が等しい場合については、必要に応じて2つの判定結果のいずれに含めてもよい。

【0170】

上述の実施の形態では、本発明を主にハードウェアの構成として説明したが、これに限定されるものではなく、任意の処理を、CPU (Central Processing Unit) にコンピュータプログラムを実行させることにより実現することも可能である。プログラムは、コンピュータに読み込まれたときに、コンピュータに上述の実施の形態に記載した1つ以上の機能を実行させる指令（又はソフトウェアコード）を含むものである。プログラムは、非一時的なコンピュータ可読媒体又は有形の記憶媒体に格納されてもよい。これに限られるものではないが、非一時的なコンピュータ可読媒体又は有形の記憶媒体の例は、RAM (Random-access memory: RAM)、ROM (Read-only memory)、フラッシュメモリ (Flash memory)、SSD (Solid-state drive)、又は、他の種類の記憶技術、例えば、CD (Compact disc) - ROM、DVD (Digital versatile disc)、ブルーレイ・ディスク (Blu-ray disc登録商標) 又は他の種類の光学ディスク記憶装置、磁気カセット (Magnetic cassette)、磁気テープ (Magnetic tape)、磁気ディスク記憶装置 (Magnetic disk storage) 又は他の磁気記憶装置を含む。プログラムは、一時的なコンピュータ可読媒体又は通信媒体によって送信されてもよい。これに限られるものではないが、一時的なコンピュータ可読媒体又は有形の記憶媒体の例は、電氣的、光学的、音響的又は他の形式の伝搬信号を含んでもよい。

【0171】

実施の形態1においては、学習データDATを学習して得られたノードをクラスタリングするものとして説明したが、実施の形態2で説明したSOINN法によって学習データDATを学習することで得られたノードに対してクラスタリング行ってもよいことは、言うまでもない。

【0172】

上記の実施形態の一部又は全部は、以下の付記のようにも記載されうるが、以下には限られない。

【0173】

(付記1) 学習対象から予め取得した第1の状態量と前記学習対象の動作を制御するために与えた前記第1の状態量に対応する第1の指令値と、を含む、多次元ベクトルで表されたデータ要素を複数含む時系列データに基づいた学習用データを学習することで、前記学習用データの分布構造を学習結果に基づいた第2の状態量及び第2の指令値を含む多次元ベクトルとして表されるノードの集合として作成されたモデルを保持するモデル保持部と、制御対象から取得した第3の状態量が入力データとして入力され、前記モデルに含まれる前記学習用データのデータ要素数よりも少数の前記ノードから前記入力データに一致又は近似するノードを探索する探索部と、前記探索したノードの前記第2の指令値に基づく値を、前記制御対象を操作するために与える出力指令値として前記制御対象へ出力する出力部と、を備える、情報処理装置。

【0174】

(付記2) 前記モデルは、前記学習用データのそれぞれをノードとして学習することで作成され、前記探索部は、前記ノードから、前記少数の前記ノードとして、前記入力データに対して時間的に比較的近いノードを選択し、前記選択したノードから前記入力データに対して最短距離のノードを探索する、付記1に記載の情報処理装置。

【0175】

(付記3) 前記モデルは、前記学習用データを時系列クラスタリングして、前記ノードを複数のクラスタに分類することで作成され、前記探索部は、前記入力データが属するクラスタと、前記入力データが属するクラスタに時間的に直後のクラスタと、を探索し、探

索した2つのクラスタに属するノードから前記入力データに対して最短距離のノードを探索する、付記2に記載の情報処理装置。

【0176】

(付記4) 前記探索部は、前記入力データに対して重心が最短距離にあるクラスタを前記入力データが属するクラスタとして探索する、付記3に記載の情報処理装置。

【0177】

(付記5) 前記探索部は、最初に入力された入力データについては、前記最初に入力された入力データに対して重心が最短距離にあるクラスタを前記最初に入力された入力データが属するクラスタとして探索し、前記最初に入力された入力データよりも後に入力される入力データについては、1つ前に入力された入力データについて探索された最短距離のノードが属するクラスタを、前記最初に入力された入力データよりも後に入力される入力データが属するクラスタとして探索する、付記3に記載の情報処理装置。

10

【0178】

(付記6) 前記探索部は、最初に入力された入力データについては、前記最初に入力された入力データに対して重心が最短距離にあるクラスタを前記最初に入力された入力データが属するクラスタとして探索し、前記最初に入力された入力データよりも後に入力される入力データについては、過去に入力された複数の入力データについて探索された最短距離のノードが属する複数のクラスタのうちで、出現頻度が最も高いクラスタを、前記最初に入力された入力データよりも後に入力される入力データが属するクラスタとして探索する、付記4に記載の情報処理装置。

20

【0179】

(付記7) 前記探索部は、前記探索したノードが属するクラスタの一部又は全部のノードの指令値から算出される統計量に基づいて、前記出力指令値を決定する、付記4乃至6のいずれか一つに記載の情報処理装置。

【0180】

(付記8) 前記一部のノードは、前記探索したノードが属するクラスタにおいて、前記探索したノードから所定の距離以内のノード、又は、前記探索したノードから近い順に選択された所定の個数のノードである、付記7に記載の情報処理装置。

【0181】

(付記9) 前記探索部は、前記探索したノードとの指令値と、前記入力データが属するクラスタ及び前記入力データが属するクラスタに時間的に直後のクラスタのうちで前記探索したノードに近似する1つ以上のノードの指令値と、から算出される統計量に基づいて、前記出力指令値を決定する、付記4乃至6のいずれか一つに記載の情報処理装置。

30

【0182】

(付記10) 前記探索部は、最初に入力された入力データについては、前記最初に入力された入力データに対して重心が最短距離にあるクラスタを前記最初に入力された入力データが属するクラスタとして探索し、前記最初に入力された入力データよりも後に入力される入力データについては、1つ前に入力された入力データについて探索したノードが属するクラスタと、前記探索したノードに近似する1つ以上のノードが属するクラスタのうちで出現頻度が最も高いクラスタを、前記最初に入力された入力データよりも後に入力される入力データが属するクラスタとして探索する、付記9に記載の情報処理装置。

40

【0183】

(付記11) 前記学習用データのデータ要素よりも少数のノードによって前記学習用データのデータ要素の分布構造を近似的に学習することで前記モデルを作成し、作成した前記モデルを前記モデル保持部へ出力するモデル作成部をさらに備え、前記探索部は、前記モデルに含まれる前記ノードから前記入力データに近似するノードを探索する、付記1に記載の情報処理装置。

【0184】

(付記12) 前記探索部は、前記探索したノードの指令値を、前記出力指令値として出力する、付記2乃至4及び11のいずれか一つに記載の情報処理装置。

50

【0185】

(付記13) 前記探索部は、前記探索したノードとの指令値と、前記探索したノードに近似する1つ以上のノードの指令値と、から算出される統計量に基づいて、前記出力指令値を決定する、付記2乃至4及び11のいずれか一つに記載の情報処理装置。

【0186】

(付記14) 前記探索したノードに近似する1以上のノードは、前記探索したノードから所定の距離以内のノード、又は、前記探索したノードから近い順に選択された所定の個数のノードである、付記9、10及び13のいずれか一つに記載の情報処理装置。

【0187】

(付記15) 前記統計量は、平均値、中央値、最大値、最小値及び最頻値のいずれかである、付記7乃至10、13及び14のいずれか一つに記載の情報処理装置。

10

【0188】

(付記16) 前記第2の状態量は、前記学習対象から予め取得した前記第1の状態量であり、かつ、前記第2の指令値は、前記予め取得した前記第1の状態量に応じて前記学習対象に与えられた前記第1の指令値である、付記1乃至15のいずれか一つに記載の情報処理装置。

【0189】

(付記17) 前記第1の指令値は、前記学習対象から予め取得した前記第1の状態量に基づいて前記学習対象の操作者が前記学習対象に与えた指令値の実績値である、付記16に記載の情報処理装置。

20

【0190】

(付記18) 前記学習用データのデータ要素に含まれる前記第1の指令値は、前記学習対象から予め取得した前記第1の状態量に対して、所定の時間だけ後に前記制御対象に与えられた指令値である、付記16に記載の情報処理装置。

【0191】

(付記19) 前記情報処理装置は、前記学習対象又は前記制御対象に設けられる、付記1乃至17のいずれか一つに記載の情報処理装置。

【0192】

(付記20) 前記学習対象及び前記制御対象は同一である、付記1乃至17のいずれか一つに記載の情報処理装置。

30

【0193】

(付記21) 前記情報処理装置は、前記制御対象に設けられる、付記20に記載の情報処理装置。

【0194】

(付記22) 学習対象から予め取得した第1の状態量と前記学習対象の動作を制御するために与えた前記第1の状態量に対応する第1の指令値と、を含む、多次元ベクトルで表されたデータ要素を複数含む時系列データに基づいた学習用データを学習することで、前記学習用データの分布構造を学習結果に基づいた第2の状態量及び第2の指令値を含む多次元ベクトルとして表されるノードの集合として作成されたモデルを保持し、制御対象から取得した第3の状態量が入力データとして入力され、前記モデルに含まれる前記学習用データのデータ要素数よりも少数の前記ノードから前記入力データに一致又は近似するノードを探索し、前記探索したノードの前記第2の指令値に基づく値を、前記制御対象を操作するために与える出力指令値として前記制御対象へ出力する、情報処理方法。

40

【0195】

(付記23) 学習対象から予め取得した第1の状態量と前記学習対象の動作を制御するために与えた前記第1の状態量に対応する第1の指令値と、を含む、多次元ベクトルで表されたデータ要素を複数含む時系列データに基づいた学習用データを学習することで、前記学習用データの分布構造を学習結果に基づいた第2の状態量及び第2の指令値を含む多次元ベクトルとして表されるノードの集合として作成されたモデルを保持する処理と、制御対象から取得した第3の状態量が入力データとして入力され、前記モデルに含まれる前

50

記学習用データのデータ要素数よりも少数の前記ノードから前記入力データに一致又は近似するノードを探索する処理と、前記探索したノードの前記第2の指令値に基づく値を、前記制御対象を操作するために与える出力指令値として前記制御対象へ出力する処理と、をコンピュータに実行させる、プログラム。

【0196】

(付記24) 学習対象から予め取得した第1の状態量と前記学習対象の動作を制御するために与えた前記第1の状態量に対応する第1の指令値と、を含む、多次元ベクトルで表されたデータ要素を複数含む時系列データに基づいた学習用データを学習することで、前記学習用データの分布構造を学習結果に基づいた第2の状態量及び第2の指令値を含む多次元ベクトルとして表されるノードの集合としてモデルを作成するモデル作成部と、作成された前記モデルを保持するモデル保持部と、制御対象から取得した第3の状態量が入力データとして入力され、前記モデルに含まれる前記学習用データのデータ要素数よりも少数の前記ノードから前記入力データに一致又は近似するノードを探索する探索部と、前記探索したノードの前記第2の指令値に基づく値を、前記制御対象を操作するために与える出力指令値として前記制御対象へ出力する出力部と、を備える、情報処理システム。

【0197】

以上、実施の形態を参照して本願発明を説明したが、本願発明は上記によって限定されるものではない。本願発明の構成や詳細には、発明のスコープ内で当業者が理解し得る様々な変更をすることができる。

【0198】

この出願は、2021年8月18日に提出された日本出願特願2021-133415を基礎とする優先権を主張し、その開示の全てをここに取り込む。

【符号の説明】

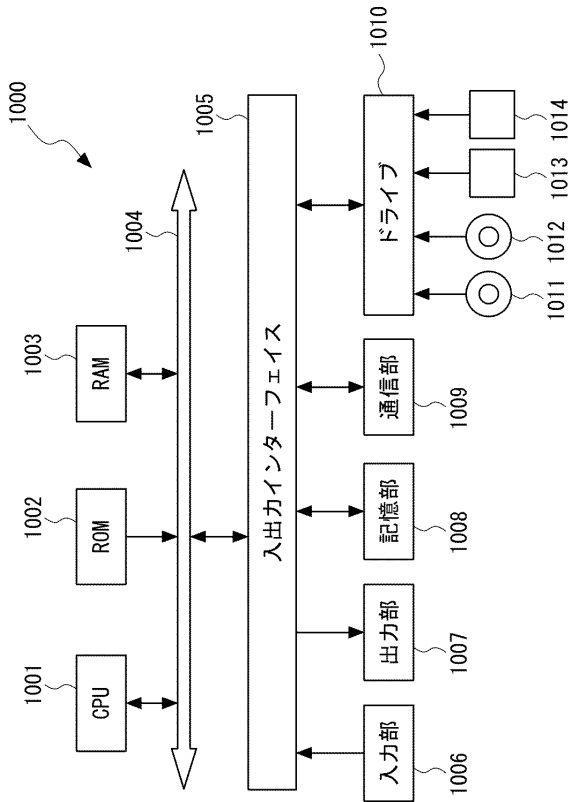
【0199】

- 1 モデル保持部
- 2 探索部
- 3 出力部
- 4 モデル作成部
- 5 探索部
- 10 操作者
- 11 指令装置
- 20 操作対象装置
- 30 ロボット
- 31 ロボットアーム
- 32 カセンサ
- 33 円柱ボール
- 41 入力情報取得手段
- 42 勝者ノード探索手段
- 43 類似度閾値算出手段
- 44 類似度閾値判定手段
- 45 ノード挿入手段
- 46 重みベクトル更新手段
- 47 ノード密度算出手段
- 48 分布重なり領域検出手段
- 49 辺接続判定手段
- 50 辺接続手段
- 51 辺削除手段
- 52 ノイズノード削除手段
- 53 出力情報表示手段
- 100 情報処理装置

- 110 処理部
- 120 表示部
- 130 入力部
- 200 情報処理装置
- 1000 コンピュータ
- 1001 CPU
- 1002 ROM
- 1003 RAM
- 1004 バス
- 1005 入出力インターフェイス
- 1006 入力部
- 1007 出力部
- 1008 記憶部
- 1009 通信部
- 1010 ドライブ
- 1011 磁気ディスク
- 1012 光ディスク
- 1013 フレキシブルディスク
- 1014 半導体メモリ

【図面】

【図1】



【図2】

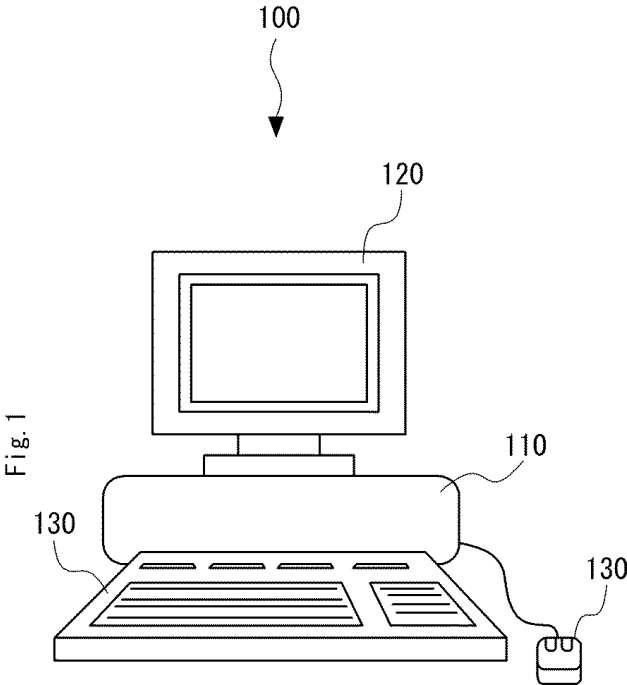


Fig. 2

【図 3】

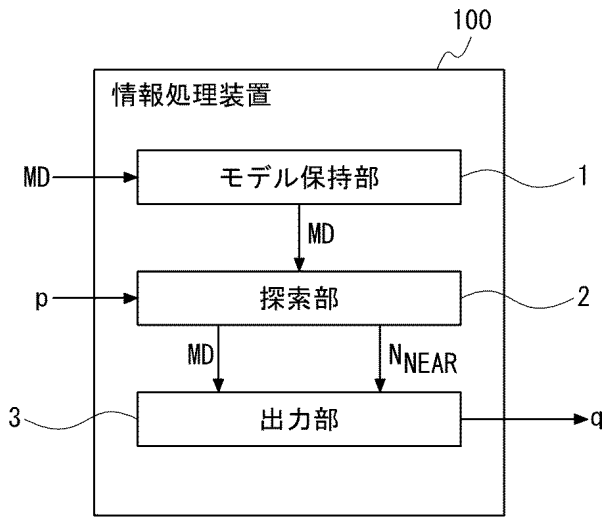


Fig. 3

【図 4】

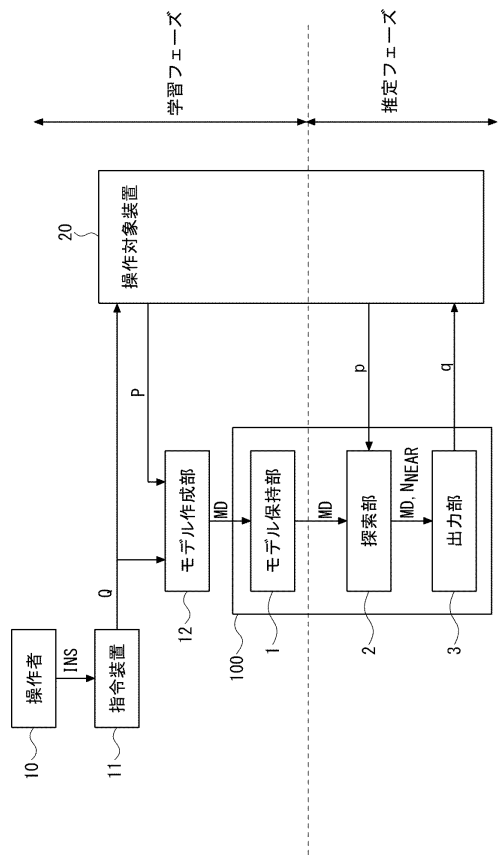


Fig. 4

【図 5】

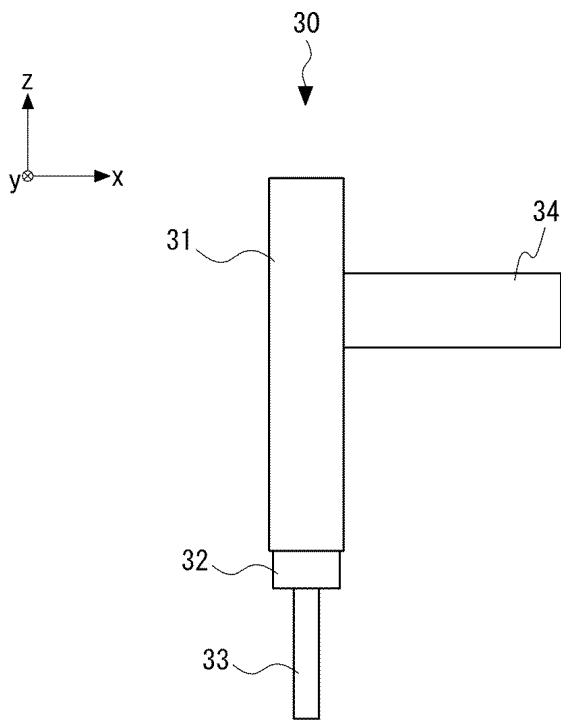


Fig. 5

【図 6】

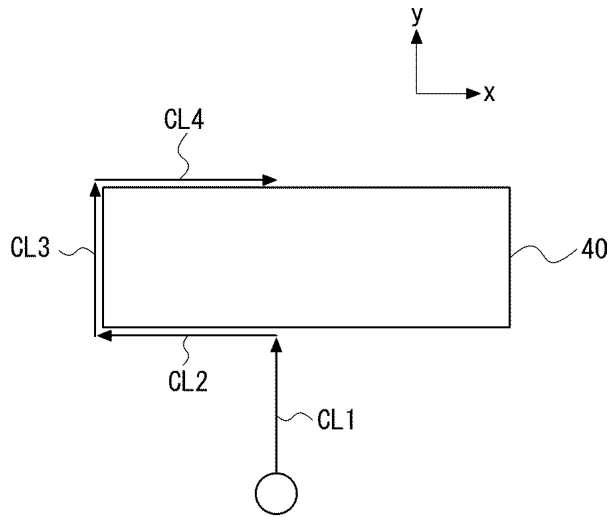


Fig. 6

10

20

30

40

50

【図 7】

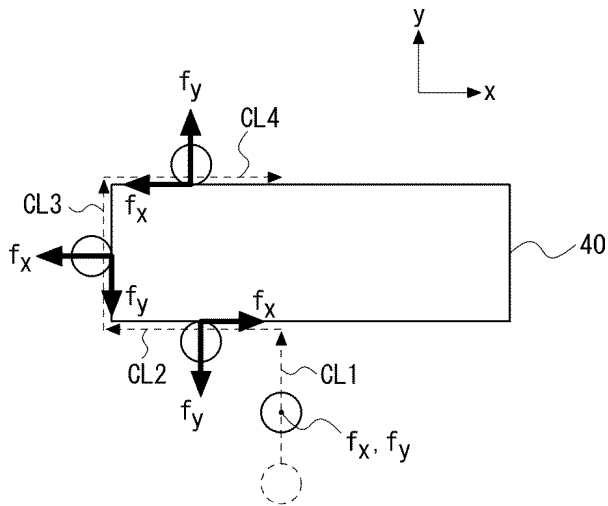


Fig. 7

【図 8】

		TSD	
		↓	
DZ1	DZ2	DZ3	DZ4
1	t (1)	$f_x(1), f_y(1), v_x(1), v_y(1), x(1), y(1)$	$F_x(1), F_y(1)$
2	t (2)	$f_x(1), f_y(2), v_x(2), v_y(2), x(2), y(2)$	$F_x(2), F_y(2)$
⋮			
n	t (n)	$f_x(n), f_y(n), v_x(n), v_y(n), x(n), y(n)$	$F_x(n), F_y(n)$

Fig. 8

【図 9】

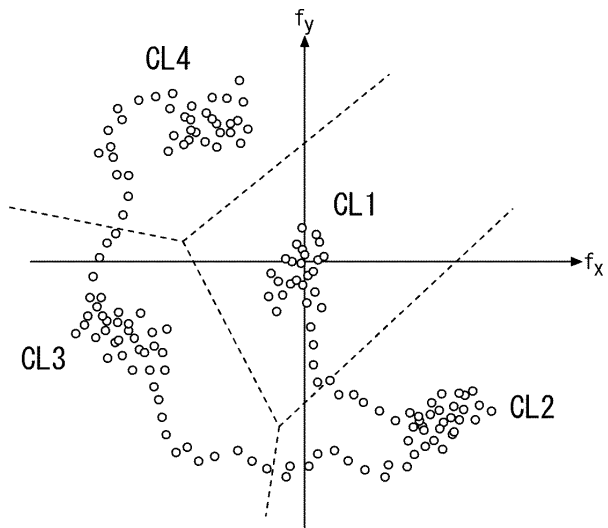


Fig. 9

【図 10】

$p(1), q(1+h), CL1$
 $p(2), q(2+h), CL1$
 $\vdots$
 $p(n), q(n+h), CL4$

Fig. 10

【図 1 1】

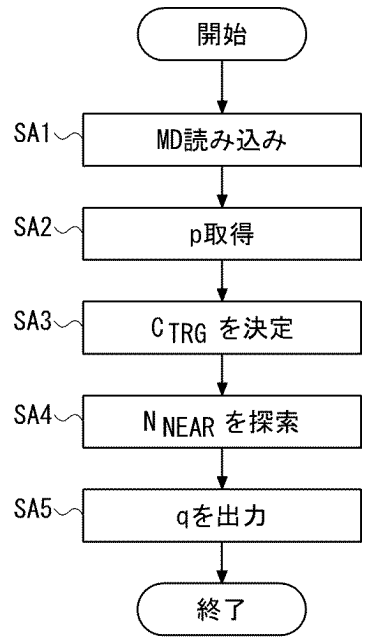


Fig. 11

【図 1 2】

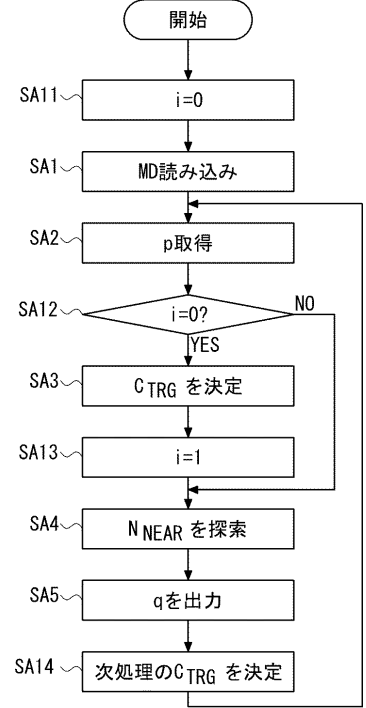


Fig. 12

【図 1 3】

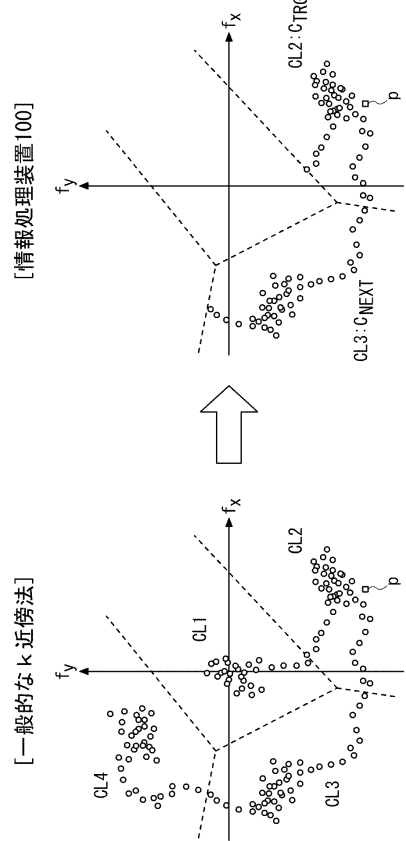


Fig. 13

【図 1 4】

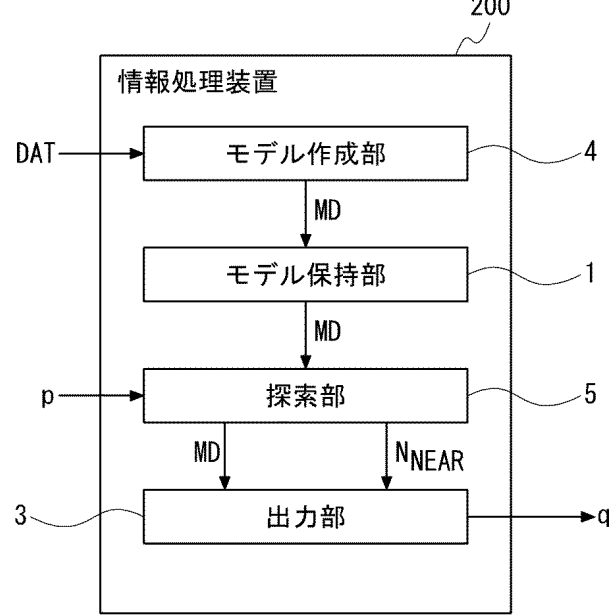


Fig. 14

10

20

30

40

50

【図 15】

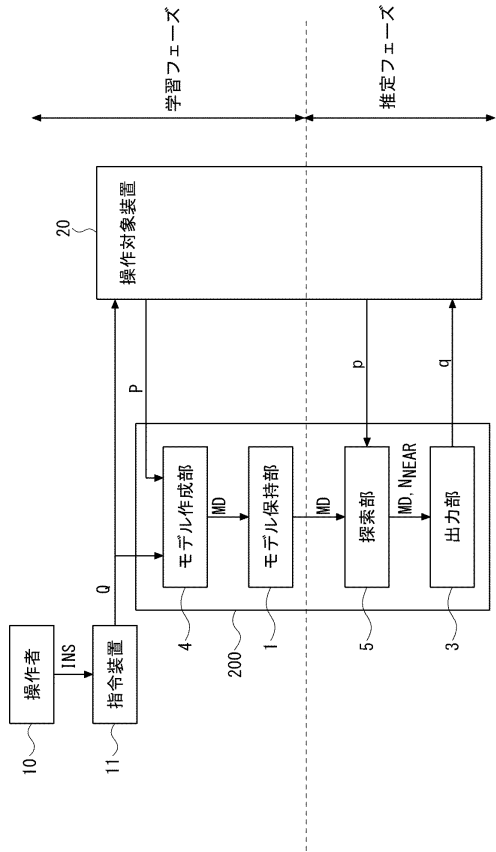


Fig. 15

【図 16】

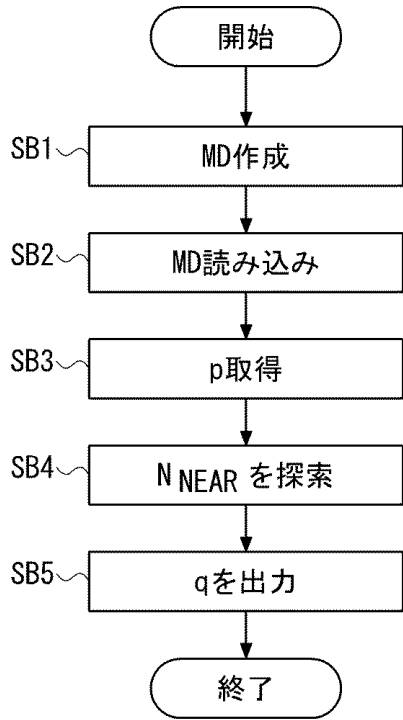


Fig. 16

【図 17】

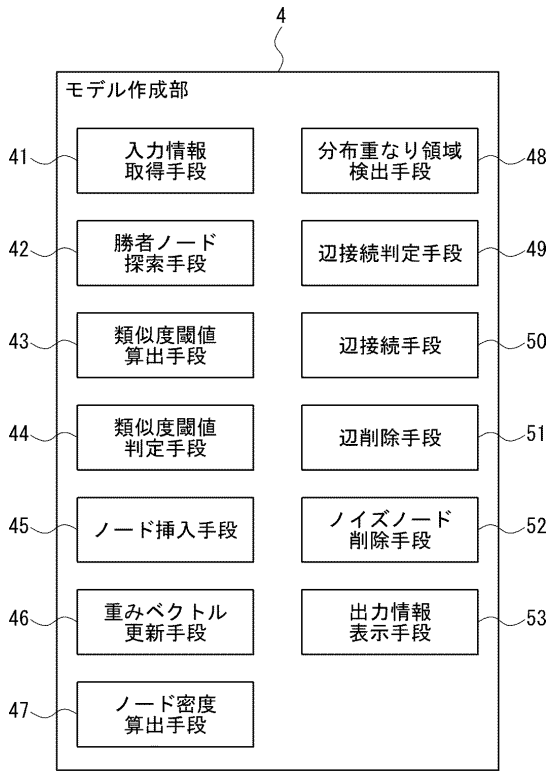


Fig. 17

【図 18】

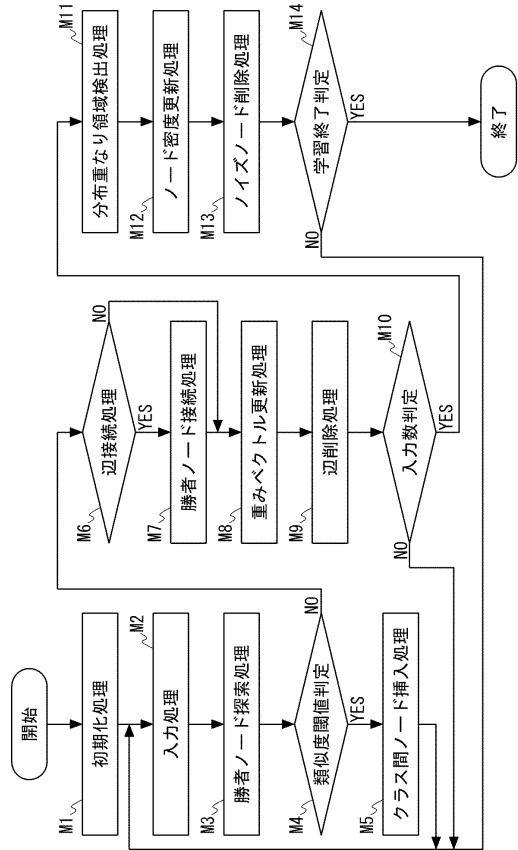


Fig. 18

【図 19】

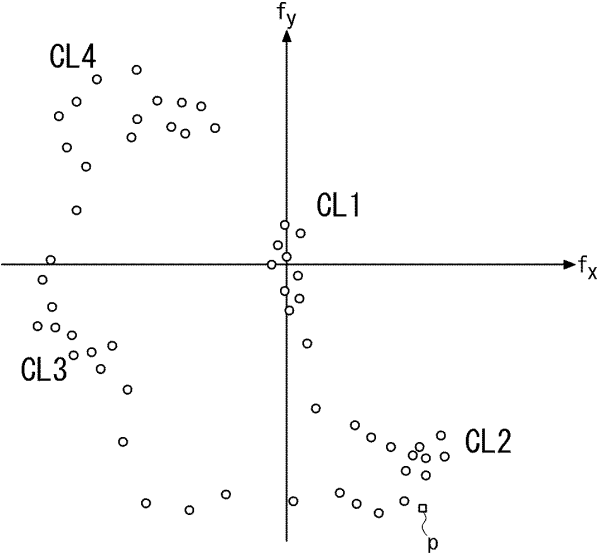


Fig. 19

フロントページの続き

ステムの研究開発」、産業技術力強化法第17条の適用を受ける特許出願

- (56)参考文献 国際公開第2016/114240 (WO, A1)
特開2000-276484 (JP, A)
特開2008-299640 (JP, A)
国際公開第2020/241419 (WO, A1)

- (58)調査した分野 (Int.Cl., DB名)
G06N 3/00-99/00
B25J 9/22
G06F 16/906
G06F 18/23