(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第5874292号
(P5874292)

(45)発行日　平成28年3月2日(2016.3.2)　(24)登録日　平成28年1月29日(2016.1.29)

(51)Int.Cl.		F I		
G06N 99/00	(2010.01)	G06N	99/00	150
A63F 13/56	(2014.01)	A63F	13/56	
A63F 13/67	(2014.01)	A63F	13/67	

請求項の数 15　(全 60 頁)

(21)出願番号　特願2011-224638 (P2011-224638)
(22)出願日　平成23年10月12日 (2011.10.12)
(65)公開番号　特開2013-84175 (P2013-84175A)
(43)公開日　平成25年5月9日 (2013.5.9)
審査請求日　平成26年9月29日 (2014.9.29)

(73)特許権者　000002185
ソニー株式会社
東京都港区港南１丁目７番１号
(74)代理人　100095957
弁理士　亀谷　美明
(74)代理人　100096389
弁理士　金本　哲男
(74)代理人　100101557
弁理士　萩原　康司
(74)代理人　100128587
弁理士　松本　一騎
(72)発明者　小林　由幸
東京都港区港南１丁目７番１号　ソニー株式会社内

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】

エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データを学習データとして用い、入力された状態データ及び行動データから報酬値を推定する報酬推定機を機械学習により生成する報酬推定機生成部を備え、

前記報酬推定機生成部は、

複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成部と、

前記行動履歴データに含まれる状態データ及び行動データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出部と、

前記特徴量ベクトルから前記行動履歴データに含まれる報酬値を推定する推定関数を回帰／判別学習により算出する推定関数算出部と、

を含み、

前記報酬推定機は、前記複数の基底関数と前記推定関数とにより構成される、

情報処理装置。

【請求項２】

エージェントがおかれた現在の状態を表す状態データと、当該エージェントが次にとりうる行動を表す行動データとを前記報酬推定機に入力し、当該行動をとった結果として当該エージェントが得る報酬値を推定する報酬値推定部と、

前記現在の状態において前記エージェントが次にとりうる行動のうち、前記報酬値推定

部により推定された報酬値が最も高い値となる行動を選択する行動選択部と、
をさらに備える、
請求項１に記載の情報処理装置。
【請求項３】
前記行動選択部による選択結果に基づいてエージェントを行動させる行動制御部と、
前記エージェントの行動に伴って更新される状態データ及び行動データを蓄積し、当該行動の結果としてエージェントが得た報酬を表す報酬値と、蓄積した状態データ及び行動データとを対応付けて前記行動履歴データに追加する履歴データ追加部と、
をさらに備える、
請求項２に記載の情報処理装置。
【請求項４】
前記状態データ、前記行動データ、及び前記報酬値の組が前記行動履歴データに追加された場合、前記特徴量ベクトル算出部は、前記行動履歴データに含まれる全ての状態データ及び行動データについて特徴量ベクトルを算出し、
前記情報処理装置は、特徴領空間において前記特徴量ベクトルにより示される座標点の分布が所定の分布に近づくように前記行動履歴データに含まれる前記状態データ、前記行動データ、及び前記報酬値の組を間引く分布調整部をさらに備える、
請求項３に記載の情報処理装置。
【請求項５】
前記状態データ、前記行動データ、及び前記報酬値の組が前記行動履歴データに追加された場合、前記特徴量ベクトル算出部は、前記行動履歴データに含まれる全ての状態データ及び行動データについて特徴量ベクトルを算出し、
前記情報処理装置は、特徴領空間において前記特徴量ベクトルにより示される座標点の分布が所定の分布に近づくように前記行動履歴データに含まれる前記状態データ、前記行動データ、及び前記報酬値の組のそれぞれに重みを設定する分布調整部をさらに備える、
請求項３に記載の情報処理装置。
【請求項６】
前記分布調整部は、間引き後に残った前記状態データ、前記行動データ、及び前記報酬値の組について、特徴領空間において前記特徴量ベクトルにより示される座標点の分布が所定の分布に近づくように前記行動履歴データに含まれる前記状態データ、前記行動データ、及び前記報酬値の組のそれぞれに重みを設定する、
請求項４に記載の情報処理装置。
【請求項７】
前記行動履歴データを学習データとして用い、現在の時刻においてエージェントがおかれた状態を表す状態データ及び現在の時刻においてエージェントがとる行動を表す行動データから次の時刻におけるエージェントの状態を表す状態データを予測する予測機を機械学習により生成する予測機生成部をさらに備え、
前記報酬値推定部は、
現在の時刻における状態データ及び行動データを前記予測機に入力して次の時刻におけるエージェントの状態を表す状態データを予測し、
前記次の時刻におけるエージェントの状態を表す状態データと、当該状態においてエージェントがとりうる行動を表す行動データとを前記報酬推定機に入力して、当該行動をとった結果として当該エージェントが得る報酬値を推定する、
請求項２～６のいずれか１項に記載の情報処理装置。
【請求項８】
前記行動履歴データを学習データとして用い、現在の時刻においてエージェントがおかれた状態を表す状態データ及び現在の時刻においてエージェントがとる行動を表す行動データから次の時刻におけるエージェントの状態を表す状態データを予測する予測機を機械学習により生成する予測機生成部をさらに備え、
前記報酬値推定部は、現在の時刻を時刻ｔ０とした場合に、

時刻ｔ０における状態データ及び行動データを前記予測機に入力して次の時刻ｔ１におけるエージェントの状態を表す状態データを予測する処理を実行し、
ｋ＝１～ｎ－１（ｎ≧２）について、時刻ｔｋにおける状態データ及び時刻ｔｋにおいてエージェントがとりうる行動を表す行動データを前記予測機に入力して時刻ｔｋ＋１におけるエージェントの状態を表す状態データを予測する処理を逐次実行し、
予測した時刻ｔｎにおけるエージェントの状態を表す状態データと、当該状態においてエージェントがとりうる行動を表す行動データとを前記報酬推定機に入力して、当該行動をとった結果として当該エージェントが得る報酬値を推定する、
請求項２～６のいずれか１項に記載の情報処理装置。
【請求項９】
前記報酬推定機生成部は、複数のエージェントの状態を表す状態データと、当該状態において各エージェントがとった行動を表す行動データと、当該行動の結果として各エージェントが得た報酬を表す報酬値とを含む行動履歴データを学習データとして用い、入力された状態データ及び行動データから報酬値を推定する報酬推定機を機械学習により生成する、
請求項１～８のいずれか１項に記載の情報処理装置。
【請求項１０】
前記基底関数生成部は、遺伝的アルゴリズムに基づいて前記基底関数を更新し、
前記特徴量ベクトル算出部は、前記基底関数が更新された場合に、更新後の前記基底関数に前記状態データ及び前記行動データを入力して特徴量ベクトルを算出し、
前記推定関数算出部は、前記更新後の基底関数を用いて算出された特徴量ベクトルの入力に応じて前記報酬値を推定する推定関数を算出する、
請求項１～９のいずれか１項に記載の情報処理装置。
【請求項１１】
エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データに基づき、高い報酬を得たエージェントがとった行動及び低い報酬を得たエージェントがとらなかった行動に高いスコアを与え、高い報酬を得たエージェントがとらなかった行動及び低い報酬を得たエージェントがとった行動に低いスコアを与える条件で、各状態データに対応する行動毎のスコアを算出するスコア算出部と、
前記行動履歴データ及び前記行動毎のスコアを学習データとして用い、入力された状態データから行動毎のスコアを推定するスコア推定機を機械学習により生成するスコア推定機生成部と、
を備え、
前記スコア推定機生成部は、
複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成部と、
前記行動履歴データに含まれる状態データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出部と、
前記特徴量ベクトルから前記行動毎のスコアを推定する推定関数を回帰／判別学習により算出する推定関数算出部と、
を含み、
前記スコア推定機は、前記複数の基底関数と前記推定関数とにより構成される、
情報処理装置。
【請求項１２】
エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データを学習データとして用い、入力された状態データ及び行動データから報酬値を推定する報酬推定機を機械学習により生成するステップを含み、
前記生成するステップは、
複数の処理関数を組み合わせて複数の基底関数を生成するステップと、

前記行動履歴データに含まれる状態データ及び行動データを前記複数の基底関数に入力して特徴量ベクトルを算出するステップと、
前記特徴量ベクトルから前記行動履歴データに含まれる報酬値を推定する推定関数を回帰／判別学習により算出するステップと、
を含み、
前記報酬推定機は、前記複数の基底関数と前記推定関数とにより構成される、
情報処理方法。
【請求項１３】
エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データに基づき、高い報酬を得たエージェントがとった行動及び低い報酬を得たエージェントがとらなかった行動に高いスコアを与え、高い報酬を得たエージェントがとらなかった行動及び低い報酬を得たエージェントがとった行動に低いスコアを与える条件で、各状態データに対応する行動毎のスコアを算出するステップと、
前記行動履歴データ及び前記行動毎のスコアを学習データとして用い、入力された状態データから行動毎のスコアを推定するスコア推定機を機械学習により生成するステップと、
を含み、
前記生成するステップは、
複数の処理関数を組み合わせて複数の基底関数を生成するステップと、
前記行動履歴データに含まれる状態データを前記複数の基底関数に入力して特徴量ベクトルを算出するステップと、
前記特徴量ベクトルから前記行動毎のスコアを推定する推定関数を回帰／判別学習により算出するステップと、
を含み、
前記スコア推定機は、前記複数の基底関数と前記推定関数とにより構成される、
情報処理方法。
【請求項１４】
エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データを学習データとして用い、入力された状態データ及び行動データから報酬値を推定する報酬推定機を機械学習により生成する報酬推定機生成機能をコンピュータに実現させるためのプログラムであり、
前記報酬推定機生成機能は、
複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成機能と、
前記行動履歴データに含まれる状態データ及び行動データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出機能と、
前記特徴量ベクトルから前記行動履歴データに含まれる報酬値を推定する推定関数を回帰／判別学習により算出する推定関数算出機能と、
を含み、
前記報酬推定機は、前記複数の基底関数と前記推定関数とにより構成される、
プログラム。
【請求項１５】
エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データに基づき、高い報酬を得たエージェントがとった行動及び低い報酬を得たエージェントがとらなかった行動に高いスコアを与え、高い報酬を得たエージェントがとらなかった行動及び低い報酬を得たエージェントがとった行動に低いスコアを与える条件で、各状態データに対応する行動毎のスコアを算出するスコア算出機能と、
前記行動履歴データ及び前記行動毎のスコアを学習データとして用い、入力された状態

データから行動毎のスコアを推定するスコア推定機を機械学習により生成するスコア推定機生成機能と、
をコンピュータに実現させるためのプログラムであり、
前記スコア推定機生成機能は、
複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成機能と、
前記行動履歴データに含まれる状態データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出機能と、
前記特徴量ベクトルから前記行動毎のスコアを推定する推定関数を回帰／判別学習により算出する推定関数算出機能と、
を含み、
前記スコア推定機は、前記複数の基底関数と前記推定関数とにより構成される、
プログラム。

【発明の詳細な説明】

【技術分野】

【０００１】

本技術は、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】

近年、定量的に特徴を決定づけることが難しい任意のデータ群から、そのデータ群の特徴量を機械的に抽出する手法に注目が集まっている。例えば、任意の音楽データを入力とし、その音楽データが属する音楽のジャンルを機械的に抽出するアルゴリズムを自動構築する手法が知られている。ジャズ、クラシック、ポップス等、音楽のジャンルは、楽器の種類や演奏形態に応じて定量的に決まるものではない。そのため、これまでは任意の音楽データが与えられたときに、その音楽データから機械的に音楽のジャンルを抽出することは一般的に難しいと考えられていた。

【０００３】

しかし、実際には、音楽データに含まれる音程の組み合わせ、音程の組み合わせ方、楽器の種類の組み合わせ、メロディーラインやベースラインの構造等、様々な情報の組み合わせの中に、音楽のジャンルを分ける特徴が潜在的に含まれている。そのため、この特徴を抽出するアルゴリズム（以下、特徴量抽出機）を機械学習により自動構築できないか、という観点から特徴量抽出機の研究が行われた。その研究成果の一つとして、例えば、下記の特許文献１に記載された遺伝アルゴリズムに基づく特徴量抽出機の自動構築方法を挙げることができる。遺伝アルゴリズムとは、生物の進化過程に倣い、機械学習の過程で、選択、交差、突然変異の要素を考慮したものを言う。

【０００４】

同文献に記載の特徴量抽出機自動構築アルゴリズムを利用することにより、任意の音楽データから、その音楽データが属する音楽のジャンルを抽出する特徴量抽出機を自動構築することができるようになる。また、同文献に記載の特徴量抽出機自動構築アルゴリズムは、非常に汎用性が高く、音楽データに限らず、任意のデータ群から、そのデータ群の特徴量を抽出する特徴量抽出機を自動構築することができる。そのため、同文献に記載の特徴量抽出機自動構築アルゴリズムは、音楽データや映像データのような人工的なデータの特徴量解析、自然界に存在する様々な観測量の特徴量解析への応用が期待されている。

【先行技術文献】

【特許文献】

【０００５】

【特許文献１】特開２００９－４８２６６号公報

【発明の概要】

【発明が解決しようとする課題】

【０００６】

ところで、本件発明者は、同文献に記載の技術に対して更なる工夫を施すことで、エー

ジェントを賢く行動させるアルゴリズムを自動構築する技術に発展させられないか検討を行ってきた。その検討の中で、本件発明者は、ある状態におかれたエージェントがとりうる行動の中で選択すべき行動を決定するための思考ルーチンを自動構築する技術に注目した。本技術は、このような技術に関するものであり、エージェントがとるべき行動を選択する際に決め手となる情報を出力する推定機を自動構築することが可能な、新規かつ改良された情報処理装置、情報処理方法、及びプログラムを提供することを意図している。

【課題を解決するための手段】

【０００７】

本技術のある観点によれば、エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データを学習データとして用い、入力された状態データ及び行動データから報酬値を推定する報酬推定機を機械学習により生成する報酬推定機生成部を備え、前記報酬推定機生成部は、複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成部と、前記行動履歴データに含まれる状態データ及び行動データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出部と、前記特徴量ベクトルから前記行動履歴データに含まれる報酬値を推定する推定関数を回帰／判別学習により算出する推定関数算出部と、を含み、前記報酬推定機は、前記複数の基底関数と前記推定関数とにより構成される、情報処理装置が提供される。

【０００８】

また、本技術の別の観点によれば、エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データに基づき、高い報酬を得たエージェントがとった行動及び低い報酬を得たエージェントがとらなかった行動に高いスコアを与え、高い報酬を得たエージェントがとらなかった行動及び低い報酬を得たエージェントがとった行動に低いスコアを与える条件で、各状態データに対応する行動毎のスコアを算出するスコア算出部と、前記行動履歴データ及び前記行動毎のスコアを学習データとして用い、入力された状態データから行動毎のスコアを推定するスコア推定機を機械学習により生成するスコア推定機生成部と、を備え、前記スコア推定機生成部は、複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成部と、前記行動履歴データに含まれる状態データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出部と、前記特徴量ベクトルから前記行動毎のスコアを推定する推定関数を回帰／判別学習により算出する推定関数算出部と、を含み、前記スコア推定機は、前記複数の基底関数と前記推定関数とにより構成される、情報処理装置が提供される。

【０００９】

また、本技術の別の観点によれば、エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データを学習データとして用い、入力された状態データ及び行動データから報酬値を推定する報酬推定機を機械学習により生成するステップを含み、前記生成するステップは、複数の処理関数を組み合わせて複数の基底関数を生成するステップと、前記行動履歴データに含まれる状態データ及び行動データを前記複数の基底関数に入力して特徴量ベクトルを算出するステップと、前記特徴量ベクトルから前記行動履歴データに含まれる報酬値を推定する推定関数を回帰／判別学習により算出するステップと、を含み、前記報酬推定機は、前記複数の基底関数と前記推定関数とにより構成される、情報処理方法が提供される。

【００１０】

また、本技術の別の観点によれば、エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データに基づき、高い報酬を得たエージェントがとった行動及び低い報酬を得たエージェントがとらなかった行動に高いスコアを与え、高い報酬を得たエージェントがとらなかった行動及び低い報酬を得たエージェントが

とった行動に低いスコアを与える条件で、各状態データに対応する行動毎のスコアを算出するステップと、前記行動履歴データ及び前記行動毎のスコアを学習データとして用い、入力された状態データから行動毎のスコアを推定するスコア推定機を機械学習により生成するステップと、を含み、前記生成するステップは、複数の処理関数を組み合わせて複数の基底関数を生成するステップと、前記行動履歴データに含まれる状態データを前記複数の基底関数に入力して特徴量ベクトルを算出するステップと、前記特徴量ベクトルから前記行動毎のスコアを推定する推定関数を回帰／判別学習により算出するステップと、を含み、前記スコア推定機は、前記複数の基底関数と前記推定関数とにより構成される、情報処理方法が提供される。

【００１１】

また、本技術の別の観点によれば、エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データを学習データとして用い、入力された状態データ及び行動データから報酬値を推定する報酬推定機を機械学習により生成する報酬推定機生成機能をコンピュータに実現させるためのプログラムであり、前記報酬推定機生成機能は、複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成機能と、前記行動履歴データに含まれる状態データ及び行動データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出機能と、前記特徴量ベクトルから前記行動履歴データに含まれる報酬値を推定する推定関数を回帰／判別学習により算出する推定関数算出機能と、を含み、前記報酬推定機は、前記複数の基底関数と前記推定関数とにより構成される、プログラムが提供される。

【００１２】

また、本技術の別の観点によれば、エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データに基づき、高い報酬を得たエージェントがとった行動及び低い報酬を得たエージェントがとらなかった行動に高いスコアを与え、高い報酬を得たエージェントがとらなかった行動及び低い報酬を得たエージェントがとった行動に低いスコアを与える条件で、各状態データに対応する行動毎のスコアを算出するスコア算出機能と、前記行動履歴データ及び前記行動毎のスコアを学習データとして用い、入力された状態データから行動毎のスコアを推定するスコア推定機を機械学習により生成するスコア推定機生成機能と、をコンピュータに実現させるためのプログラムであり、前記スコア推定機生成機能は、複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成機能と、前記行動履歴データに含まれる状態データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出機能と、前記特徴量ベクトルから前記行動毎のスコアを推定する推定関数を回帰／判別学習により算出する推定関数算出機能と、を含み、前記スコア推定機は、前記複数の基底関数と前記推定関数とにより構成される、プログラムが提供される。

【００１３】

また、本技術の別の観点によれば、上記のプログラムが記録された、コンピュータにより読み取り可能な記録媒体が提供される。

【発明の効果】

【００１４】

以上説明したように本技術によれば、エージェントがとるべき行動を選択する際に決め手となる情報を出力する推定機を自動構築することが可能になる。

【図面の簡単な説明】

【００１５】

【図１】推定機の自動構築方法について説明するための説明図である。
【図２】推定機の自動構築方法について説明するための説明図である。
【図３】推定機の自動構築方法について説明するための説明図である。
【図４】推定機の自動構築方法について説明するための説明図である。

【図５】推定機の自動構築方法について説明するための説明図である。
【図６】推定機の自動構築方法について説明するための説明図である。
【図７】推定機の自動構築方法について説明するための説明図である。
【図８】推定機の自動構築方法について説明するための説明図である。
【図９】推定機の自動構築方法について説明するための説明図である。
【図１０】推定機の自動構築方法について説明するための説明図である。
【図１１】推定機の自動構築方法について説明するための説明図である。
【図１２】推定機の自動構築方法について説明するための説明図である。
【図１３】オンライン学習に基づく推定機の自動構築方法について説明するための説明図である。
【図１４】データセットの統合方法について説明するための説明図である。
【図１５】データセットの統合方法について説明するための説明図である。
【図１６】データセットの統合方法について説明するための説明図である。
【図１７】データセットの統合方法について説明するための説明図である。
【図１８】データセットの統合方法について説明するための説明図である。
【図１９】データセットの統合方法について説明するための説明図である。
【図２０】データセットの統合方法について説明するための説明図である。
【図２１】データセットの統合方法について説明するための説明図である。
【図２２】データセットの統合方法について説明するための説明図である。
【図２３】データセットの統合方法について説明するための説明図である。
【図２４】データセットの統合方法について説明するための説明図である。
【図２５】データセットの統合方法について説明するための説明図である。
【図２６】データセットの統合方法について説明するための説明図である。
【図２７】データセットの統合方法について説明するための説明図である。
【図２８】データセットの統合方法について説明するための説明図である。
【図２９】データセットの統合方法について説明するための説明図である。
【図３０】データセットの統合方法について説明するための説明図である。
【図３１】データセットの統合方法について説明するための説明図である。
【図３２】データセットの統合方法について説明するための説明図である。
【図３３】データセットの統合方法について説明するための説明図である。
【図３４】思考ルーチンの構成について説明するための説明図である。
【図３５】思考ルーチンの構成について説明するための説明図である。
【図３６】思考ルーチンの構成について説明するための説明図である。
【図３７】思考ルーチンの構成について説明するための説明図である。
【図３８】思考ルーチンの構築方法について説明するための説明図である。
【図３９】情報処理装置１０の機能構成例について説明するための説明図である。
【図４０】情報処理装置１０の機能構成例について説明するための説明図である。
【図４１】効率的な報酬推定機の構築方法について説明するための説明図である。
【図４２】効率的な報酬推定機の構築方法について説明するための説明図である。
【図４３】アクションスコア推定機を用いた思考ルーチンの構成について説明するための説明図である。
【図４４】アクションスコア推定機を用いた思考ルーチンの構成について説明するための説明図である。
【図４５】予測機を用いた報酬の推定方法について説明するための説明図である。
【図４６】予測機を用いた報酬の推定方法について説明するための説明図である。
【図４７】予測機を用いた報酬の推定方法について説明するための説明図である。
【図４８】「三目並べ」への応用について説明するための説明図である。
【図４９】「三目並べ」への応用について説明するための説明図である。
【図５０】「三目並べ」への応用について説明するための説明図である。
【図５１】「三目並べ」への応用について説明するための説明図である。

【図５２】「三目並べ」への応用について説明するための説明図である。
【図５３】「三目並べ」への応用について説明するための説明図である。
【図５４】「三目並べ」への応用について説明するための説明図である。
【図５５】「対戦ゲーム」への応用について説明するための説明図である。
【図５６】「対戦ゲーム」への応用について説明するための説明図である。
【図５７】「対戦ゲーム」への応用について説明するための説明図である。
【図５８】「対戦ゲーム」への応用について説明するための説明図である。
【図５９】「対戦ゲーム」への応用について説明するための説明図である。
【図６０】「対戦ゲーム」への応用について説明するための説明図である。
【図６１】「対戦ゲーム」への応用について説明するための説明図である。
【図６２】「対戦ゲーム」への応用について説明するための説明図である。
【図６３】「五目並べ」への応用について説明するための説明図である。
【図６４】「五目並べ」への応用について説明するための説明図である。
【図６５】「ポーカー」への応用について説明するための説明図である。
【図６６】「ポーカー」への応用について説明するための説明図である。
【図６７】「ポーカー」への応用について説明するための説明図である。
【図６８】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図６９】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図７０】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図７１】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図７２】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図７３】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図７４】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図７５】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図７６】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図７７】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図７８】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図７９】「ロールプレイングゲーム」への応用について説明するための説明図である。
【図８０】情報処理装置の機能を実現することが可能なハードウェア構成例について説明するための説明図である。

【発明を実施するための形態】
【００１６】
以下に添付図面を参照しながら、本技術に係る好適な実施の形態について詳細に説明する。なお、本明細書及び図面において、実質的に同一の機能構成を有する構成要素については、同一の符号を付することにより重複説明を省略する。
【００１７】
［説明の流れについて］
ここで、以下に記載する説明の流れについて簡単に述べる。
【００１８】
まず、本実施形態に係る基盤技術について説明する。具体的には、まず、図１～図１２を参照しながら、推定機の自動構築方法について説明する。次いで、図１３を参照しながら、オンライン学習に基づく推定機の自動構築方法について説明する。
【００１９】
次いで、図１４～図１６を参照しながら、データセットの統合方法について説明する。次いで、図１７～図２３を参照しながら、効率的なデータセットのサンプリング方法について説明する。次いで、図２４～図２７を参照しながら、効率的な重み付け方法について説明する。次いで、図２８を参照しながら、効率的なデータセットのサンプリング方法及び重み付け方法を組み合わせる方法について説明する。次いで、図２９～図３３を参照しながら、その他のデータセットのサンプリング方法及び重み付け方法について説明する。
【００２０】

次いで、図３４～図３８を参照しながら、思考ルーチンの構成及び思考ルーチンの構築方法について説明する。次いで、図３９及び図４０を参照しながら、本実施形態に係る情報処理装置１０の機能構成について説明する。次いで、図４１及び図４２を参照しながら、効率的な報酬推定機の構築方法について説明する。次いで、図４３及び図４４を参照しながら、アクションスコア推定機を用いた思考ルーチンの構成について説明する。次いで、図４５～図４７を参照しながら、予測機を用いた報酬の推定方法について説明する。

【００２１】

次いで、図４８～図５４を参照しながら、本実施形態に係る技術を「三目並べ」へ応用する方法について説明する。次いで、図５５～図６２を参照しながら、本実施形態に係る技術を「対戦ゲーム」へ応用する方法について説明する。次いで、図６３及び図６４を参照しながら、本実施形態に係る技術を「五目並べ」へ応用する方法について説明する。次いで、図６５～図６７を参照しながら、本実施形態に係る技術を「ポーカー」へ応用する方法について説明する。次いで、図６８～図７９を参照しながら、本実施形態に係る技術を「ロールプレイングゲーム」へ応用する方法について説明する。

【００２２】

次いで、図８０を参照しながら、本実施形態に係る情報処理装置１０の機能を実現することが可能なハードウェア構成例について説明する。最後に、同実施形態の技術的思想について纏め、当該技術的思想から得られる作用効果について簡単に説明する。

【００２３】

（説明項目）

１：基盤技術

１－１：推定機の自動構築方法

１－１－１：推定機の構成

１－１－２：構築処理の流れ

１－２：オンライン学習について

１－３：学習用データの統合方法

１－３－１：特徴量空間における学習用データの分布と推定機の精度

１－３－２：データ統合時にサンプリングする構成

１－３－３：データ統合時に重み付けする構成

１－３－４：データ統合時にサンプリング及び重み付けする構成

１－４：効率的なサンプリング／重み付け方法

１－４－１：サンプリング方法

１－４－２：重み付け方法

１－４－３：組み合わせ方法

１－５：サンプリング処理及び重み付け処理に関する変形例

１－５－１：変形例１（距離に基づく処理）

１－５－２：変形例２（クラスタリングに基づく処理）

１－５－３：変形例３（密度推定手法に基づく処理）

２：実施形態

２－１：思考ルーチンの自動構築方法

２－１－１：思考ルーチンとは

２－１－２：思考ルーチンの構成

２－１－３：報酬推定機の構築方法

２－２：情報処理装置１０の構成

２－３：効率的な推定報酬機の構築方法

２－４：（変形例１）アクションスコア推定機を用いる思考ルーチン

２－５：（変形例２）予測機を用いた報酬の推定

２－５－１：予測機の構築方法

２－５－２：報酬の推定方法

２－６：（変形例３）複数エージェントの同時学習

【００２４】

＜１：基盤技術＞

後述する実施形態は、推定機の自動構築方法に関する。また、同実施形態は、推定機の構築に用いる学習用データを追加できるようにする仕組み（以下、オンライン学習）に関する。そこで、同実施形態に係る技術について詳細に説明するに先立ち、推定機の自動構築方法及びオンライン学習方法（以下、基盤技術）について説明する。なお、以下では遺伝アルゴリズムに基づく推定機の自動構築方法を例に挙げて説明を進めるが、同実施形態に係る技術の適用範囲はこれに限定されない。

【００２５】

［１－１：推定機の自動構築方法］

推定機の自動構築方法について説明する。

【００２６】

（１－１－１：推定機の構成）

はじめに、図１～図３を参照しながら、推定機の構成について説明する。図１は、推定機を利用するシステムのシステム構成例を示した説明図である。また、図２は、推定機の構築に利用する学習用データの構成例を示した説明図である。そして、図３は、推定機の構造及び構築方法の概要を示した説明図である。

【００２７】

まず、図１を参照する。図１に示すように、推定機の構築及び推定値の算出は、例えば、情報処理装置１０により実行される。情報処理装置１０は、学習用データ（X_1，t_1），…，（X_N，t_N）を利用して推定機を構築する。また、情報処理装置１０は、構築した推定機を利用して入力データＸから推定値ｙを算出する。この推定値ｙは、入力データＸの認識に利用される。例えば、推定値ｙが所定の閾値Ｔｈより大きい場合に認識結果ＹＥＳが得られ、推定値ｙが所定の閾値Ｔｈより小さい場合に認識結果ＮＯが得られる。

【００２８】

図２を参照しながら、より具体的に推定機の構成について考えてみよう。図２に例示した学習用データの集合は、“海”の画像を認識する画像認識機の構築に利用されるものである。この場合、情報処理装置１０により構築される推定機は、入力された画像の“海らしさ”を表す推定値ｙを出力するものとなる。図２に示すように、学習用データは、データX_kと目的変数t_kとのペア（但し、ｋ＝１～Ｎ）により構成される。データＸｋは、ｋ番目の画像データ（画像＃ｋ）である。また、目的変数t_kは、画像＃ｋが“海”の画像である場合に１、画像＃ｋが“海”の画像でない場合に０となる変数である。

【００２９】

図２の例では、画像＃１が“海”の画像であり、画像＃２が“海”の画像であり、…、画像＃Ｎが“海”の画像でない。この場合、$t_1=1$、$t_2=1$、…、$t_N=0$となる。この学習用データが入力されると、情報処理装置１０は、入力された学習用データに基づく機械学習により、入力された画像の“海らしさ”を表す推定値ｙを出力する推定機を構築する。この推定値ｙは、入力された画像の“海らしさ”が高いほど１に近づき、“海らしさ”が低いほど０に近づく値である。

【００３０】

また、新たに入力データＸ（画像Ｘ）が入力されると、情報処理装置１０は、学習用データの集合を利用して構築された推定機に画像Ｘを入力し、画像Ｘの“海らしさ”を表す

推定値ｙを算出する。この推定値ｙを利用すると、画像Ｘが“海”の画像であるか否かを認識することが可能になる。例えば、推定値ｙ≧所定の閾値Ｔｈの場合、入力された画像Ｘが“海”の画像であると認識される。一方、推定値ｙ＜所定の閾値Ｔｈの場合、入力された画像Ｘが“海”の画像でないと認識される。

【００３１】

本実施形態は、上記のような推定機を自動構築する技術に関する。なお、ここでは画像認識機の構築に利用される推定機について説明したが、本実施形態に係る技術は、様々な推定機の自動構築方法に適用することができる。例えば、言語解析機の構築に適用することもできるし、楽曲のメロディーラインやコード進行などを解析する音楽解析機の構築にも適用することができる。さらに、蝶の動きや雲の流れなどの自然現象を再現したり、自然の振る舞いを予測したりする動き予測機の構築などにも適用することができる。

【００３２】

例えば、特開２００９－４８２６６号公報、特願２０１０－１５９５９８号明細書、特願２０１０－１５９５９７号明細書、特願２００９－２７７０８３号明細書、特願２００９－２７７０８４号明細書などに記載のアルゴリズムに適用することができる。また、ＡｄａＢｏｏｓｔなどのアンサンブル学習手法や、ＳＶＭやＳＶＲなどのカーネルを用いた学習手法などにも適用できる。ＡｄａＢｏｏｓｔなどのアンサンブル学習手法に適用する場合、弱学習機（Ｗｅａｋ　Ｌｅａｒｎｅｒ）が後述する基底関数φに対応する。また、ＳＶＭやＳＶＲなどの学習手法に適用する場合、カーネルが後述する基底関数φに対応する。なお、ＳＶＭはＳｕｐｐｏｒｔ　Ｖｅｃｔｏｒ　Ｍａｃｈｉｎｅ、ＳＶＲはＳｕｐｐｏｒｔ　Ｖｅｃｔｏｒ　Ｒｅｇｒｅｓｓｉｏｎ、ＲＶＭはＲｅｌｅｖａｎｃｅ　Ｖｅｃｔｏｒ　Ｍａｃｈｉｎｅの略である。

【００３３】

ここで、図３を参照しながら、推定機の構造について説明する。図３に示すように、推定機は、基底関数リスト（ϕ_1，…，ϕ_M）及び推定関数ｆにより構成される。基底関数リスト（ϕ_1，…，ϕ_M）は、Ｍ個の基底関数ϕ_k（ｋ＝１～Ｍ）を含む。また、基底関数ϕ_kは、入力データＸの入力に応じて特徴量z_kを出力する関数である。さらに、推定関数ｆは、Ｍ個の特徴量z_k（ｋ＝１～Ｍ）を要素として含む特徴量ベクトルＺ＝（z_1，…，z_M）の入力に応じて推定値ｙを出力する関数である。基底関数ϕ_kは、予め用意された１又は複数の処理関数を組み合わせて生成される。

【００３４】

処理関数としては、例えば、三角関数、指数関数、四則演算、デジタルフィルタ、微分演算、中央値フィルタ、正規化演算、ホワイトノイズの付加処理、画像処理フィルタなどが利用可能である。例えば、入力データＸが画像の場合、ホワイトノイズの付加処理ＡｄｄＷｈｉｔｅＮｏｉｓｅ（）、中央値フィルタＭｅｄｉａｎ（）、ぼかし処理Ｂｌｕｒ（）を組み合わせた基底関数ϕ_j（Ｘ）＝ＡｄｄＷｈｉｔｅＮｏｉｓｅ（Ｍｅｄｉａｎ（Ｂｌｕｒ（Ｘ））などが利用される。この基底関数ϕ_jは、入力データＸに対し、ぼかし処理、中央値フィルタ処理、及びホワイトノイズの付加処理を順次施すことを意味する。

【００３５】

（１－１－２：構築処理の流れ）

さて、基底関数ϕ_k（ｋ＝１～Ｍ）の構成、基底関数リストの構成、推定関数ｆの構成は、学習用データに基づく機械学習により決定される。以下、この機械学習による推定機の構築処理について、より詳細に説明する。

【００３６】

（全体構成）

まず、図４を参照しながら、全体的な処理の流れについて説明する。図４は、全体的な処理の流れについて説明するための説明図である。なお、以下で説明する処理は、情報処理装置１０により実行される。

【００３７】

図４に示すように、まず、情報処理装置１０に学習用データが入力される（Ｓ１０１）

。なお、学習用データとしては、データＸと目的変数ｔの組が入力される。学習用データが入力されると、情報処理装置１０は、処理関数を組み合わせて基底関数を生成する（Ｓ１０２）。次いで、情報処理装置１０は、基底関数にデータＸを入力して特徴量ベクトルＺを算出する（Ｓ１０３）。次いで、情報処理装置１０は、基底関数の評価及び推定関数の生成を行う（Ｓ１０４）。

【００３８】

次いで、情報処理装置１０は、所定の終了条件を満たしたか否かを判定する（Ｓ１０５）。所定の終了条件を満たした場合、情報処理装置１０は、処理をステップＳ１０６に進める。一方、所定の終了条件を満たしていない場合、情報処理装置１０は、処理をステップＳ１０２に戻し、ステップＳ１０２～Ｓ１０４の処理を繰り返し実行する。処理をステップＳ１０６に進めた場合、情報処理装置１０は、推定関数を出力する（Ｓ１０６）。上記の通り、ステップＳ１０２～Ｓ１０４の処理は、繰り返し実行される。そこで、以下の説明においては、第τ回目の繰り返し処理においてステップＳ１０２で生成される基底関数を第τ世代の基底関数と呼ぶことにする。

【００３９】

（基底関数の生成（Ｓ１０２））

ここで、図５～図１０を参照しながら、ステップＳ１０２の処理（基底関数の生成）について、より詳細に説明する。

【００４０】

まず、図５を参照する。図５に示すように、情報処理装置１０は、現在の世代が２世代目以降であるか否かを判定する（Ｓ１１１）。つまり、情報処理装置１０は、現在実行しようとしているステップＳ１０２の処理が第２回目以降の繰り返し処理であるか否かを判定する。２世代目以降である場合、情報処理装置１０は、処理をステップＳ１１３に進める。一方、２世代目以降でない場合（第１世代である場合）、情報処理装置１０は、処理をステップＳ１１２に進める。処理をステップＳ１１２に進めた場合、情報処理装置１０は、基底関数をランダムに生成する（Ｓ１１２）。一方、処理をステップＳ１１３に進めた場合、情報処理装置１０は、基底関数を進化的に生成する（Ｓ１１３）。そして、情報処理装置１０は、ステップＳ１１２又はＳ１１３の処理が完了すると、ステップＳ１０２の処理を終了する。

【００４１】

（Ｓ１１２：基底関数をランダムに生成）

次に、図６及び図７を参照しながら、ステップＳ１１２の処理について、より詳細に説明する。ステップＳ１１２の処理は、第１世代の基底関数を生成する処理に関する。

【００４２】

まず、図６を参照する。図６に示すように、情報処理装置１０は、基底関数のインデックスｍ（ｍ＝０～Ｍ－１）に関する処理ループを開始する（Ｓ１２１）。次いで、情報処理装置１０は、基底関数$\phi_m(x)$をランダムに生成する（Ｓ１２２）。次いで、情報処理装置１０は、基底関数のインデックスｍがＭ－１に達したか否かを判定し、基底関数のインデックスｍがＭ－１に達していない場合、情報処理装置１０は、基底関数のインデックスｍをインクリメントしてステップＳ１２１に処理を戻す（Ｓ１２４）。一方、基底関数のインデックスｍがｍ＝Ｍ－１の場合、情報処理装置１０は、処理ループを終了する（Ｓ１２４）。ステップＳ１２４で処理ループを終了すると、情報処理装置１０は、ステップＳ１１２の処理を完了する。

【００４３】

（ステップＳ１２２の詳細）

次に、図７を参照しながら、ステップＳ１２２の処理について、より詳細に説明する。

【００４４】

ステップＳ１２２の処理を開始すると、図７に示すように、情報処理装置１０は、基底関数のプロトタイプをランダムに決定する（Ｓ１３１）。プロトタイプとしては、既に例示した処理関数の他、線形項、ガウシアンカーネル、シグモイドカーネルなどの処理関数

が利用可能である。次いで、情報処理装置１０は、決定したプロトタイプのパラメータをランダムに決定し、基底関数を生成する（Ｓ１３２）。

【００４５】

（Ｓ１１３：基底関数を進化的に生成）

次に、図８～図１０を参照しながら、ステップＳ１１３の処理について、より詳細に説明する。ステップＳ１１３の処理は、第τ世代（τ≧２）の基底関数を生成する処理に関する。従って、ステップＳ１１３を実行する際には、第τ－１世代の基底関数$\phi_{m,\tau-1}$（ｍ＝１～Ｍ）及び当該基底関数$\phi_{m,\tau-1}$の評価値$v_{m,\tau-1}$が得られている。

【００４６】

まず、図８を参照する。図８に示すように、情報処理装置１０は、基底関数の数Ｍを更新する（Ｓ１４１）。つまり、情報処理装置１０は、第τ世代の基底関数の数M_τを決定する。次いで、情報処理装置１０は、第τ－１世代の基底関数$\phi_{m,\tau-1}$（ｍ＝１～Ｍ）に対する評価値$v_{\tau-1}=\{v_{1,\tau-1},\cdots,v_{M,\tau-1}\}$に基づき、第τ－１世代の基底関数の中からｅ個の有用な基底関数を選択して第τ世代の基底関数$\phi_{1,\tau}$、…、$\phi_{e,\tau}$に設定する（Ｓ１４２）。

【００４７】

次いで、情報処理装置１０は、残り（M_τ－ｅ）個の基底関数$\phi_{e+1,\tau}$、…、$\phi_{M_\tau,\tau}$を生成する方法を交差、突然変異、ランダム生成の中からランダムに選択する（Ｓ１４３）。交差を選択した場合、情報処理装置１０は、処理をステップＳ１４４に進める。また、突然変異を選択した場合、情報処理装置１０は、処理をステップＳ１４５に進める。そして、ランダム生成を選択した場合、情報処理装置１０は、処理をステップＳ１４６に進める。

【００４８】

処理をステップＳ１４４に進めた場合、情報処理装置１０は、ステップＳ１４２で選択された基底関数$\phi_{1,\tau}$、…、$\phi_{e,\tau}$の中から選択された基底関数を交差させて新たな基底関数$\phi_{m',,\tau}$（ｍ’≧ｅ＋１）を生成する（Ｓ１４４）。また、処理をステップＳ１４５に進めた場合、情報処理装置１０は、ステップＳ１４２で選択された基底関数$\phi_{1,\tau}$、…、$\phi_{e,\tau}$の中から選択された基底関数を突然変異させて新たな基底関数$\phi_{m',,\tau}$（ｍ’≧ｅ＋１）を生成する（Ｓ１４５）。一方、処理をステップＳ１４６に進めた場合、情報処理装置１０は、ランダムに新たな基底関数$\phi_{m',,\tau}$（ｍ’≧ｅ＋１）を生成する（Ｓ１４６）。

【００４９】

ステップＳ１４４、Ｓ１４５、Ｓ１４６のいずれかの処理を終えると、情報処理装置１０は、処理をステップＳ１４７に進める。処理をステップＳ１４７に進めると、情報処理装置１０は、第τ世代の基底関数がＭ個（Ｍ＝M_τ）に達したか否かを判定する（Ｓ１４７）。第τ世代の基底関数がＭ個に達していない場合、情報処理装置１０は、処理を再びステップＳ１４３に戻す。一方、第τ世代の基底関数がＭ個に達した場合、情報処理装置１０は、ステップＳ１１３の処理を終了する。

【００５０】

（Ｓ１４４の詳細：交差）

次に、図９を参照しながら、ステップＳ１４４の処理について、より詳細に説明する。

【００５１】

ステップＳ１４４の処理を開始すると、図９に示すように、情報処理装置１０は、ステップＳ１４２で選択された基底関数$\phi_{1,\tau}$、…、$\phi_{e,\tau}$の中から同じプロトタイプを持つ基底関数をランダムに２つ選択する（Ｓ１５１）。次いで、情報処理装置１０は、選択した２つの基底関数が持つパラメータを交差させて新たな基底関数を生成する（Ｓ１５２）。

【００５２】

（Ｓ１４５の詳細：突然変異）

次に、図１０を参照しながら、ステップＳ１４５の処理について、より詳細に説明する

。
【００５３】
ステップＳ１４５の処理を開始すると、図１０に示すように、情報処理装置１０は、ステップＳ１４２で選択された基底関数$\phi_{1,\tau}$、…、$\phi_{e,\tau}$の中から基底関数をランダムに１つ選択する（Ｓ１６１）。次いで、情報処理装置１０は、選択した基底関数が持つパラメータの一部をランダムに変更して新たな基底関数を生成する（Ｓ１６２）。
【００５４】
（Ｓ１４６の詳細：ランダム生成）
次に、図７を参照しながら、ステップＳ１４６の処理について、より詳細に説明する。
【００５５】
ステップＳ１２２の処理を開始すると、図７に示すように、情報処理装置１０は、基底関数のプロトタイプをランダムに決定する（Ｓ１３１）。プロトタイプとしては、既に例示した処理関数の他、線形項、ガウシアンカーネル、シグモイドカーネルなどの処理関数が利用可能である。次いで、情報処理装置１０は、決定したプロトタイプのパラメータをランダムに決定し、基底関数を生成する（Ｓ１３２）。
【００５６】
以上、ステップＳ１０２の処理（基底関数の生成）について、より詳細に説明した。
【００５７】
（基底関数の計算（Ｓ１０３））
次に、図１１を参照しながら、ステップＳ１０３の処理（基底関数の計算）について、より詳細に説明する。
【００５８】
図１１に示すように、情報処理装置１０は、学習用データに含まれるｉ番目のデータ$X^{(i)}$のインデックスｉに関する処理ループを開始する（Ｓ１７１）。例えば、学習用データとしてＮ個のデータの組$\{X^{(1)},\cdots,X^{(N)}\}$が入力された場合には、ｉ＝１～Ｎに関して処理ループが実行される。次いで、情報処理装置１０は、基底関数ϕ_mのインデックスｍに関する処理ループを開始する（Ｓ１７２）。例えば、Ｍ個の基底関数を生成した場合には、ｍ＝１～Ｍに関して処理ループが実行される。
【００５９】
次いで、情報処理装置１０は、特徴量$z_{mi}=\phi_m(x^{(i)})$を計算する（Ｓ１７３）。次いで、情報処理装置１０は、処理をステップＳ１７４に進め、基底関数のインデックスｍに関する処理ループを続ける。そして、情報処理装置１０は、基底関数のインデックスｍに関する処理ループが終了すると、処理をステップＳ１７５に進め、インデックスｉに関する処理ループを続ける。インデックスｉに関する処理ループが終了した場合、情報処理装置１０は、ステップＳ１０３の処理を終了する。
【００６０】
以上、ステップＳ１０３の処理（基底関数の計算）について、より詳細に説明した。
【００６１】
（基底関数の評価・推定関数の生成（Ｓ１０４））
次に、図１２を参照しながら、ステップＳ１０４の処理（基底関数の評価・推定関数の生成）について、より詳細に説明する。
【００６２】
図１２に示すように、情報処理装置１０は、ＡＩＣ基準の増減法に基づく回帰／判別学習により推定関数のパラメータ$w=\{w_0,\cdots,w_M\}$を算出する（Ｓ１８１）。つまり、情報処理装置１０は、特徴量$z_{mi}=\phi_{m,\tau}(x^{(i)})$と目的変数$t^{(i)}$の組（ｉ＝１～Ｎ）が推定関数ｆによりフィッティングされるように、回帰／判別学習によりベクトル$w=\{w_0,\cdots,w_M\}$を求める。但し、推定関数ｆ（ｘ）は、$f(x)=\Sigma w_m\phi_{m,\tau}(x)+w_0$であるとする。次いで、情報処理装置１０は、パラメータｗが０となる基底関数の評価値ｖを０に設定し、それ以外の基底関数の評価値ｖを１に設定する（Ｓ１８２）。つまり、評価値ｖが１の基底関数は有用な基底関数である。

【００６３】
以上、ステップＳ１０４の処理（基底関数の評価・推定関数の生成）について、より詳細に説明した。
【００６４】
推定機の構築に係る処理の流れは上記の通りである。このように、ステップＳ１０２～Ｓ１０４の処理が繰り返し実行され、基底関数が進化的手法により逐次更新されることにより推定精度の高い推定関数が得られる。つまり、上記の方法を適用することで、高性能な推定機を自動構築することができる。
【００６５】
［１－２：オンライン学習について］
さて、上記のように、機械学習により推定機を自動構築するアルゴリズムの場合、学習用データの数が多いほど、構築される推定機の性能が高くなる。そのため、可能な限り多くの学習用データを利用して推定機を構築するのが好ましい。また、後述する実施形態に係る技術においては、学習用データを追加する仕組みが利用される。そこで、学習用データを追加できるようにする新たな仕組み（以下、オンライン学習）について紹介する。
【００６６】
オンライン学習に係る推定機の構築は、図１３に示すような処理の流れに沿って行われる。図１３に示すように、まず、学習用データの集合が情報処理装置１０に入力される（Ｓｔｅｐ　１）。次いで、情報処理装置１０は、入力された学習用データの集合を利用し、既に説明した推定機の自動構築方法により推定機を構築する（Ｓｔｅｐ　２）。
【００６７】
次いで、情報処理装置１０は、随時又は所定のタイミングで追加の学習用データを取得する（Ｓｔｅｐ　３）。次いで、情報処理装置１０は、（Ｓｔｅｐ　１）で入力された学習用データの集合に、（Ｓｔｅｐ　３）で取得した学習用データを統合する（Ｓｔｅｐ　４）。このとき、情報処理装置１０は、学習用データのサンプリング処理や重み付け処理を実行し、統合後の学習用データの集合を生成する。そして、情報処理装置１０は、統合後の学習用データの集合を利用し、再び推定機を構築する（Ｓｔｅｐ　２）。このとき、情報処理装置１０は、既に説明した推定機の自動構築方法により推定機を構築する。
【００６８】
また、（Ｓｔｅｐ　２）～（Ｓｔｅｐ　４）の処理は繰り返し実行される。そして、学習用データは、処理が繰り返される度に更新される。例えば、繰り返しの度に学習用データが追加されるようにすれば、推定機の構築処理に利用される学習用データの数が増加するため、推定機の性能が向上する。なお、（Ｓｔｅｐ　４）で実行される学習用データの統合処理においては、情報処理装置１０のリソースをより有効に利用すべく、より有用な学習用データが推定機の構築に利用されるように統合の仕方を工夫する。以下、この工夫について紹介する。
【００６９】
［１－３：学習用データの統合方法］
学習用データの統合方法について、より詳細に説明する。
【００７０】
（１－３－１：特徴量空間における学習用データの分布と推定機の精度）
まず、図１４を参照しながら、特徴量空間における学習用データの分布と推定機の精度との関係について考察する。図１４は、特徴量空間における学習用データの分布例を示した説明図である。
【００７１】
１つの特徴量ベクトルは、１つの学習用データを構成するデータを基底関数リストに含まれる各基底関数に入力することで得られる。つまり、１つの学習用データには１つの特徴量ベクトル（特徴量座標）が対応する。そのため、特徴量座標の分布を特徴量空間における学習用データの分布と呼ぶことにする。特徴量空間における学習用データの分布は、例えば、図１４のようになる。なお、表現の都合上、図１４の例では２次元の特徴量空間

を考えているが、特徴量空間の次元数はこれに限定されない。

【００７２】

さて、図１４に例示した特徴量座標の分布を参照すると、第４象限に疎な領域が存在していることに気づくであろう。既に説明した通り、推定関数は、全ての学習用データについて特徴量ベクトルと目的変数との関係がうまく表現されるように回帰／判別学習により生成される。そのため、特徴量座標の密度が疎な領域について、推定関数は、特徴量ベクトルと目的変数との関係をうまく表現できていない可能性が高い。従って、認識処理の対象となる入力データに対応する特徴量座標が上記の疎な領域に位置する場合、高精度の認識結果を期待することは難しい。

【００７３】

図１５に示すように、学習データセットの数が多くなると疎な領域が生じにくくなり、どの領域に対応する入力データが入力されても高い精度で認識結果を出力することが可能な推定機を構築できるようになると期待される。また、学習データセットの数が比較的少なくても、特徴量座標が特徴量空間において満遍なく分布していれば、高い精度で認識結果を出力することが可能な推定機を構築できるものと期待される。そこで、本件発明者は、学習用データを統合する際に特徴量座標の分布を考慮し、統合後の学習用データの集合に対応する特徴量座標の分布が所定の分布（例えば、一様分布やガウス分布など）となるように調整する仕組みを考案した。

【００７４】

（１－３－２：データ統合時にサンプリングする構成）

まず、図１６を参照しながら、学習用データをサンプリングする方法について説明する。図１６は、学習用データをサンプリングする方法について説明するための説明図である。

【００７５】

既に説明したように、オンライン学習を適用する場合、逐次的に学習用データを追加できるため、多量の学習用データを用いて推定機を構築することが可能になる。しかし、情報処理装置１０のメモリリソースが限られている場合、学習用データの統合時に、推定機の構築に利用する学習用データの数を絞り込む必要がある。このとき、ランダムに学習用データを間引くのではなく、特徴量座標の分布を考慮して学習用データを間引くことで、推定機の精度を低下させることなく、学習用データの数を絞り込むことができる。例えば、図１６に示すように、密な領域に含まれる特徴量座標を多く間引き、疎な領域に含まれる特徴量座標を極力残すようにする。

【００７６】

このような方法で学習用データを間引くことにより、統合後の学習用データの集合に対応する特徴量座標の密度が均一になる。つまり、学習用データの数は少なくなったが、特徴量空間の全体に満遍なく特徴量座標が分布しているため、推定関数の生成時に実行する回帰／判別学習の際に特徴量空間の全体が考慮されることになる。その結果、情報処理装置１０のメモリリソースが限られていても、高い精度で正しい認識結果を推定することが可能な推定機を構築することが可能になる。

【００７７】

（１－３－３：データ統合時に重み付けする構成）

次に、学習用データに重みを設定する方法について説明する。

【００７８】

情報処理装置１０のメモリリソースが限られている場合、学習用データの統合時に学習用データを間引く方法は有効である。一方、メモリリソースに余裕がある場合、学習用データを間引く代わりに、学習用データに重みを設定することで推定機の性能を向上させることが可能になる。例えば、疎な領域に特徴量座標が含まれる学習用データには大きな重みを設定し、密な領域に特徴量座標が含まれる学習用データには小さな重みを設定する。そして、推定関数の生成時に実行する回帰／判別学習の際に各学習用データに設定された重みを考慮するようにする。

【００７９】
（１－３－４：データ統合時にサンプリング及び重み付けする構成）
また、学習用データをサンプリングする方法と、学習用データに重みを設定する方法とを組み合わせてもよい。例えば、特徴量座標の分布が所定の分布となるように学習用データを間引いた後、間引き後の学習用データの集合に属する学習用データに対し、特徴量座標の密度に応じた重みを設定する。このように、間引き処理と重み付け処理とを組み合わせることにより、メモリリソースが限られていても、より高精度の推定機を構築することが可能になる。
【００８０】
［１－４：効率的なサンプリング／重み付け方法］
次に、学習用データの効率的なサンプリング／重み付け方法について説明する。
【００８１】
（１－４－１：サンプリング方法）
まず、図１７を参照しながら、学習用データの効率的なサンプリング方法について説明する。図１７は、学習用データの効率的なサンプリング方法について説明するための説明図である。
【００８２】
図１７に示すように、情報処理装置１０は、全ての学習用データについて特徴量ベクトル（特徴量座標）を算出する（Ｓ２０１）。次いで、情報処理装置１０は、算出した特徴量座標を正規化する（Ｓ２０２）。例えば、情報処理装置１０は、図１８に示すように、各特徴量について、分散が１、平均が０となるように値を正規化する。
【００８３】
次いで、情報処理装置１０は、ランダムにハッシュ関数ｇを生成する（Ｓ２０３）。例えば、情報処理装置１０は、下記の式（１）に示すような５ビットの値を出力するハッシュ関数ｇを複数生成する。このとき、情報処理装置１０は、Ｑ個のハッシュ関数g_q（ｑ＝１～Ｑ）を生成する。但し、関数h_j（ｊ＝１～５）は、下記の式（２）により定義される。また、ｄ及びＴｈｒｅｓｈｏｌｄは、乱数により決定される。
【００８４】
但し、特徴量座標の分布を一様分布に近づける場合、Ｔｈｒｅｓｈｏｌｄの決定に用いる乱数として一様乱数を用いる。また、特徴量座標の分布をガウス分布に近づける場合、Ｔｈｒｅｓｈｏｌｄの決定に用いる乱数としてガウス乱数を用いる。他の分布についても同様である。また、ｄの決定は、z_dの算出に用いた基底関数の寄与率に応じた偏りのある乱数を用いて行われる。例えば、z_dの算出に用いた基底関数の寄与率が大きいほど、ｄの発生する確率が高くなる乱数が用いられる。
【００８５】
【数１】

$$g(Z)=\{h_1(Z),h_2(Z),h_3(Z),h_4(Z),h_5(Z)\} \quad \cdots(1)$$

$$h_j(Z)=\begin{cases}1 & (z_d>Threshold)\\ 0 & (z_d\le Threshold)\end{cases} \quad \cdots(2)$$

【００８６】
ハッシュ関数g_q（ｑ＝１～Ｑ）を生成すると、情報処理装置１０は、各学習用データに対応する特徴量ベクトルＺをハッシュ関数g_qに入力し、ハッシュ値を算出する。そし

て、情報処理装置１０は、算出したハッシュ値に基づいて学習用データをバケットに割り当てる（Ｓ２０４）。但し、ここで言うバケットとは、ハッシュ値として取り得る値が対応付けられた領域を意味する。

【００８７】

例えば、ハッシュ値が５ビット、Ｑ＝２５６の場合について考えてみよう。この場合、バケットの構成は図１９のようになる。図１９に示すように、ハッシュ値が５ビットであるから、１つのハッシュ関数g_qに対し、３２個のバケット（以下、バケットセット）が設けられる。また、Ｑ＝２５６であるから、２５６組のバケットセットが設けられる。この例に沿って、学習用データをバケットに割り当てる方法について説明する。

【００８８】

ある学習用データに対応する特徴量ベクトルＺが与えられると、２５６個のハッシュ関数g_1～g_{256}を用いて２５６個のハッシュ値が算出される。例えば、g_1（Ｚ）＝２（１０進数表示）であった場合、情報処理装置１０は、その学習用データをg_1に対応するバケットセットの中で２に対応するバケットに割り当てる。同様に、g_q（Ｚ）（ｑ＝２～２５６）を算出し、各値に対応するバケットに学習用データを割り当てる。図１９の例では、２種類の学習用データを白丸と黒丸とで表現し、各バケットとの対応関係を模式的に表現している。

【００８９】

このようにして各学習用データをバケットに割り当てると、情報処理装置１０は、所定の順序でバケットから学習用データを１つ選択する（Ｓ２０５）。例えば、情報処理装置１０は、図２０に示すように、左上（ハッシュ関数のインデックスｑが小さく、バケットに割り当てられた値が小さい側）から順にバケットを走査し、バケットに割り当てられた学習用データを１つ選択する。

【００９０】

バケットから学習用データを選択するルールは、図２１に示した通りである。第１に、情報処理装置１０は、空のバケットをスキップする。第２に、情報処理装置１０は、１つの学習用データを選択した場合、同じ学習用データを他の全てのバケットから除く。第３に、情報処理装置１０は、１つのバケットに複数の学習用データが割り当てられている場合にはランダムに１つの学習用データを選択する。なお、選択された学習用データの情報は、情報処理装置１０により保持される。

【００９１】

１つの学習用データを選択した後、情報処理装置１０は、所定数の学習用データを選択し終えたか否かを判定する（Ｓ２０６）。所定数の学習用データを選択し終えた場合、情報処理装置１０は、選択した所定数の学習用データを統合後の学習用データの集合として出力し、学習用データの統合に係る一連の処理を終了する。一方、所定数の学習用データを選択し終えていない場合、情報処理装置１０は、処理をステップＳ２０５に進める。

【００９２】

以上、学習用データの効率的なサンプリング方法について説明した。なお、特徴量空間と上記のバケットとの対応関係は図２２に示したイメージ図のようになる。また、上記の方法により学習用データのサンプリングを行った結果は、例えば、図２３（一様分布の例）のようになる。図２３を参照すると、疎な領域に含まれる特徴量座標は残り、密な領域に含まれる特徴量座標が間引かれていることが分かる。なお、上記のバケットを利用しない場合、学習用データのサンプリングに要する演算負荷は格段に大きくなる点に注意されたい。

【００９３】

（１－４－２：重み付け方法）

次に、図２４を参照しながら、学習用データの効率的な重み付け方法について説明する。図２４は、学習用データの効率的な重み付け方法について説明するための説明図である。

【００９４】

図２４に示すように、情報処理装置１０は、全ての学習用データについて特徴量ベクトル（特徴量座標）を算出する（Ｓ２１１）。次いで、情報処理装置１０は、算出した特徴量座標を正規化する（Ｓ２１２）。例えば、情報処理装置１０は、図２４に示すように、各特徴量について、分散が１、平均が０となるように値を正規化する。

【００９５】

次いで、情報処理装置１０は、ランダムにハッシュ関数ｇを生成する（Ｓ２１３）。例えば、情報処理装置１０は、上記の式（１）に示すような５ビットの値を出力するハッシュ関数ｇを複数生成する。このとき、情報処理装置１０は、Ｑ個のハッシュ関数g_q（ｑ＝１～Ｑ）を生成する。但し、関数h_j（ｊ＝１～５）は、上記の式（２）により定義される。また、ｄ及びＴｈｒｅｓｈｏｌｄは、乱数により決定される。

【００９６】

但し、特徴量座標の分布を一様分布に近づける場合、Ｔｈｒｅｓｈｏｌｄの決定に用いる乱数として一様乱数を用いる。また、特徴量座標の分布をガウス分布に近づける場合、Ｔｈｒｅｓｈｏｌｄの決定に用いる乱数としてガウス乱数を用いる。他の分布についても同様である。また、ｄの決定は、z_dの算出に用いた基底関数の寄与率に応じた偏りのある乱数を用いて行われる。例えば、z_dの算出に用いた基底関数の寄与率が大きいほど、ｄの発生する確率が高くなる乱数が用いられる。

【００９７】

ハッシュ関数g_q（ｑ＝１～Ｑ）を生成すると、情報処理装置１０は、各学習用データに対応する特徴量ベクトルＺをハッシュ関数g_qに入力し、ハッシュ値を算出する。そして、情報処理装置１０は、算出したハッシュ値に基づいて学習用データをバケットに割り当てる（Ｓ２１４）。次いで、情報処理装置１０は、各学習用データについて密度を算出する（Ｓ２１５）。例えば、図２５に示すように、学習データセットがバケットに割り当てられているものとしよう。また、白丸で表現された学習用データに注目する。

【００９８】

この場合、情報処理装置１０は、まず、各ハッシュ関数に対応するバケットセットについて、白丸を含むバケットに割り当てられている学習用データの数をカウントする。例えば、ハッシュ関数g_1に対応するバケットセットを参照すると、白丸を含むバケットに割り当てられている学習用データの数は１である。同様に、ハッシュ関数g_2に対応するバケットセットを参照すると、白丸を含むバケットに割り当てられている学習用データの数は２である。情報処理装置１０は、ハッシュ関数g_1～g_{256}に対応するバケットセットについて、白丸を含むバケットに割り当てられている学習用データの数をカウントする。

【００９９】

そして、情報処理装置１０は、カウントした数の平均値を算出し、算出した平均値を白丸に対応する学習用データの密度とみなす。同様にして、情報処理装置１０は、全ての学習用データの密度を算出する。なお、各学習用データの密度は図２６のＢ図のように表現される。但し、色が濃い部分の密度が高く、色が薄い部分の密度が低い。

【０１００】

さて、全ての学習用データについて密度を算出し終えると、情報処理装置１０は、処理をステップＳ２１７に進める（Ｓ２１６）。ステップＳ２１７に処理を進めた場合、情報処理装置１０は、算出した密度から各学習用データに設定する重みを算出する（Ｓ２１７）。例えば、情報処理装置１０は、密度の逆数を重みに設定する。なお、各学習用データに設定される重みの分布は図２７のＢ図のように表現される。但し、色が濃い部分の重みが大きく、色が薄い部分の重みが小さい。図２７を参照すると、密な領域の重みが小さく、疎な領域の重みが大きくなっていることが分かるであろう。

【０１０１】

上記のようにして各学習用データに設定する重みを算出し終えると、情報処理装置１０は、重み付けに係る一連の処理を終了する。以上、学習用データの効率的な重み付け方法について説明した。なお、上記のバケットを利用しない場合、学習用データの重み付けに

要する演算負荷は格段に大きくなる点に注意されたい。

【０１０２】

（１－４－３：組み合わせ方法）

次に、図２８を参照しながら、上記の効率的なサンプリング方法と効率的な重み付け方法とを組み合わせる方法について説明する。図２８は、上記の効率的なサンプリング方法と効率的な重み付け方法とを組み合わせる方法について説明するための説明図である。

【０１０３】

図２８に示すように、情報処理装置１０は、まず、学習用データのサンプリング処理を実行する（Ｓ２２１）。このサンプリング処理は、図１７に示した処理の流れに沿って実行される。そして、所定数の学習用データが得られると、情報処理装置１０は、得られた学習用データを対象に重み付け処理を実行する（Ｓ２２２）。この重み付け処理は、図２４に示した処理の流れに沿って実行される。なお、サンプリング処理の際に算出した特徴量ベクトルやハッシュ関数を流用してもよい。サンプリング処理及び重み付け処理を実行し終えると、情報処理装置１０は、一連の処理を終了する。

【０１０４】

以上、学習用データの効率的なサンプリング／重み付け方法について説明した。

【０１０５】

［１－５：サンプリング処理及び重み付け処理に関する変形例］

次に、サンプリング処理及び重み付け処理に関する変形例を紹介する。

【０１０６】

（１－５－１：変形例１（距離に基づく処理））

まず、図２９を参照しながら、特徴量座標間の距離に基づく学習用データのサンプリング方法について説明する。図２９は、特徴量座標間の距離に基づく学習用データのサンプリング方法について説明するための説明図である。

【０１０７】

図２９に示すように、情報処理装置１０は、まず、ランダムに１つの特徴量座標を選択する（Ｓ２３１）。次いで、情報処理装置１０は、インデックスｊを１に初期化する（Ｓ２３２）。次いで、情報処理装置１０は、未だ選択されてないＪ個の特徴量座標の中からｊ番目の特徴量座標を対象座標に設定する（Ｓ２３３）。次いで、情報処理装置１０は、既に選択された全ての特徴量座標と対象座標との距離Ｄを算出する（Ｓ２３４）。次いで、情報処理装置１０は、算出した距離Ｄの最小値$Ｄ_{ｍｉｎ}$を抽出する（Ｓ２３５）。

【０１０８】

次いで、情報処理装置１０は、ｊ＝Ｊであるか否かを判定する（Ｓ２３６）。ｊ＝Ｊである場合、情報処理装置１０は、処理をステップＳ２３７に進める。一方、ｊ≠Ｊである場合、情報処理装置１０は、処理をステップＳ２３３に進める。処理をステップＳ２３７に進めた場合、情報処理装置１０は、最小値$Ｄ_{ｍｉｎ}$が最大となる対象座標（特徴量座標）を選択する（Ｓ２３７）。次いで、情報処理装置１０は、ステップＳ２３１及びＳ２３７において選択された特徴量座標の数が所定数に達したか否かを判定する（Ｓ２３８）。

【０１０９】

ステップＳ２３１及びＳ２３７において選択された特徴量座標の数が所定数に達した場合、情報処理装置１０は、選択された特徴量座標に対応する学習用データを統合後の学習用データの集合として出力し、一連の処理を終了する。一方、ステップＳ２３１及びＳ２３７において選択された特徴量座標の数が所定数に達していない場合、情報処理装置１０は、処理をステップＳ２３２に進める。

【０１１０】

以上、特徴量座標間の距離に基づく学習用データのサンプリング方法について説明した。

【０１１１】

（１－５－２：変形例２（クラスタリングに基づく処理））

次に、クラスタリングに基づく学習用データのサンプリング／重み付け方法について説

明する。なお、以下ではサンプリング方法及び重み付け方法についてそれぞれ別々に説明するが、これらの方法を組み合わせてもよい。

【０１１２】

（データセットの選択）

まず、図３０を参照しながら、クラスタリングに基づく学習用データのサンプリング方法について説明する。図３０は、クラスタリングに基づく学習用データのサンプリング方法について説明するための説明図である。

【０１１３】

図３０に示すように、まず、情報処理装置１０は、特徴量ベクトルを所定数のクラスタに分類する（Ｓ２４１）。クラスタリング手法としては、例えば、ｋ－ｍｅａｎｓ法や階層的クラスタリングなどの手法が利用可能である。次いで、情報処理装置１０は、各クラスタから順に１つずつ特徴量ベクトルを選択する（Ｓ２４２）。そして、情報処理装置１０は、選択した特徴量ベクトルに対応する学習用データを統合後の学習用データの集合として出力し、一連の処理を終了する。

【０１１４】

（重みの設定）

次に、図３１を参照しながら、クラスタリングに基づく学習用データの重み付け方法について説明する。図３１は、クラスタリングに基づく学習用データの重み付け方法について説明するための説明図である。

【０１１５】

図３１に示すように、まず、情報処理装置１０は、特徴量ベクトルを所定数のクラスタに分類する（Ｓ２５１）。クラスタリング手法としては、例えば、ｋ－ｍｅａｎｓ法や階層的クラスタリングなどの手法が利用可能である。次いで、情報処理装置１０は、各クラスタの要素数をカウントし、要素数の逆数を算出する（Ｓ２５２）。そして、情報処理装置１０は、算出した要素数の逆数を重みとして出力し、一連の処理を終了する。

【０１１６】

以上、クラスタリングに基づく学習用データのサンプリング／重み付け方法について説明した。

【０１１７】

（１－５－３：変形例３（密度推定手法に基づく処理））

次に、密度推定手法に基づく学習用データのサンプリング／重み付け方法について説明する。なお、以下ではサンプリング方法及び重み付け方法についてそれぞれ別々に説明するが、これらの方法を組み合わせてもよい。

【０１１８】

（データセットの選択）

まず、図３２を参照しながら、密度推定手法に基づく学習用データのサンプリング方法について説明する。図３２は、密度推定手法に基づく学習用データのサンプリング方法について説明するための説明図である。

【０１１９】

図３２に示すように、まず、情報処理装置１０は、特徴量座標の密度をモデル化する（Ｓ２６１）。密度のモデル化には、例えば、ＧＭＭ（Ｇａｕｓｓｉａｎ　Ｍｉｘｔｕｒｅ　Ｍｏｄｅｌ）などの密度推定手法が利用される。次いで、情報処理装置１０は、構築したモデルに基づいて各特徴量座標の密度を算出する（Ｓ２６２）。次いで、情報処理装置１０は、未だ選択されていない特徴量座標の中から、密度の逆数に比例する確率でランダムに特徴量座標を選択する（Ｓ２６３）。

【０１２０】

次いで、情報処理装置１０は、所定数の特徴量座標を選択したか否かを判定する（Ｓ２６４）。所定数の特徴量座標を選択していない場合、情報処理装置１０は、処理をステップＳ２６３に進める。一方、所定数の特徴量座標を選択した場合、情報処理装置１０は、選択した特徴量座標に対応する学習用データを統合後の学習用データの集合として出力し

、一連の処理を終了する。
【０１２１】
（重みの設定）
次に、図３３を参照しながら、密度推定手法に基づく学習用データの重み付け方法について説明する。図３３は、密度推定手法に基づく学習用データの重み付け方法について説明するための説明図である。
【０１２２】
図３３に示すように、まず、情報処理装置１０は、特徴量座標の密度をモデル化する（Ｓ２７１）。密度のモデル化には、例えば、ＧＭＭなどの密度推定手法が利用される。次いで、情報処理装置１０は、構築したモデルに基づいて各特徴量座標の密度を算出する（Ｓ２７２）。そして、情報処理装置１０は、算出した密度の逆数を重みに設定し、一連の処理を終了する。
【０１２３】
以上、密度推定手法に基づく学習用データのサンプリング／重み付け方法について説明した。
【０１２４】
以上、後述する実施形態において利用可能な基盤技術について説明した。但し、後述する実施形態に係る技術は、ここで説明した基盤技術の全てを利用しなくてもよいし、当該基盤技術を変形して利用したり、或いは、他の機械学習アルゴリズムを組み合わせて利用したりしてもよい点に注意されたい。
【０１２５】
＜２：実施形態＞
以下、本技術の一実施形態について説明する。
【０１２６】
［２－１：思考ルーチンの自動構築方法］
本実施形態は、ロボットのようなエージェントの思考ルーチンや様々なゲームに登場するＮＰＣ（Ｎｏｎ－Ｐｌａｙｅｒ　Ｃｈａｒａｃｔｅｒ）の思考ルーチンを自動構築する技術に関する。例えば、本実施形態は、ある状態Ｓにおかれたが次にとる行動ａを決定する思考ルーチンを自動構築する技術に関する。本稿においては、状態Ｓの入力に応じて行動ａを出力するプログラムを思考ルーチンと呼ぶことにする。また、以下では、ＮＰＣの行動ａを決定する思考ルーチンを例に挙げて説明を進めることにする。もちろん、ロボットなどの行動を決定する思考ルーチンも同様に自動構築することが可能である。
【０１２７】
（２－１－１：思考ルーチンとは）
上記の通り、本稿に言う思考ルーチンは、図３４に示すように、状態Ｓの入力に応じて行動ａを出力するプログラムである。なお、状態Ｓとは、ある瞬間に、行動ａを決定すべきＮＰＣがおかれた環境を意味する。例えば、図３４に示すように、２つのＮＰＣ（ＮＰＣ＃１及び＃２）が対戦する対戦ゲームについて考えてみよう。この対戦ゲームは、ＮＰＣ＃１及び＃２がそれぞれヒットポイントを有しており、ダメージを受けるとヒットポイントが減少していく仕組みになっているものとする。この例において、ある瞬間における状態Ｓは、ＮＰＣ＃１及び＃２のヒットポイント及び位置関係になる。
【０１２８】
この状態Ｓが入力されると、思考ルーチンは、ＮＰＣ＃１がＮＰＣ＃２にダメージを与え、最終的にＮＰＣ＃２のヒットポイントを０にできることが期待されるＮＰＣ＃１の行動ａを決定する。例えば、ＮＰＣ＃１のヒットポイントが十分に残っており、ＮＰＣ＃２のヒットポイントが僅かである場合、思考ルーチンは、ＮＰＣ＃１が多少のダメージを受けることを許容してＮＰＣ＃２に素早くダメージを与える行動ａを決定するかもしれない。また、ＮＰＣ＃１のヒットポイントが残り僅かであり、ＮＰＣ＃２のヒットポイントが十分に残っている場合、思考ルーチンは、ＮＰＣ＃１がダメージを受けないようにしつつ、ＮＰＣ＃２にダメージを与える行動ａを決定するだろう。

【０１２９】
これまで、ＮＰＣの行動を決定する思考ルーチンは、熟練した技術者により長い時間をかけて設計されていた。もちろん、ＮＰＣの行動をランダムに決定する思考ルーチンも存在するであろう。しかし、賢いＮＰＣの行動を実現することが可能な思考ルーチンを構築するには、ユーザ操作の分析や環境に応じた最適な行動の研究が欠かせなかった。さらに、こうした分析や研究の結果を踏まえて、環境に応じたＮＰＣの最適な行動を決定するための条件設計を行う必要があった。そのため、思考ルーチンの構築には長い時間と大きな労力とが必要であった。こうした事情を踏まえ、本件発明者は、このような思考ルーチンを人手に依らずに自動構築する技術を開発した。
【０１３０】
（２－１－２：思考ルーチンの構成）
図３５に示すように、本実施形態に係る思考ルーチンは、行動履歴データに基づく思考ルーチンの自動構築技術により生成される。この行動履歴データは、状態Ｓ、行動ａ、報酬ｒにより構成される。例えば、状態Ｓ＝S_1において、ＮＰＣ＃１が行動ａ＝“右へ移動”をとった場合にＮＰＣ＃２からダメージを受けてヒットポイントが０になったとしよう。この場合、行動履歴データは、状態Ｓ＝S_1、行動ａ＝“右へ移動”、報酬ｒ＝“０”となる。このような構成を有する行動履歴データを予め蓄積しておき、この行動履歴データを学習データとする機械学習により思考ルーチンを自動構築することができる。
【０１３１】
本実施形態に係る思考ルーチンは、図３６に示すような構成を有する。図３６に示すように、この思考ルーチンは、状態Ｓの入力に応じてＮＰＣがとりうる行動ａをリストアップし、各行動ａについてＮＰＣが得るであろう報酬ｒの推定値（以下、推定報酬ｙ）を算出する。そして、思考ルーチンは、推定報酬ｙが最も高い行動ａを選択する。なお、推定報酬ｙは、報酬推定機を利用して算出される。この推定報酬機は、状態Ｓ及び行動ａの入力に応じて推定報酬ｙを出力するアルゴリズムである。また、この報酬推定機は、行動履歴データを学習データとする機械学習により自動構築される。例えば、先に紹介した推定機の自動構築方法を応用することにより、報酬推定機を自動構築することができる。
【０１３２】
報酬推定機は、図３７に示すように、基底関数リスト（ϕ_1，…，ϕ_M）及び推定関数ｆにより構成される。基底関数リスト（ϕ_1，…，ϕ_M）は、Ｍ個の基底関数ϕ_k（ｋ＝１～Ｍ）を含む。また、基底関数ϕ_kは、入力データＸ（状態Ｓ及び行動ａ）の入力に応じて特徴量z_kを出力する関数である。さらに、推定関数ｆは、Ｍ個の特徴量z_k（ｋ＝１～Ｍ）を要素として含む特徴量ベクトルＺ＝（z_1，…，z_M）の入力に応じて推定報酬ｙを出力する関数である。基底関数ϕ_kは、予め用意された１又は複数の処理関数を組み合わせて生成される。処理関数としては、例えば、三角関数、指数関数、四則演算、デジタルフィルタ、微分演算、中央値フィルタ、正規化演算などが利用可能である。
【０１３３】
また、本実施形態に係る思考ルーチンの自動構築技術は、自動構築された思考ルーチンを利用してＮＰＣを行動させ、その行動の結果として得られた行動履歴データを追加した行動履歴データを利用して思考ルーチンを更新する。但し、行動履歴データの追加には、先に紹介したオンライン学習に係る技術を利用することができる。
【０１３４】
（２－１－３：報酬推定機の構築方法）
例えば、オンライン学習に係る技術を利用した報酬推定機の構築及び更新は、図３８に示すような処理の流れに沿って行われる。なお、これらの処理は、情報処理装置１０により実行されるものとする。図３８に示すように、まず、行動履歴データが情報処理装置１０に入力される（Ｓｔｅｐ　１）。
【０１３５】
（Ｓｔｅｐ　１）において、情報処理装置１０は、予め設計された簡易な思考ルーチンを用いて行動ａを決定しながらＮＰＣを環境中で振る舞わせ、行動履歴データ（Ｓ，ａ，

ｒ）を得る。この簡易な思考ルーチンは、強化学習の分野においてＩｎｎａｔｅ（赤ちゃんが行う本能的な動きに相当）と呼ばれる。このＩｎｎａｔｅは、ＮＰＣが取り得るアクションの中からランダムに行動を選択するものであってもよい。この場合、Ｉｎｎａｔｅの設計も不要になる。情報処理装置１０は、所定数の行動履歴データが得られるまでＩｎｎａｔｅに基づくＮＰＣの行動を繰り返し実行する。次いで、情報処理装置１０は、入力された行動履歴データを利用し、既に説明した推定機の自動構築方法と同様にして報酬推定機を構築する（Ｓｔｅｐ　２）。

【０１３６】

次いで、情報処理装置１０は、随時又は所定のタイミングで追加の行動履歴データを取得する（Ｓｔｅｐ　３）。次いで、情報処理装置１０は、（Ｓｔｅｐ　１）で入力された行動履歴データと、（Ｓｔｅｐ　３）で取得した行動履歴データとを統合する（Ｓｔｅｐ　４）。このとき、情報処理装置１０は、行動履歴データのサンプリング処理や重み付け処理を実行し、統合後の行動履歴データを生成する。そして、情報処理装置１０は、統合後の行動履歴データを利用し、再び報酬推定機を構築する（Ｓｔｅｐ　２）。また、（Ｓｔｅｐ　２）～（Ｓｔｅｐ　４）の処理は繰り返し実行される。そして、行動履歴データは、処理が繰り返される度に更新される。

【０１３７】

以上、思考ルーチンの自動構築方法について簡単に説明した。ここではＮＰＣの行動を決定する思考ルーチンの自動構築方法について述べたが、行動履歴データの構成を変えることで様々な種類の思考ルーチンを同じように自動構築することができる。つまり、本実施形態の技術を適用することにより、統一的な仕組みで様々な思考ルーチンを構築できるようになる。また、自動的に思考ルーチンが構築されるため、思考ルーチンの構築に人が時間を費やさずに済み、労力が大幅に軽減される。

【０１３８】

［２－２：情報処理装置１０の構成］

ここで、図３９及び図４０を参照しながら、本実施形態に係る情報処理装置１０の機能構成について説明する。図３９は、本実施形態に係る情報処理装置１０の全体的な機能構成を示した説明図である。一方、図４０は、本実施形態に係る情報処理装置１０を構成する報酬推定機構築部１２の詳細な機能構成を示した説明図である。

【０１３９】

（全体的な機能構成）

まず、図３９を参照しながら、全体的な機能構成について説明する。図３９に示すように、情報処理装置１０は、主に、行動履歴データ取得部１１と、報酬推定機構築部１２と、入力データ取得部１３と、行動選択部１４とにより構成される。

【０１４０】

思考ルーチンの構築処理が開始されると、行動履歴データ取得部１１は、報酬推定機の構築に利用する行動履歴データを取得する。例えば、行動履歴データ取得部１１は、簡易な思考ルーチン（Ｉｎｎａｔｅ）に基づいて繰り返しＮＰＣを行動させ、所定数の行動履歴データを取得する。但し、行動履歴データ取得部１１は、記憶装置（非図示）に予め格納された行動履歴データを読み出したり、或いは、行動履歴データを提供するシステムなどからネットワークを介して行動履歴データを取得したりしてもよい。

【０１４１】

行動履歴データ取得部１１により取得された行動履歴データは、報酬推定機構築部１２に入力される。行動履歴データが入力されると、報酬推定機構築部１２は、入力された行動履歴データに基づく機械学習により報酬推定機を構築する。例えば、報酬推定機構築部１２は、既に説明した遺伝アルゴリズムに基づく推定機の自動構築方法を利用して報酬推定機を構築する。また、行動履歴データ取得部１１から追加の行動履歴データが入力された場合、報酬推定機構築部１２は、行動履歴データを統合し、統合後の行動履歴データを利用して報酬推定機を構築する。

【０１４２】

報酬推定機構築部１２により構築された報酬推定機は、行動選択部１４に入力される。この報酬推定機は、任意の入力データ（状態Ｓ）に対して最適な行動を選択するために利用される。入力データ取得部１３により入力データ（状態Ｓ）が取得されると、取得された入力データは、行動選択部１４に入力される。入力データが入力されると、行動選択部１４は、入力された入力データが示す状態Ｓ及び状態ＳにおいてＮＰＣがとりうる行動ａを報酬推定機に入力し、報酬推定機から出力される推定報酬ｙに基づいて行動ａを選択する。例えば、図３６に示すように、行動選択部１４は、推定報酬ｙが最も高くなる行動ａを選択する。

【０１４３】

以上、情報処理装置１０の全体的な機能構成について説明した。

【０１４４】

（報酬推定機構築部１２の機能構成）

次に、図４０を参照しながら、報酬推定機構築部１２の機能構成について詳細に説明する。図４０に示すように、報酬推定機構築部１２は、基底関数リスト生成部１２１と、特徴量計算部１２２と、推定関数生成部１２３と、行動履歴データ統合部１２４とにより構成される。

【０１４５】

思考ルーチンの構築処理が開始されると、まず、基底関数リスト生成部１２１は、基底関数リストを生成する。そして、基底関数リスト生成部１２１により生成された基底関数リストは、特徴量計算部１２２に入力される。また、特徴量計算部１２２には、行動履歴データが入力される。基底関数リスト及び行動履歴データが入力されると、特徴量計算部１２２は、入力された行動履歴データを基底関数リストに含まれる各基底関数に入力して特徴量を算出する。特徴量計算部１２２により算出された特徴量の組（特徴量ベクトル）は、推定関数生成部１２３に入力される。

【０１４６】

特徴量ベクトルが入力されると、推定関数生成部１２３は、入力された特徴量ベクトル及び行動履歴データを構成する報酬値ｒに基づいて回帰／判別学習により推定関数を生成する。なお、遺伝アルゴリズムに基づく推定機の構築方法を適用する場合、推定関数生成部１２３は、生成した推定関数に対する各基底関数の寄与率（評価値）を算出し、その寄与率に基づいて終了条件を満たすか否かを判定する。終了条件を満たす場合、推定関数生成部１２３は、基底関数リスト及び推定関数を含む報酬推定機を出力する。

【０１４７】

一方、終了条件を満たさない場合、推定関数生成部１２３は、生成した推定関数に対する各基底関数の寄与率を基底関数リスト生成部１２１に通知する。この通知を受けた基底関数リスト生成部１２１は、遺伝アルゴリズムにより各基底関数の寄与率に基づいて基底関数リストを更新する。基底関数リストを更新した場合、基底関数リスト生成部１２１は、更新後の基底関数リストを特徴量計算部１２２に入力する。更新後の基底関数リストが入力された場合、特徴量計算部１２２は、更新後の基底関数リストを用いて特徴量ベクトルを算出する。そして、特徴量計算部１２２により算出された特徴量ベクトルは、推定関数生成部１２３に入力される。

【０１４８】

上記のように、遺伝アルゴリズムに基づく推定機の構築方法を適用する場合、終了条件が満たされるまで、推定関数生成部１２３による推定関数の生成処理、基底関数リスト生成部１２１による基底関数リストの更新処理、及び特徴量計算部１２２による特徴量ベクトルの算出処理が繰り返し実行される。そして、終了条件が満たされた場合、推定関数生成部１２３から報酬推定機が出力される。

【０１４９】

また、追加の行動履歴データが入力されると、入力された追加の行動履歴データは、特徴量計算部１２２及び行動履歴データ統合部１２４に入力される。追加の行動履歴データが入力されると、特徴量計算部１２２は、追加の行動履歴データを基底関数リストに含ま

れる各基底関数に入力して特徴量を生成する。そして、追加の行動履歴データに対応する特徴量ベクトル及び既存の行動履歴データに対応する特徴量ベクトルは、行動履歴データ統合部１２４に入力される。なお、行動履歴データ統合部１２４には、既存の行動履歴データも入力されているものとする。

【０１５０】

行動履歴データ統合部１２４は、先に紹介したデータセットの統合方法を応用して既存の行動履歴データと追加の行動履歴データとを統合する。例えば、行動履歴データ統合部１２４は、特徴量空間において特徴量ベクトルにより示される座標（特徴量座標）の分布が所定の分布となるように行動履歴データを間引いたり、行動履歴データに重みを設定したりする。行動履歴データを間引いた場合、間引き後の行動履歴データが統合後の行動履歴データとして利用される。一方、行動履歴データに重みを設定した場合、推定関数生成部１２３による回帰／判別学習の際に各行動履歴データに設定された重みが考慮される。

【０１５１】

行動履歴データが統合されると、統合後の行動履歴データを用いて報酬推定機の自動構築処理が実行される。具体的には、行動履歴データ統合部１２４から推定関数生成部１２３に統合後の行動履歴データと、統合後の行動履歴データに対応する特徴量ベクトルとが入力され、推定関数生成部１２３により推定関数が生成される。また、遺伝アルゴリズムに基づく推定機の構築方法を適用する場合、統合後の行動履歴データを利用して推定関数の生成、寄与率の算出、基底関数リストの更新などの処理が実行される。

【０１５２】

以上、報酬推定機構築部１２の詳細な機能構成について説明した。

【０１５３】

以上、本実施形態に係る情報処理装置１０の構成について説明した。上記の構成を適用することにより、任意の状態ＳからＮＰＣがとるべき次の行動ａを決定する思考ルーチンを自動構築することができる。また、この思考ルーチンを利用して賢くＮＰＣを行動させることが可能になる。なお、利用する行動履歴データを変えることで、ロボットなどのエージェントについても同様に賢く行動させることが可能になる。

【０１５４】

［２－３：効率的な推定報酬機の構築方法］

これまで、先に紹介した推定機の自動構築方法に基づく思考ルーチンの自動構築方法について説明してきた。確かに、この方法を適用すると、思考ルーチンを自動構築することが可能になる。しかし、賢く行動するＮＰＣの思考ルーチンを自動構築するには、ある程度長い時間をかけて学習処理を繰り返し実行する必要がある。そこで、本件発明者は、より効率良く高性能な推定報酬機を構築する方法を考案した。

【０１５５】

以下、図４１及び図４２を参照しながら、効率的な推定報酬機の構築方法について説明する。この方法は、より学習効率の高い行動履歴データを取得する方法に関する。より学習効率の高い行動履歴データとは、より推定報酬が高く、より推定誤差が大きく、かつ、特徴量空間における密度が疎な領域にある特徴量座標に対応するデータである。そこで、図４２に示す３つのスコアを導入する。１つ目は、推定報酬が高いほど大きな値となる報酬スコアである。２つ目は、特徴量空間における密度が疎であるほど大きな値となる未知スコアである。３つ目は、推定誤差が大きいほど大きな値となる誤差スコアである。

【０１５６】

例えば、図４１に示した行動a_1、a_2、a_3に注目しよう。仮に、鎖線で囲まれた領域は、推定誤差の小さい領域であるとする。また、図の右上方向に向かうにつれて推定報酬が高くなっているとする。この場合、行動a_1は、報酬スコアが比較的高く、未知スコアが比較的高く、誤差スコアが比較的低い行動であると言える。また、行動a_2は、報酬スコアが比較的低く、未知スコアが比較的低く、誤差スコアが比較的高い行動であると言える。そして、行動a_3は、報酬スコアが比較的高く、未知スコアが比較的高く、誤差スコアが比較的高い行動であると言える。

【０１５７】
より報酬スコアの高い行動を優先的に選択することにより、高い報酬を実現するために必要な行動履歴データを収集することができる。また、より未知スコアが高いか、より誤差スコアが高い行動を優先的に選択することにより、その行動を選択した結果が不定であるような行動履歴データを収集することができる。例えば、図４１の例では、行動a_3を選択することにより、より高い報酬を得られることが期待され、かつ、その行動を選択した結果が不定であるような行動履歴データを収集できると考えられる。図３８に示した処理のうち、（Ｓｔｅｐ　１）及び／又は（Ｓｔｅｐ　３）において上記の方法による行動履歴データの取得を行うことで、（Ｓｔｅｐ　２）における報酬推定機の構築をより効率的に実現することが可能になる。

【０１５８】
以上、効率的な推定報酬機の構築方法について説明した。

【０１５９】
［２－４：（変形例１）アクションスコア推定機を用いる思考ルーチン］
さて、これまでは報酬推定機を用いて報酬を推定し、推定した報酬に基づいて行動を選択する思考ルーチンについて考えてきた。ここでは、図４４に示すように、アクションスコア推定機を用いてアクションスコアを推定し、推定したアクションスコアに基づいて行動を選択する思考ルーチンについて考えてみたい。ここで言うアクションスコアとは、とりうる各行動に対応付けられたスコアであり、対応する行動をとることで好ましい結果が得られる確率の高さを表す。

【０１６０】
アクションスコアを利用する場合、行動履歴データは、図４３に示すような形で与えられる。まず、情報処理装置１０は、これまで説明してきた行動履歴データと同様にして状態Ｓ、行動ａ、報酬ｒの組を収集する。その後、情報処理装置１０は、報酬ｒに基づいてアクションスコアを算出する。

【０１６１】
例えば、状態Ｓ＝S_1において行動ａ＝“Ｒ（右へ移動）”をとった場合に報酬ｒ＝“０”が得られたものとしよう。この場合、行動ａ＝“Ｒ”に対応するアクションスコアは“０”となり、それ以外の行動（“Ｌ”“Ｎ”“Ｊ”）に対応するアクションスコアは“１”となる。その結果、状態Ｓ＝S_1及び行動ａ＝“Ｒ”に対応するアクションスコア（Ｒ，Ｌ，Ｎ，Ｊ）＝（０，１，１，１）が得られる。

【０１６２】
また、状態Ｓ＝S_2において行動ａ＝“Ｌ（左へ移動）”をとった場合に報酬ｒ＝“１”が得られたものとしよう。この場合、行動ａ＝“Ｌ”に対応するアクションスコアは“１”となり、それ以外の行動（“Ｒ”“Ｎ”“Ｊ”）に対応するアクションスコアは“０”となる。その結果、状態Ｓ＝S_2及び行動ａ＝“Ｌ”に対応するアクションスコア（Ｒ，Ｌ，Ｎ，Ｊ）＝（０，１，０，０）が得られる。

【０１６３】
上記のようにして得られた状態Ｓ、行動ａ、アクションスコアの組を行動履歴データとして利用すると、機械学習により、状態Ｓの入力に応じてアクションスコアの推定値を算出するアクションスコア推定機が得られる。例えば、遺伝アルゴリズムに基づく推定機の自動構築方法を適用すれば、高性能なアクションスコア推定機を自動構築することができる。また、行動履歴データを収集する際に、効率的な報酬推定機の構築方法と同様の方法を用いれば、効率的にアクションスコア推定機を自動構築することができる。

【０１６４】
アクションスコア推定機を用いる場合、思考ルーチンの構成は図４４のようになる。つまり、状態Ｓを思考ルーチンに入力すると、思考ルーチンは、アクションスコア推定機に状態Ｓを入力し、アクションスコアの推定値を算出する。そして、思考ルーチンは、アクションスコアの推定値が最も高い行動を選択する。例えば、図４４に示すように、アクションスコアの推定値が（Ｒ，Ｌ，Ｊ，Ｎ）＝（０．６，０．３，０．４，０．２）であっ

た場合、思考ルーチンは、推定値“０．６”に対応する行動“Ｒ”を選択する。
【０１６５】
以上、アクションスコア推定機を用いる思考ルーチンについて説明した。
【０１６６】
［２－５：（変形例２）予測機を用いた報酬の推定］
次に、予測機を用いた報酬の推定方法について説明する。なお、ここで言う予測機とは、ある時刻t_1における状態Ｓ（t_1）及び状態Ｓ（t_1）においてＮＰＣがとった行動ａ（t_1）を入力した場合に、次の時刻t_2における状態Ｓ（t_2）を出力するアルゴリズムのことを意味する。
【０１６７】
（２－５－１：予測機の構築方法）
上記の予測機は、図４５に示すような方法で構築される。図４５に示すように、時刻毎に取得された行動履歴データが学習データとして利用される。例えば、時刻t_2において状態S_2にあるＮＰＣが何もしなかった場合に好ましい結果が得られた場合、行動履歴データは、時刻ｔ＝t_2、状態Ｓ＝S_2、行動ａ＝“何もせず”、報酬ｒ＝“１”となる。なお、予測機の自動構築方法については、特願２００９－２７７０８４号明細書に詳しく記載されている。同明細書には、ある時点までの観測値から将来の時点における観測値を予測する予測機を機械学習により自動構築する方法が記載されている。
【０１６８】
（２－５－２：報酬の推定方法）
上記の予測機を利用すると、図４６に示すように、将来得るであろう報酬を推定することが可能になる。例えば、時刻ｔにおいて状態Ｓ（ｔ）にあるＮＰＣが行動ａ（ｔ）をとった場合に時刻ｔ＋１において実現される状態Ｓ（ｔ＋１）を予測し、その状態Ｓ（ｔ＋１）においてＮＰＣがとりうる行動毎に推定報酬ｙを算出することができるようになる。そのため、時刻ｔ＋１において推定される報酬に基づいて時刻ｔにおいてＮＰＣがとるべき行動を選択することができるようになる。また、図４７に示すように、予測機を繰り返し用いて数ステップ先の状態Ｓ（ｔ＋ｑ）から推定される推定報酬ｙを算出することもできる。この場合、各時刻においてＮＰＣがとりうる行動の組み合わせを考慮し、最終的に最も高い推定報酬が得られる行動の組み合わせを選択することができるようになる。
【０１６９】
以上、予測機を用いた報酬の推定方法について説明した。
【０１７０】
［２－６：（変形例３）複数エージェントの同時学習］
さて、これまでは１つのＮＰＣに注目して最適な行動を選択する思考ルーチンの構築方法について考えてきた。しかし、２つ以上のＮＰＣがとる行動を同時に考慮して思考ルーチンを構築することも可能である。２つのＮＰＣが同じ環境中で行動する場合、両ＮＰＣがとる行動は状態Ｓに反映される。そのため、この方法を適用すると、他のＮＰＣが最も高い推定報酬を見込める行動を選択して行動する環境中において、自身のＮＰＣが最も高い推定報酬を見込める行動を選択するような思考ルーチンを自動構築することができる。例えば、ＭｉｎＭａｘ法などを用いることにより、このような思考ルーチンの自動構築が実現される。以上、複数エージェントの同時学習について説明した。
【０１７１】
以上、本技術の一実施形態について説明した。
【０１７２】
＜３：応用例＞
次に、本実施形態の技術を具体的に応用する方法について紹介する。
【０１７３】
［３－１：「三目並べ」への応用］
まず、図４８～図５４を参照しながら、本実施形態に係る技術を「三目並べ」へ応用する方法について説明する。図４８に示すように、「三目並べ」の主なルールは、（１）交

互に手を打つ、（２）先に３つのマークが１列に並んだ方が勝ち、の２点である。また、「三目並べ」において、状態Ｓは盤面であり、行動ａは各プレーヤが打つ手である。

【０１７４】

「三目並べ」は、互いに最適な手を打つと必ず引き分けになることが知られている。このような完全情報ゲームに用いられる思考ルーチンの多くは、静的評価関数と先読みアルゴリズムとにより構成されている。この静的評価関数は、ある局面の有利／不利を数値化する関数である。例えば、図４９に示すような局面が与えられた場合、静的評価関数は、その局面の有利／不利を表す数値ｙ（“○不利”：－１、“どちらでもない”：０、“○有利”：＋１など）を出力する。本実施形態の場合、この静的評価関数の機能は、報酬推定機により実現される。

【０１７５】

また、先読みアルゴリズムは、先の手を読み、将来の静的評価関数の出力値がより高くなるような手を選択するアルゴリズムである。例えば、先読みアルゴリズムは、ＭｉｎＭａｘ法などを利用して実現される。例えば、図５０に示すように、先読みアルゴリズムは、自分の手番で手を打った後に、相手の手番で相手が打つ可能性のある手を想定し、想定した各手に対して自分が打てる手を想定して、自分が最も有利になる手を選択する。

【０１７６】

ところで、上記のような静的評価関数は、これまで人手により設計されていた。例えば、将棋のＡＩとして有名なボナンザでさえ、静的評価関数で考慮する局面の特徴などの設計事項は人手により設計されていた。また、ゲームの種類が変わると、特徴量の設計も変更する必要がある。そのため、これまでは試行錯誤を繰り返しながら静的評価関数をゲーム毎に人手で設計する必要があった。しかし、本実施形態に係る技術を適用すると、人手による設計作業を省いて思考ルーチンを自動構築することが可能になる。

【０１７７】

「三目並べ」の場合、図５１に示すように、状態Ｓ及び行動ａを３×３のマトリックスで表現する。但し、状態Ｓは、自分の手番となた時点の盤面を表す。また、自分の手番で打った手を反映した盤面を（Ｓ，ａ）と表現する。さらに、自分の手を“１”、相手の手を“－１”、空白を“０”と表現する。つまり、盤面及び手を数値で表現する。このようにして盤面及び手が数値で表現できると、本実施形態に係る報酬推定機の自動構築方法を用いて思考ルーチンを自動構築することが可能になる。

【０１７８】

例えば、情報処理装置１０は、まず、ランダムな場所に自分の手と相手の手とを打つＩｎｎａｔｅを利用して行動履歴データを生成する。上記の通り、（Ｓ，ａ）は、３×３マトリックスにより表現される。また、情報処理装置１０は、図５２に示すように、勝ちに至るまでに打った全ての手に対応する（Ｓ，ａ）に報酬“１”を与える。一方、情報処理装置１０は、図５３に示すように、負けに至るまでに打った全ての手に対応する（Ｓ，ａ）に報酬“－１”を与える。このようにして行動履歴データを蓄積すると、情報処理装置１０は、蓄積した行動履歴データを利用して報酬推定機を構築する。

【０１７９】

実際に手を選択する場合、情報処理装置１０は、図５４に示すように、報酬推定機を利用して現在の状態Ｓから推定報酬ｙを算出し、推定報酬ｙが最大となる手を選択する。図５４の例では、最大の推定報酬に対応する手（Ｃ）が選択される。なお、図５４の例では１手先の報酬を評価して手の選択を行っているが、対戦相手についても同じように推定報酬を算出し、ＭｉｎＭａｘ法などを用いて数手先読みした結果を用いて現在の手を選択するように構成してもよい。

【０１８０】

また、学習により得られた報酬推定機を用いて常に最適な行動を選択するように構成すると、ＮＰＣによる手の選択が毎回同じになってしまうことがある。そこで、推定報酬を算出する工程に何らかのランダムネスを加えてもよい。例えば、報酬推定機により算出した推定報酬に僅かだけ乱数を加える方法が考えられる。また、遺伝アルゴリズムに基づく

機械学習により報酬推定機を算出している場合、学習世代毎に算出される報酬推定機を保持しておき、利用する報酬推定機をランダムに切り替えるように構成してもよい。

【０１８１】

以上、「三目並べ」への応用について説明した。

【０１８２】

［３－２：「対戦ゲーム」への応用］

次に、図５５～図６２を参照しながら、本実施形態に係る技術を「対戦ゲーム」へと応用する方法について説明する。ここで考える「対戦ゲーム」の主なルールは、図５５に示すように、（１）２人対戦ゲームであること、（２）各プレーヤの行動は「左移動」「右移動」「左右移動なし」「ジャンプ」「ジャンプなし」の組み合わせであること、（３）相手のプレーヤを踏んだらＹ軸方向の加速度差に応じて相手にダメージを与えられること、の３点である。また、ヒットポイントが０になったプレーヤが負けである。なお、「対戦ゲーム」への応用には、先に説明したアクションスコア推定機を用いる思考ルーチンの構築方法が用いられる。

【０１８３】

この場合、状態Ｓとしては、自分の絶対座標、相手の絶対座標、時刻が利用される。そのため、状態Ｓは、図５６に示すように３次元マトリックスにより表現される。また、ここでは、３次元マトリックスで表現される状態Ｓの入力に応じて５つの要素（Ｎ，Ｌ，Ｒ，Ｊ，ＮＪ）を持つアクションスコアを推定するアクションスコア推定機の自動構築方法について考える。但し、要素Ｎは、行動ａ＝“左右移動なし”に対応するアクションスコアである。また、要素Ｌは、行動ａ＝“左移動”に対応するアクションスコアである。要素Ｒは、行動ａ＝“右移動”に対応するアクションスコアである。要素Ｊは、行動ａ＝“ジャンプ”に対応するアクションスコアである。要素ＮＪは、行動ａ＝“ジャンプなし”に対応するアクションスコアである。

【０１８４】

行動履歴データを収集するためのＩｎｎａｔｅとしては、例えば、完全にランダムにプレーヤの行動を選択するものが用いられる。例えば、このＩｎｎａｔｅは、Ｎ（左右移動なし）、Ｌ（左移動）、Ｒ（右移動）の中から１つの行動をランダムに選び、選んだ行動に組み合わせる行動をＪ（ジャンプ）又はＮＪ（ジャンプなし）からランダムに１つ選ぶ。また、情報処理装置１０は、図５７に示すように、自分が相手にダメージを与えた時点で、前回自分又は相手がダメージを受けた時点から現時点までの行動履歴データの報酬を１に設定する。一方、自分が相手からダメージを受けた場合、情報処理装置１０は、図５７に示すように、前回自分又は相手がダメージを受けた時点から現時点までの行動履歴データの報酬を０に設定する。

【０１８５】

なお、報酬が１に設定された行動履歴データについて、情報処理装置１０は、実際に行った行動のアクションスコアを１、行わなかった行動のアクションスコアを０に設定する。一方、報酬が０に設定された行動履歴データについて、情報処理装置１０は、実際に行った行動のアクションスコアを０に設定し、行わなかった行動のアクションスコアを０に設定する。このような処理を繰り返すことにより、状態Ｓとアクションスコアとで構成される図５７に示すような行動履歴データが得られる。

【０１８６】

行動履歴データが得られると、情報処理装置１０は、図５８に示した処理の流れに沿って思考ルーチンを構築する。図５８に示すように、行動履歴データを取得すると（Ｓ３０１）、情報処理装置１０は、取得した行動履歴データを利用した機械学習により思考ルーチンを構築する（Ｓ３０２）。次いで、情報処理装置１０は、必要に応じて追加の行動履歴データを取得する（Ｓ３０３）。次いで、情報処理装置１０は、追加した行動履歴データと元の行動履歴データとを統合する（Ｓ３０４）。次いで、情報処理装置１０は、終了条件を満たしたか否かを判定する（Ｓ３０５）。

【０１８７】

例えば、ユーザによる終了操作が与えられた場合や、ランダムに行動するプレーヤに対する勝率が所定の閾値を越えた場合などに、情報処理装置１０は、終了条件を満たしたと判定する。終了条件を満たしていない場合、情報処理装置１０は、処理をステップＳ３０２に進める。一方、終了条件を満たした場合、情報処理装置１０は、思考ルーチンの構築に係る一連の処理を終了する。

【０１８８】

このようにして自動構築された思考ルーチンを用いてプレーヤを行動させた結果、ランダムに行動するプレーヤに対する勝率について、図５９に示すような結果が得られた。図５９に示すように、１５世代（図５８のステップＳ３０２～Ｓ３０４の繰り返し回数が１５）で思考ルーチンを利用して行動するプレーヤの勝率が１００％近くに達した。なお、行動の選択は、最もアクションスコアの高い行動を選択する方法で行われている。但し、この例では、行動を選択する際に、各アクションスコアに僅かの乱数を加えてから行動を選択するようにしている。

【０１８９】

また、先に説明した複数エージェントの同時学習を適用し、２人のプレーヤの行動を同時に学習して思考ルーチンを構築してみた。複数エージェントの同時学習を適用すると、ランダムでない動きをするプレーヤに対して勝とうとする思考ルーチンが自動構築されるため、より賢くプレーヤを行動させる思考ルーチンが構築される。なお、互いに思考ルーチンを用いて行動する２人のプレーヤを対戦させた結果を図６０に示した。図６０に示すように、学習世代によりプレーヤ１が大きく勝ち越す場合もあるが、プレーヤ２が大きく勝ち越す場合もある。

【０１９０】

また、ある学習世代において実験的に１０００試合のゲームを行った結果、図６１に示すように、プレーヤ１が大きく勝ち越す結果（対戦勝率）が得られた。但し、ランダムに行動するプレーヤを相手にした場合（ランダム相手）、プレーヤ１もプレーヤ２も相手に対して９割以上の高い勝率を得た。つまり、思考ルーチンを利用して行動するプレーヤは、十分に賢く行動しているのである。このように、複数エージェントの同時学習を適用すると、相手に勝とうとして思考ルーチンを強化しているうちに、ランダムに行動する相手に対しても高い確率で勝てる汎用的なアルゴリズムが得られる。

【０１９１】

ところで、これまでは状態Ｓとして、自分の座標、相手の座標、時刻を表現した３次元マトリクスを用いていたが、この３次元マトリクスに代えてゲーム画面の画像情報をそのまま用いる方法も考えられる。例えば、状態Ｓとして、図６２に示すようなゲーム画面の輝度画像を用いることができる。つまり、状態Ｓは、行動を決定するために有用な情報が含まれてさえいれば何でもよいのである。この考えに基づくと、本実施形態に係る技術が様々なゲームやタスクに関する思考ルーチンの自動構築方法に応用できることが容易に想像できるであろう。

【０１９２】

以上、「対戦ゲーム」への応用について説明した。

【０１９３】

［３－３：「五目並べ」への応用］

次に、図６３及び図６４を参照しながら、本実施形態に係る技術を「五目並べ」へと応用する方法について説明する。「五目並べ」の主なルールは、（１）交互に手を打つ、（２）縦横斜めに先に５つの石を並べた方が勝ち、の２点である。また、「五目並べ」において、状態Ｓは盤面であり、行動ａは各プレーヤが打つ手である。

【０１９４】

「五目並べ」への応用方法は、基本的に「三目並べ」への応用方法と同じである。つまり、状態Ｓ及び行動ａは、図６３に示すように、２次元マトリクスで表現される。また、最初に用いる行動履歴データは、完全にランダムに石を配置するＩｎｎａｔｅを用いて取得される。そして、最終的に勝ちに至った全ての（Ｓ，ａ）に報酬１が設定され、負けに

至った全ての（Ｓ，ａ）に報酬０が設定される。情報処理装置１０は、この行動履歴データを用いて思考ルーチンを構築する。また、情報処理装置１０は、思考ルーチンを用いて対局し、その結果を統合した行動履歴データを用いて思考ルーチンを構築する。これらの処理を繰り返すことにより、賢い行動を選択する思考ルーチンが構築される。

【０１９５】

また、行動を選択する際、情報処理装置１０は、「三目並べ」の場合と同様に全ての行動の可能性について（石を置ける全ての点について石を置いたとして）推定報酬を求め、最も推定報酬の高くなる点に石を置く。もちろん、情報処理装置１０は、数手先を読んで石を置く位置を選択するように構成されていてもよい。なお、「五目並べ」は、「三目並べ」に比べて盤面の組み合わせ数が膨大である。そのため、ランダムに石を置くプレーヤは、見当違いの手を打ちがちであるために非常に弱い。

【０１９６】

従って、ランダムに石を置くプレーヤを相手に学習を行っても非常に弱い相手に勝つための思考ルーチンができあがるだけで、賢い思考ルーチンはなかなか得られない。そこで、対戦ゲームと同様に、複数エージェントの同時学習を適用し、自分と相手とを同じ環境で学習させる手法を用いる方が好ましい。このような構成にすることで、比較的高性能な思考ルーチンを自動構築することが可能になる。互いに思考ルーチンを用いて行動するプレーヤによる対局結果を図６４に示した。

【０１９７】

以上、「五目並べ」への応用について説明した。

【０１９８】

［３－４：「ポーカー」への応用］

次に、図６５～図６７を参照しながら、本実施形態に係る技術を「ポーカー」へと応用する方法について説明する。「ポーカー」の主なルールは、図６５に示すように、（１）５枚のカードを配る、（２）捨てるカードを選択する、（３）役の強い方が勝ち、の３点である。ここでは、カードが配られたときに、捨てるカードを決める思考ルーチンの構築方法について考える。

【０１９９】

図６６に示すように、状態Ｓ及び行動ａは、記号列で表現される。例えば、ハートのエースを“ＨＡ”、クラブの２を“Ｃ２”、ダイヤのＫを“ＤＫ”などと表現する。図６６の場合、状態Ｓは、記号列“ＳＪＣＪＣ０Ｄ９Ｄ７”で表現される。また、ダイヤの９及びダイヤの７を捨てた場合、行動ａは、記号列“Ｄ９Ｄ７”で表現される。また、ゲームに勝った場合には報酬“１”が与えられ、負けた場合には報酬“０”が与えられる。このような表現を用いると、例えば、図６７に示すような行動履歴データが得られる。

【０２００】

最初に行動履歴データを取得するＩｎｎａｔｅとしては、例えば、完全にランダムに５枚のカードそれぞれを捨てるかどうか決定するものを利用する。また、情報処理装置１０は、勝った行動履歴データには報酬“１”を設定し、負けた行動履歴データには報酬“０”を設定する。そして、情報処理装置１０は、蓄積された行動履歴データを用いて思考ルーチンを構築する。このとき、行動を選択した結果、どのような役が揃ったか、相手の役がどのようなものだったかなどの情報は利用されない。つまり、純粋に勝ち負けだけを考慮して思考ルーチンが構築される。但し、自分が強い役を揃えるのに有利なカードの切り方を選択した行動履歴データほど報酬が１になる確率は高くなる傾向にある。

【０２０１】

さて、行動を選択する際、配られた５枚のカードそれぞれについて、カードを切る、カードを切らない、の選択肢が与えられる。そのため、行動の組み合わせは、２の５乗＝３２通り存在する。従って、思考ルーチンは、報酬推定機を利用し、３２通りの（Ｓ，ａ）について推定報酬を算出し、最も推定報酬の高い行動を選択する。

【０２０２】

以上、「ポーカー」への応用について説明した。

【０２０３】
［３－５：「ロールプレイングゲーム」への応用］
次に、図６８～図７９を参照しながら、本実施形態に係る技術を「ロールプレイングゲーム」へと応用する方法について説明する。ここでは、「ロールプレイングゲーム」の戦闘シーンにおいてプレーヤに代わってキャラクタを賢く自動操作する思考ルーチンの自動構築方法について考える。なお、ここで考える「ロールプレイングゲーム」のルールは、図６８に示した通りである。また、図６８に示すように、状態Ｓはプレーヤに提供される情報であり、行動ａはキャラクタを操作するコマンドである。
【０２０４】
戦闘シーンの環境は、図６９に示した通りである。まず、戦闘に勝つと生存者で経験値が山分けされる。さらに、経験値が貯まるとレベルアップする。また、レベルアップすると、キャラクタの職業に応じてステータスの値がアップしたり、魔法を覚えたりする。また、戦闘に５回連続で勝つと敵のレベルが１つアップすると共に、キャラクタのヒットポイントが回復する。また、敵のレベルが３１に達するとゲームをクリアしたことになる。
【０２０５】
なお、戦闘シーンにおいて、キャラクタが持つステータスの１つである“素早さ”の値に応じて各キャラクタが行動をおこせるタイミングが決まる。また、キャラクタがとれる行動は、“攻撃”及び“魔法（魔法を覚えている場合）”である。魔法の種類としては、Ｈｅａｌ、Ｆｉｒｅ、Ｉｃｅがある。Ｈｅａｌは、味方のヒットポイント（ＨＰ）を回復する魔法である。Ｆｉｒｅは、火を用いて敵を攻撃する魔法である。Ｉｃｅは、氷を用いて敵を攻撃する魔法である。また、魔法をかけるターゲットは、単体又は全体のいずれかを選択可能である。但し、全体を選択した場合には魔法の効果が半減する。また、使える魔法の種類やレベルは、キャラクタのレベルに応じて変わる。さらに、同じ魔法でもレベルの高い魔法ほどマジックポイント（ＭＰ）を多く消費するが、効果は高い。
【０２０６】
キャラクタの職業及び職業毎のステータスは、図７０に示した通りである。ステータス上昇率は、キャラクタのレベルが１つアップする度にステータスがアップする割合を示している。また、魔法を覚えるＬｖは、記載された値のレベルに達した場合にキャラクタが魔法を覚えるレベルを示している。但し、空欄に対応する魔法は覚えられない。また、０と記載されている箇所は、最初から魔法を覚えていることを示している。なお、味方のパーティは、上側４種類の職業を持つキャラクタにより構成される。一方、敵のパーティは、下側４種類の職業を持つキャラクタから選択されたキャラクタにより構成される。
【０２０７】
状態Ｓとして利用される味方側の情報は、図７１に示した通りである。例えば、生存する味方のレベル、職業、ＨＰ、最大ＨＰ、ＭＰ、最大ＭＰ、攻撃力、防御力、素早さなどが状態Ｓとして利用される。なお、職業の欄は、当てはまる職業の欄に１、それ以外の欄に０が記入される。また、その他の欄には現状の値が記入される。一方、状態Ｓとして利用される敵側の情報は、図７２に示した通りである。例えば、生存する敵のレベル、職業、累積ダメージなどが状態Ｓとして利用される。なお、累積ダメージは、それまでに与えたダメージの合計値を示している。
【０２０８】
また、行動ａとして利用される味方側の情報は、図７３に示した通りである。例えば、行動者の欄には、これから行動を行うキャラクタの場合に１、それ以外の場合に０が記入される。また、行動対象の欄には、行動の対象となるキャラクタの場合に１、それ以外の場合に０が記入される。例えば、回復魔法を受けるキャラクタに対応する行動対象の欄には１が記入される。また、アクションの種類の欄には、行う行動の欄に１、行わない行動の欄に０が記入される。一方、行動ａとして利用される敵側の情報は、図７４に示した通りである。図７４に示すように、敵側の情報としては行動対象の情報が利用される。
【０２０９】
さて、これまで説明してきた応用例と同様、情報処理装置１０は、まず、行動履歴デー

タを取得する。このとき、情報処理装置１０は、行動の種類毎に選択確率に重みを付けた上で、ランダムに行動を選択するＩｎｎａｔｅを用いて行動履歴データを取得する。例えば、情報処理装置１０は、魔法よりも攻撃を選択する確率を高く設定したＩｎｎａｔｅを用いてキャラクタを行動させる。また、図７５に示すように、情報処理装置１０は、味方がやられた場合には報酬“－５”を行動履歴データに設定し、敵を倒した場合には報酬“１”を行動履歴データに設定する。その結果、図７６のＡ図に示すような行動履歴データが得られる。但し、味方や敵がやられる過程の評価も考慮するため、情報処理装置１０は、図７６のＢ図に示すように、直線的に報酬の値をＤｅｃａｙさせる。

【０２１０】

情報処理装置１０は、上記のようにして取得された行動履歴データを用いて思考ルーチンを構築する。このとき、情報処理装置１０は、時刻ｔにおける状態Ｓ及び行動ａから時刻ｔ＋１における状態Ｓ’を推定する予測機を構築する。また、情報処理装置１０は、時刻ｔ＋１における状態Ｓ’から推定報酬を算出する報酬推定機を構築する。そして、情報処理装置１０は、図７７に示すように、現在の状態Ｓにおいてキャラクタがとりうる行動毎に、予測機を用いて次の状態Ｓ’を予測する。さらに、情報処理装置１０は、予測した状態Ｓ’を報酬推定機に入力して推定報酬ｙを算出する。推定報酬ｙを算出した情報処理装置１０は、推定報酬ｙが最大となる行動ａを選択する。

【０２１１】

図７７の例では、行動ａ＝“敵全体にＦｉｒｅ”に対応する推定報酬ｙが最大となっている。そのため、この例においては、最適な行動として、行動ａ＝“敵全体にＦｉｒｅ”が選択される。但し、思考ルーチンは、図７８に示すように、推定報酬が高く、推定誤差が大きく、特徴量空間における密度が疎な特徴量座標に対応する行動を選択するように構成されていてもよい。つまり、先に説明した効率的な推定報酬機の構築方法で紹介した報酬スコア、未知スコア、誤差スコアに基づいて思考ルーチンが構築されていてもよい。

【０２１２】

なお、報酬スコアは、とりうる全ての行動について報酬推定機を用いて推定報酬を求め、推定報酬の低い方から順に１、２、３、…と、推定報酬が高くなるほど大きくなるようにスコアを与えることで得られる。また、未知スコアは、図２５などに示した方法を用いて、全ての行動について特徴量座標の周辺密度を求め、密度が高い方から順に１、２、３、…と、密度が低くなるほど大きくなるようにスコアを与えることで得られる。

【０２１３】

また、誤差スコアを求める場合、情報処理装置１０は、まず、既存の行動履歴データの全てについて、推定報酬ｙの値を実際の報酬ｒと比較し、その誤差を求める。次いで、情報処理装置１０は、平均値よりも誤差の大きい行動履歴データに対応する特徴量座標を特徴量空間にプロットする。次いで、情報処理装置１０は、プロットした特徴量座標の密度分布を求める。最後に、情報処理装置１０は、全ての行動履歴データに対応する特徴量座標について密度を求め、密度が低い方から順に１、２、３、…と、密度が高くなるほど大きくなるようにスコアを与える。

【０２１４】

例えば、報酬スコアをs_1、未知スコアをs_2、誤差スコアをs_3と表記した場合、情報処理装置１０は、行動を選択する際に、$s_1 * w_1 + s_2 * w_2 + s_3 * w_3$（但し、$w_1$～$w_3$は所定の重み）の値を算出し、この値が最も大きくなる行動を選択する。このようにして行動を選択することにより、報酬が高く、推定誤差が大きく、特徴量空間における特徴量座標の密度が疎な行動を選択することが可能になる。

【０２１５】

ここで、図７９を参照しながら、効率的な推定報酬機の構築方法を適用した場合の効果について述べる。図７９のグラフは、最も高い推定報酬が得られる行動を選択した場合（最適戦略）と、効率的な推定報酬機の構築方法を適用した場合（探索行動）とでシナリオクリアまでの１ステップ当たりの平均Ｒｅｗａｒｄを比較したグラフである。図７９のグラフから明らかなように、３つのスコアを利用して構築された思考ルーチン（探索行動）

の方が安定して高い報酬が得られている。この評価結果から、効率的な推定報酬機の構築方法を適用することで、演算負荷を軽減できるばかりか、より高性能な思考ルーチンを構築できることが分かった。

【０２１６】

なお、「ロールプレイングゲーム」に応用すべく自動構築された思考ルーチンは、以下のような戦略を自身で身に付けていることも分かった。

（Ａ）集中攻撃：

集中攻撃することで敵の数を素早く減らす。

（Ｂ）ＨＰ減少したら回復：

ＨＰが減少した味方のＨＰを回復して味方が倒されにくくする。

（Ｃ）単体攻撃、全体攻撃の使い分け：

敵の数がある程度多い時は全体攻撃魔法を使う。現在集中攻撃中の敵が残りわずかなダメージで倒せそうなときは、全体攻撃魔法を用いて集中攻撃中の敵を倒しながら、他の敵にもダメージを与える。

（Ｄ）魔法の無駄打ちはしない：

ＨＰの減っていない味方に対して回復魔法を使わない。魔法の効かない敵に対して魔法を使わない。

【０２１７】

以上、「ロールプレイングゲーム」への応用について説明した。

【０２１８】

以上説明したように、本実施形態に係る技術を適用すると、人手による調整を介さずに様々な思考ルーチンを自動構築することが可能になる。

【０２１９】

＜４：ハードウェア構成例＞

上記の情報処理装置１０が有する各構成要素の機能は、例えば、図８０に示すハードウェア構成を用いて実現することが可能である。つまり、当該各構成要素の機能は、コンピュータプログラムを用いて図８０に示すハードウェアを制御することにより実現される。なお、このハードウェアの形態は任意であり、例えば、パーソナルコンピュータ、携帯電話、ＰＨＳ、ＰＤＡ等の携帯情報端末、ゲーム機、又は種々の情報家電がこれに含まれる。但し、上記のＰＨＳは、Ｐｅｒｓｏｎａｌ Ｈａｎｄｙ－ｐｈｏｎｅ Ｓｙｓｔｅｍの略である。また、上記のＰＤＡは、Ｐｅｒｓｏｎａｌ Ｄｉｇｉｔａｌ Ａｓｓｉｓｔａｎｔの略である。

【０２２０】

図８０に示すように、このハードウェアは、主に、ＣＰＵ９０２と、ＲＯＭ９０４と、ＲＡＭ９０６と、ホストバス９０８と、ブリッジ９１０と、を有する。さらに、このハードウェアは、外部バス９１２と、インターフェース９１４と、入力部９１６と、出力部９１８と、記憶部９２０と、ドライブ９２２と、接続ポート９２４と、通信部９２６と、を有する。但し、上記のＣＰＵは、Ｃｅｎｔｒａｌ Ｐｒｏｃｅｓｓｉｎｇ Ｕｎｉｔの略である。また、上記のＲＯＭは、Ｒｅａｄ Ｏｎｌｙ Ｍｅｍｏｒｙの略である。そして、上記のＲＡＭは、Ｒａｎｄｏｍ Ａｃｃｅｓｓ Ｍｅｍｏｒｙの略である。

【０２２１】

ＣＰＵ９０２は、例えば、演算処理装置又は制御装置として機能し、ＲＯＭ９０４、ＲＡＭ９０６、記憶部９２０、又はリムーバブル記録媒体９２８に記録された各種プログラムに基づいて各構成要素の動作全般又はその一部を制御する。ＲＯＭ９０４は、ＣＰＵ９０２に読み込まれるプログラムや演算に用いるデータ等を格納する手段である。ＲＡＭ９０６には、例えば、ＣＰＵ９０２に読み込まれるプログラムや、そのプログラムを実行する際に適宜変化する各種パラメータ等が一時的又は永続的に格納される。

【０２２２】

これらの構成要素は、例えば、高速なデータ伝送が可能なホストバス９０８を介して相互に接続される。一方、ホストバス９０８は、例えば、ブリッジ９１０を介して比較的デ

ータ伝送速度が低速な外部バス９１２に接続される。また、入力部９１６としては、例えば、マウス、キーボード、タッチパネル、ボタン、スイッチ、及びレバー等が用いられる。さらに、入力部９１６としては、赤外線やその他の電波を利用して制御信号を送信することが可能なリモートコントローラ（以下、リモコン）が用いられることもある。

【０２２３】

出力部９１８としては、例えば、ＣＲＴ、ＬＣＤ、ＰＤＰ、又はＥＬＤ等のディスプレイ装置、スピーカ、ヘッドホン等のオーディオ出力装置、プリンタ、携帯電話、又はファクシミリ等、取得した情報を利用者に対して視覚的又は聴覚的に通知することが可能な装置である。但し、上記のＣＲＴは、Ｃａｔｈｏｄｅ　Ｒａｙ　Ｔｕｂｅの略である。また、上記のＬＣＤは、Ｌｉｑｕｉｄ　Ｃｒｙｓｔａｌ　Ｄｉｓｐｌａｙの略である。そして、上記のＰＤＰは、Ｐｌａｓｍａ　ＤｉｓｐｌａｙＰａｎｅｌの略である。さらに、上記のＥＬＤは、Ｅｌｅｃｔｒｏ－Ｌｕｍｉｎｅｓｃｅｎｃｅ　Ｄｉｓｐｌａｙの略である。

【０２２４】

記憶部９２０は、各種のデータを格納するための装置である。記憶部９２０としては、例えば、ハードディスクドライブ（ＨＤＤ）等の磁気記憶デバイス、半導体記憶デバイス、光記憶デバイス、又は光磁気記憶デバイス等が用いられる。但し、上記のＨＤＤは、Ｈａｒｄ　Ｄｉｓｋ　Ｄｒｉｖｅの略である。

【０２２５】

ドライブ９２２は、例えば、磁気ディスク、光ディスク、光磁気ディスク、又は半導体メモリ等のリムーバブル記録媒体９２８に記録された情報を読み出し、又はリムーバブル記録媒体９２８に情報を書き込む装置である。リムーバブル記録媒体９２８は、例えば、ＤＶＤメディア、Ｂｌｕ－ｒａｙメディア、ＨＤ　ＤＶＤメディア、各種の半導体記憶メディア等である。もちろん、リムーバブル記録媒体９２８は、例えば、非接触型ＩＣチップを搭載したＩＣカード、又は電子機器等であってもよい。但し、上記のＩＣは、Ｉｎｔｅｇｒａｔｅｄ　Ｃｉｒｃｕｉｔの略である。

【０２２６】

接続ポート９２４は、例えば、ＵＳＢポート、ＩＥＥＥ１３９４ポート、ＳＣＳＩ、ＲＳ－２３２Ｃポート、又は光オーディオ端子等のような外部接続機器９３０を接続するためのポートである。外部接続機器９３０は、例えば、プリンタ、携帯音楽プレーヤ、デジタルカメラ、デジタルビデオカメラ、又はＩＣレコーダ等である。但し、上記のＵＳＢは、Ｕｎｉｖｅｒｓａｌ　Ｓｅｒｉａｌ　Ｂｕｓの略である。また、上記のＳＣＳＩは、Ｓｍａｌｌ　Ｃｏｍｐｕｔｅｒ　Ｓｙｓｔｅｍ　Ｉｎｔｅｒｆａｃｅの略である。

【０２２７】

通信部９２６は、ネットワーク９３２に接続するための通信デバイスであり、例えば、有線又は無線ＬＡＮ、Ｂｌｕｅｔｏｏｔｈ（登録商標）、又はＷＵＳＢ用の通信カード、光通信用のルータ、ＡＤＳＬ用のルータ、又は各種通信用のモデム等である。また、通信部９２６に接続されるネットワーク９３２は、有線又は無線により接続されたネットワークにより構成され、例えば、インターネット、家庭内ＬＡＮ、赤外線通信、可視光通信、放送、又は衛星通信等である。但し、上記のＬＡＮは、Ｌｏｃａｌ　Ａｒｅａ　Ｎｅｔｗｏｒｋの略である。また、上記のＷＵＳＢは、Ｗｉｒｅｌｅｓｓ　ＵＳＢの略である。そして、上記のＡＤＳＬは、Ａｓｙｍｍｅｔｒｉｃ　Ｄｉｇｉｔａｌ　Ｓｕｂｓｃｒｉｂｅｒ　Ｌｉｎｅの略である。

【０２２８】

以上、ハードウェア構成例について説明した。

【０２２９】

＜５：まとめ＞

最後に、本実施形態の技術的思想について簡単に纏める。以下に記載する技術的思想は、例えば、ＰＣ、携帯電話、携帯ゲーム機、携帯情報端末、情報家電、カーナビゲーションシステム等、種々の情報処理装置に対して適用することができる。

【０２３０】

上記の情報処理装置の機能構成は以下のように表現することができる。例えば、下記（１）に記載の情報処理装置は、行動履歴データを用いて報酬推定機を自動構築することができる。この報酬推定機を利用すると、エージェントがおかれた状態に応じて、その状態でエージェントがとりうる行動毎に、行動を行ったエージェントが得る報酬を推定することができる。そのため、高い報酬を得ると推定される行動をエージェントがとるように制御することで、賢く行動するエージェントの動きを実現することが可能になる。言い換えると、下記（１）に記載の情報処理装置は、賢く行動するエージェントの動きを実現することが可能な思考ルーチンを自動構築することができる。

【０２３１】

（１）

エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データを学習データとして用い、入力された状態データ及び行動データから報酬値を推定する報酬推定機を機械学習により生成する報酬推定機生成部を備え、

前記報酬推定機生成部は、

複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成部と、

前記行動履歴データに含まれる状態データ及び行動データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出部と、

前記特徴量ベクトルから前記行動履歴データに含まれる報酬値を推定する推定関数を回帰／判別学習により算出する推定関数算出部と、

を含み、

前記報酬推定機は、前記複数の基底関数と前記推定関数とにより構成される、

情報処理装置。

【０２３２】

（２）

エージェントがおかれた現在の状態を表す状態データと、当該エージェントが次にとりうる行動を表す行動データとを前記報酬推定機に入力し、当該行動をとった結果として当該エージェントが得る報酬値を推定する報酬値推定部と、

前記現在の状態において前記エージェントが次にとりうる行動のうち、前記報酬値推定部により推定された報酬値が最も高い値となる行動を選択する行動選択部と、

をさらに備える、

上記（１）に記載の情報処理装置。

【０２３３】

（３）

前記行動選択部による選択結果に基づいてエージェントを行動させる行動制御部と、

前記エージェントの行動に伴って更新される状態データ及び行動データを蓄積し、当該行動の結果としてエージェントが得た報酬を表す報酬値と、蓄積した状態データ及び行動データとを対応付けて前記行動履歴データに追加する履歴データ追加部と、

をさらに備える、

上記（２）に記載の情報処理装置。

【０２３４】

（４）

前記状態データ、前記行動データ、及び前記報酬値の組が前記行動履歴データに追加された場合、前記特徴量ベクトル算出部は、前記行動履歴データに含まれる全ての状態データ及び行動データについて特徴量ベクトルを算出し、

前記情報処理装置は、特徴領空間において前記特徴量ベクトルにより示される座標点の分布が所定の分布に近づくように前記行動履歴データに含まれる前記状態データ、前記行動データ、及び前記報酬値の組を間引く分布調整部をさらに備える、

上記（３）に記載の情報処理装置。

【０２３５】

（５）
前記状態データ、前記行動データ、及び前記報酬値の組が前記行動履歴データに追加された場合、前記特徴量ベクトル算出部は、前記行動履歴データに含まれる全ての状態データ及び行動データについて特徴量ベクトルを算出し、
前記情報処理装置は、特徴領空間において前記特徴量ベクトルにより示される座標点の分布が所定の分布に近づくように前記行動履歴データに含まれる前記状態データ、前記行動データ、及び前記報酬値の組のそれぞれに重みを設定する分布調整部をさらに備える、
上記（３）に記載の情報処理装置。
【０２３６】
（６）
前記分布調整部は、間引き後に残った前記状態データ、前記行動データ、及び前記報酬値の組について、特徴領空間において前記特徴量ベクトルにより示される座標点の分布が所定の分布に近づくように前記行動履歴データに含まれる前記状態データ、前記行動データ、及び前記報酬値の組のそれぞれに重みを設定する、
上記（４）に記載の情報処理装置。
【０２３７】
（７）
前記行動履歴データを学習データとして用い、現在の時刻においてエージェントがおかれた状態を表す状態データ及び現在の時刻においてエージェントがとる行動を表す行動データから次の時刻におけるエージェントの状態を表す状態データを予測する予測機を機械学習により生成する予測機生成部をさらに備え、
前記報酬値推定部は、
現在の時刻における状態データ及び行動データを前記予測機に入力して次の時刻におけるエージェントの状態を表す状態データを予測し、
前記次の時刻におけるエージェントの状態を表す状態データと、当該状態においてエージェントがとりうる行動を表す行動データとを前記報酬推定機に入力して、当該行動をとった結果として当該エージェントが得る報酬値を推定する、
上記（２）～（６）のいずれか１項に記載の情報処理装置。
【０２３８】
（８）
前記行動履歴データを学習データとして用い、現在の時刻においてエージェントがおかれた状態を表す状態データ及び現在の時刻においてエージェントがとる行動を表す行動データから次の時刻におけるエージェントの状態を表す状態データを予測する予測機を機械学習により生成する予測機生成部をさらに備え、
前記報酬値推定部は、現在の時刻を時刻ｔ０とした場合に、
時刻ｔ０における状態データ及び行動データを前記予測機に入力して次の時刻ｔ１におけるエージェントの状態を表す状態データを予測する処理を実行し、
ｋ＝１～ｎ－１（ｎ≧２）について、時刻ｔｋにおける状態データ及び時刻ｔｋにおいてエージェントがとりうる行動を表す行動データを前記予測機に入力して時刻ｔｋ＋１におけるエージェントの状態を表す状態データを予測する処理を逐次実行し、
予測した時刻ｔｎにおけるエージェントの状態を表す状態データと、当該状態においてエージェントがとりうる行動を表す行動データとを前記報酬推定機に入力して、当該行動をとった結果として当該エージェントが得る報酬値を推定する、
上記（２）～（６）のいずれか１項に記載の情報処理装置。
【０２３９】
（９）
前記報酬推定機生成部は、複数のエージェントの状態を表す状態データと、当該状態において各エージェントがとった行動を表す行動データと、当該行動の結果として各エージェントが得た報酬を表す報酬値とを含む行動履歴データを学習データとして用い、入力された状態データ及び行動データから報酬値を推定する報酬推定機を機械学習により生成す

る、
上記（１）～（８）のいずれか１項に記載の情報処理装置。
【０２４０】
（１０）
前記基底関数生成部は、遺伝的アルゴリズムに基づいて前記基底関数を更新し、
前記特徴量ベクトル算出部は、前記基底関数が更新された場合に、更新後の前記基底関数に前記状態データ及び前記行動データを入力して特徴量ベクトルを算出し、
前記推定関数算出部は、前記更新後の基底関数を用いて算出された特徴量ベクトルの入力に応じて前記報酬値を推定する推定関数を算出する、
上記（１）～（９）のいずれか１項に記載の情報処理装置。
【０２４１】
（１１）
エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データに基づき、高い報酬を得たエージェントがとった行動及び低い報酬を得たエージェントがとらなかった行動に高いスコアを与え、高い報酬を得たエージェントがとらなかった行動及び低い報酬を得たエージェントがとった行動に低いスコアを与える条件で、各状態データに対応する行動毎のスコアを算出するスコア算出部と、
前記行動履歴データ及び前記行動毎のスコアを学習データとして用い、入力された状態データから行動毎のスコアを推定するスコア推定機を機械学習により生成するスコア推定機生成部と、
を備え、
前記スコア推定機生成部は、
複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成部と、
前記行動履歴データに含まれる状態データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出部と、
前記特徴量ベクトルから前記行動毎のスコアを推定する推定関数を回帰／判別学習により算出する推定関数算出部と、
を含み、
前記スコア推定機は、前記複数の基底関数と前記推定関数とにより構成される、
情報処理装置。
【０２４２】
（１２）
エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データを学習データとして用い、入力された状態データ及び行動データから報酬値を推定する報酬推定機を機械学習により生成するステップを含み、
前記生成するステップは、
複数の処理関数を組み合わせて複数の基底関数を生成するステップと、
前記行動履歴データに含まれる状態データ及び行動データを前記複数の基底関数に入力して特徴量ベクトルを算出するステップと、
前記特徴量ベクトルから前記行動履歴データに含まれる報酬値を推定する推定関数を回帰／判別学習により算出するステップと、
を含み、
前記報酬推定機は、前記複数の基底関数と前記推定関数とにより構成される、
情報処理方法。
【０２４３】
（１３）
エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含

む行動履歴データに基づき、高い報酬を得たエージェントがとった行動及び低い報酬を得たエージェントがとらなかった行動に高いスコアを与え、高い報酬を得たエージェントがとらなかった行動及び低い報酬を得たエージェントがとった行動に低いスコアを与える条件で、各状態データに対応する行動毎のスコアを算出するステップと、
前記行動履歴データ及び前記行動毎のスコアを学習データとして用い、入力された状態データから行動毎のスコアを推定するスコア推定機を機械学習により生成するステップと、
を含み、
前記生成するステップは、
複数の処理関数を組み合わせて複数の基底関数を生成するステップと、
前記行動履歴データに含まれる状態データを前記複数の基底関数に入力して特徴量ベクトルを算出するステップと、
前記特徴量ベクトルから前記行動毎のスコアを推定する推定関数を回帰／判別学習により算出するステップと、
を含み、
前記スコア推定機は、前記複数の基底関数と前記推定関数とにより構成される、
情報処理方法。
【０２４４】
（１４）
エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データを学習データとして用い、入力された状態データ及び行動データから報酬値を推定する報酬推定機を機械学習により生成する報酬推定機生成機能をコンピュータに実現させるためのプログラムであり、
前記報酬推定機生成機能は、
複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成機能と、
前記行動履歴データに含まれる状態データ及び行動データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出機能と、
前記特徴量ベクトルから前記行動履歴データに含まれる報酬値を推定する推定関数を回帰／判別学習により算出する推定関数算出機能と、
を含み、
前記報酬推定機は、前記複数の基底関数と前記推定関数とにより構成される、
プログラム。
【０２４５】
（１５）
エージェントの状態を表す状態データと、当該状態においてエージェントがとった行動を表す行動データと、当該行動の結果としてエージェントが得た報酬を表す報酬値とを含む行動履歴データに基づき、高い報酬を得たエージェントがとった行動及び低い報酬を得たエージェントがとらなかった行動に高いスコアを与え、高い報酬を得たエージェントがとらなかった行動及び低い報酬を得たエージェントがとった行動に低いスコアを与える条件で、各状態データに対応する行動毎のスコアを算出するスコア算出機能と、
前記行動履歴データ及び前記行動毎のスコアを学習データとして用い、入力された状態データから行動毎のスコアを推定するスコア推定機を機械学習により生成するスコア推定機生成機能と、
をコンピュータに実現させるためのプログラムであり、
前記スコア推定機生成機能は、
複数の処理関数を組み合わせて複数の基底関数を生成する基底関数生成機能と、
前記行動履歴データに含まれる状態データを前記複数の基底関数に入力して特徴量ベクトルを算出する特徴量ベクトル算出機能と、
前記特徴量ベクトルから前記行動毎のスコアを推定する推定関数を回帰／判別学習によ

り算出する推定関数算出機能と、
を含み、
前記スコア推定機は、前記複数の基底関数と前記推定関数とにより構成される、
プログラム。

【０２４６】

（備考）

上記の報酬推定機構築部１２は、報酬推定機生成部の一例である。上記の基底関数リスト生成部１２１は、基底関数生成部の一例である。上記の特徴量計算部１２２は、特徴量ベクトル算出部の一例である。上記の推定関数生成部１２３は、推定関数算出部の一例である。上記の行動選択部１４は、報酬値推定部、行動選択部、行動制御部の一例である。上記の行動履歴データ取得部１１は、履歴データ追加部の一例である。上記の行動履歴データ統合部１２４は、分布調整部の一例である。上記の報酬推定機構築部１２は、予測機生成部の一例である。上記の報酬推定機構築部１２は、スコア算出部、スコア推定機生成部の一例である。

【０２４７】

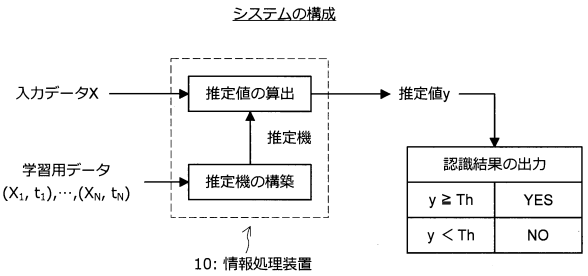
以上、添付図面を参照しながら本技術に係る好適な実施形態について説明したが、本技術はここで開示した構成例に限定されないことは言うまでもない。当業者であれば、特許請求の範囲に記載された範疇内において、各種の変更例又は修正例に想到し得ることは明らかであり、それらについても当然に本技術の技術的範囲に属するものと了解される。

【符号の説明】

【０２４８】

１０　情報処理装置
１１　行動履歴データ取得部
１２　報酬推定機構築部
１２１　基底関数リスト生成部
１２２　特徴量計算部
１２３　推定関数生成部
１２４　行動履歴データ統合部
１３　入力データ取得部
１４　行動選択部

【図１】

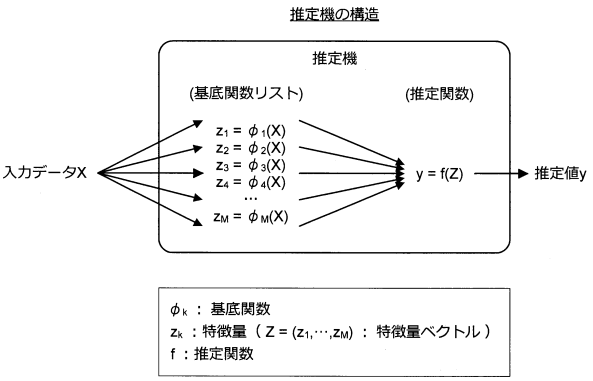


【図２】

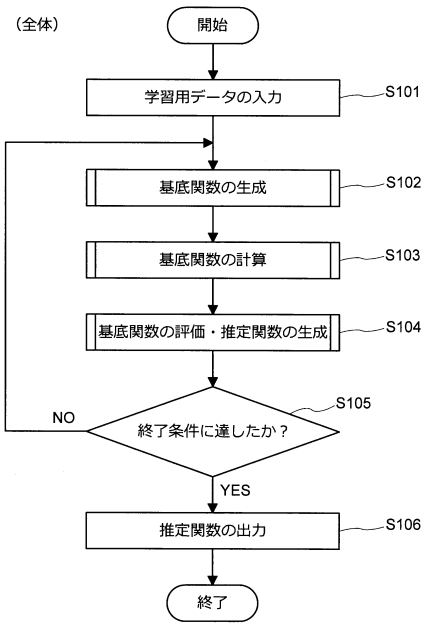
学習用データの例

データX_k	目的変数t_k	意味
画像#1	1	画像#1=海
画像#2	1	画像#2=海
⋮	⋮	⋮
画像#N	0	画像#N≠海

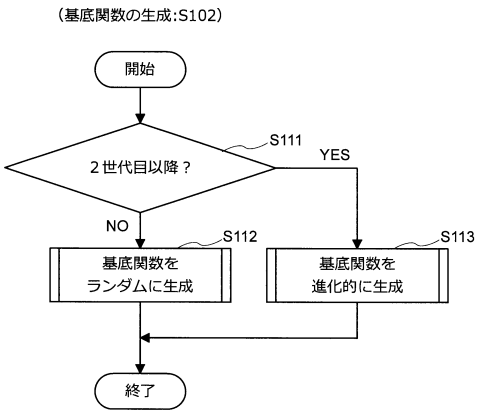
【図３】



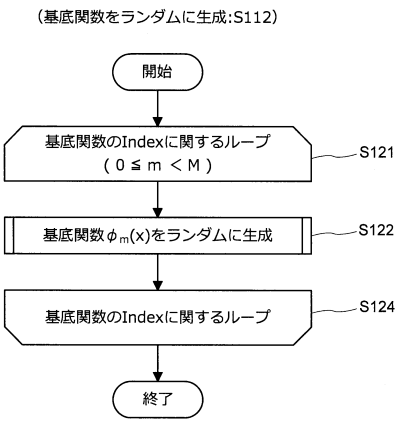
【図４】



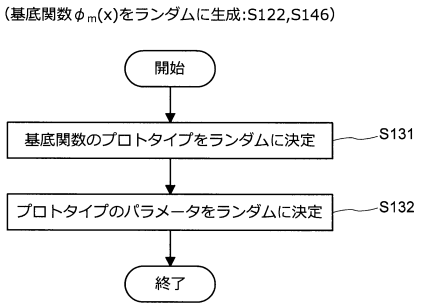
【図５】



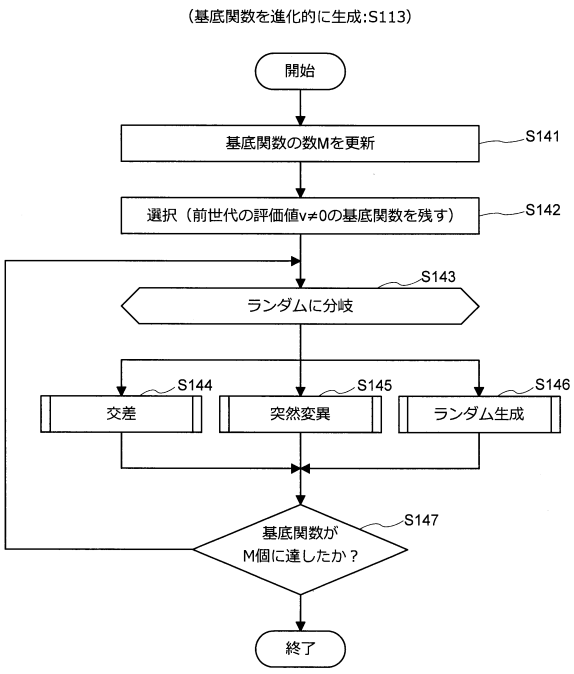
【図６】



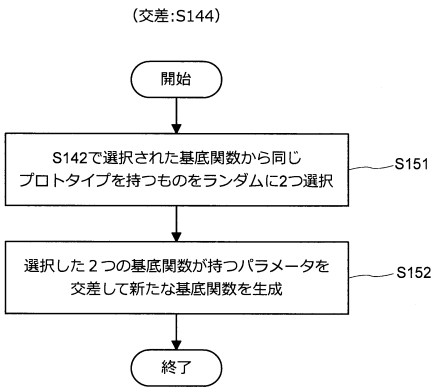
【図７】



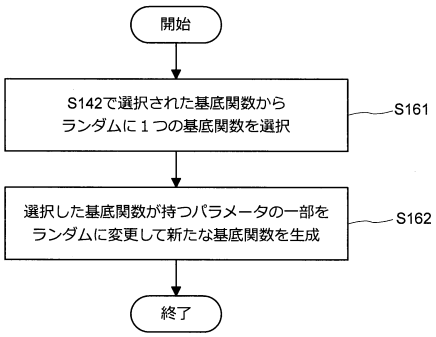
【図８】



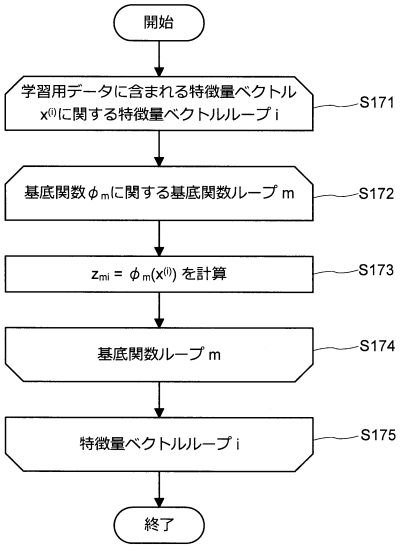
【図９】



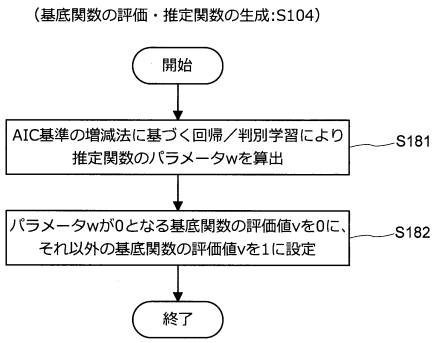
【図１０】



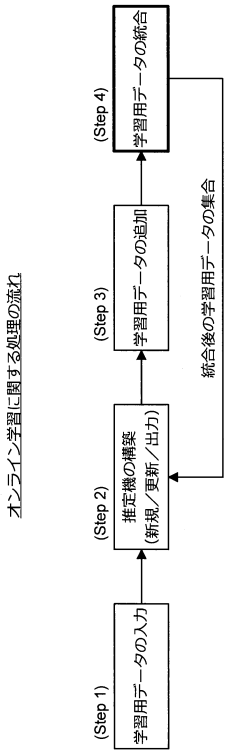
【図１１】



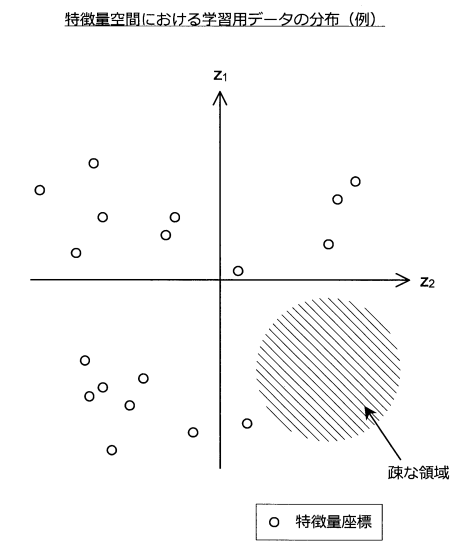
【図１２】



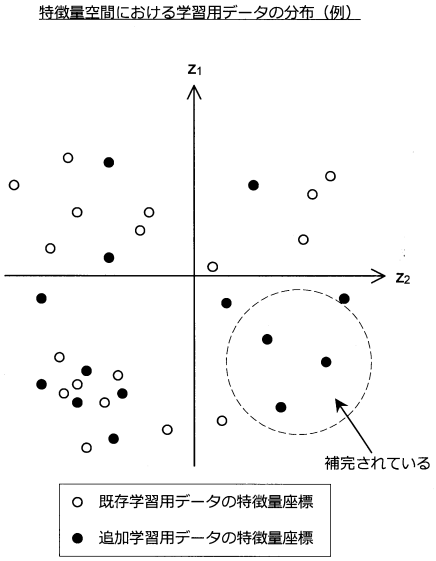
【図１３】



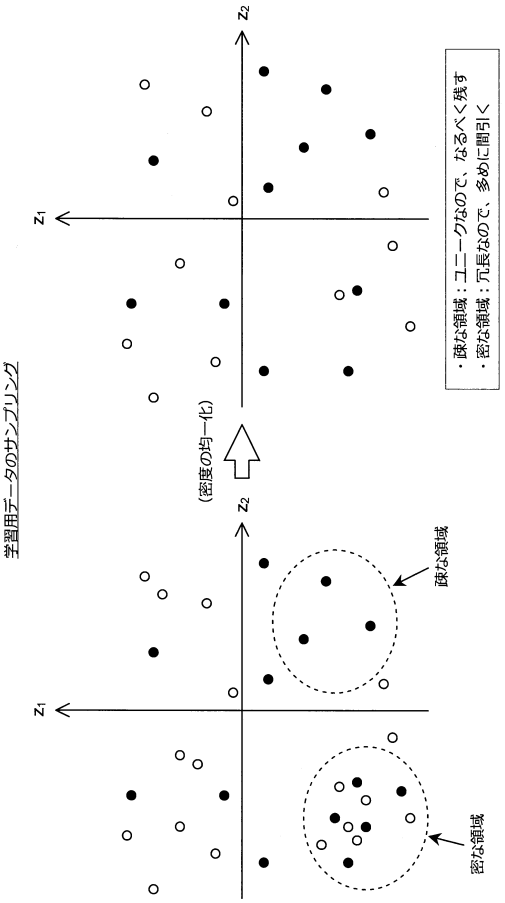
【図１４】



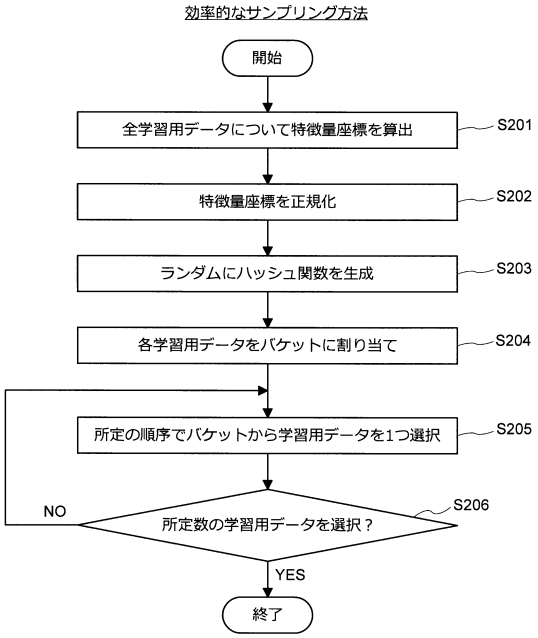
【図１５】



【図１６】



【図１７】



【図１８】

S202: 特徴量座標を正規化

正規化前

	z_1	z_2	z_3	…
学習用データ#1	12	0.01	-1028	…
学習用データ#2	15	0.03	-650	…
学習用データ#3	11	0.02	-20	…
学習用データ#4	13	0.05	-34	…
⋮	⋮	⋮	⋮	⋮

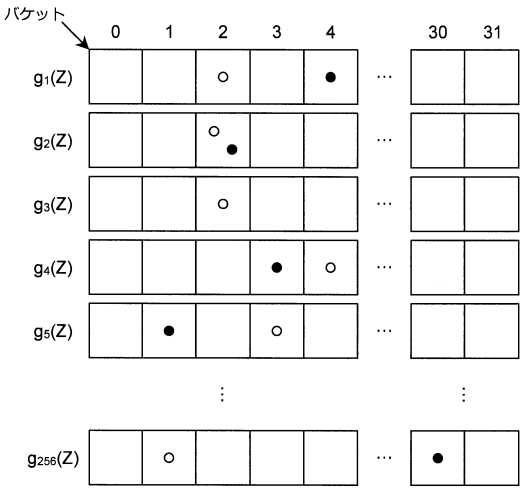
（分散1、平均0）

正規化後

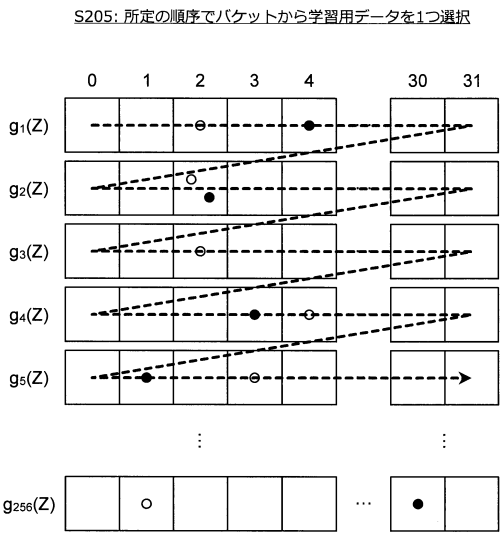
	z_1	z_2	z_3	…
学習用データ#1	-0.43916	-1.02470	-1.20546	…
学習用データ#2	1.31747	0.14639	-0.43964	…
学習用データ#3	-1.02470	-0.43916	0.83673	…
学習用データ#4	0.14639	1.31747	0.80837	…
⋮	⋮	⋮	⋮	⋮

【図１９】

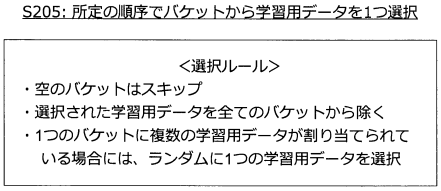
S204: 各学習用データをバケットに割り当て



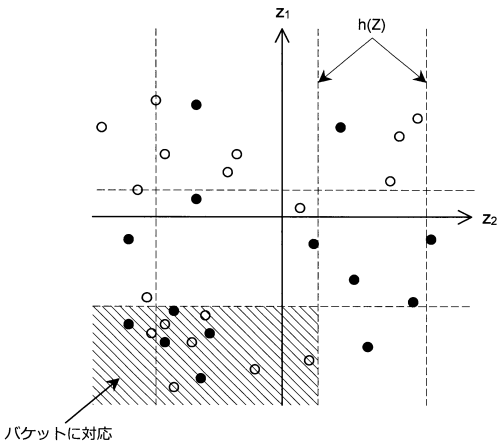
【図２０】



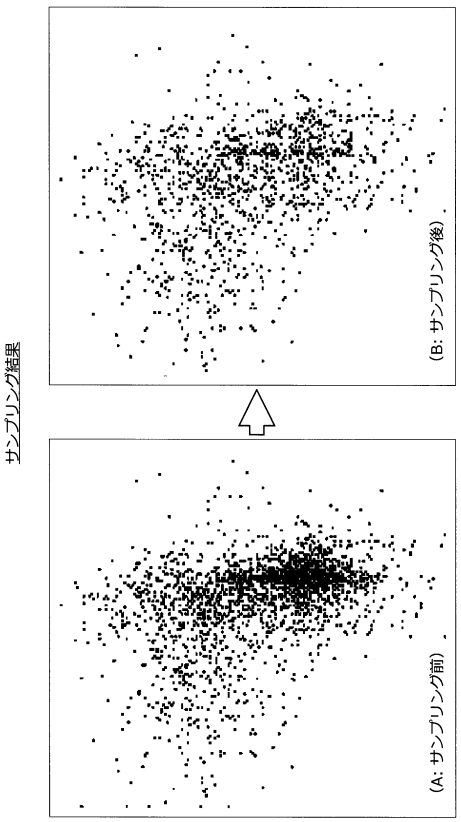
【図２１】



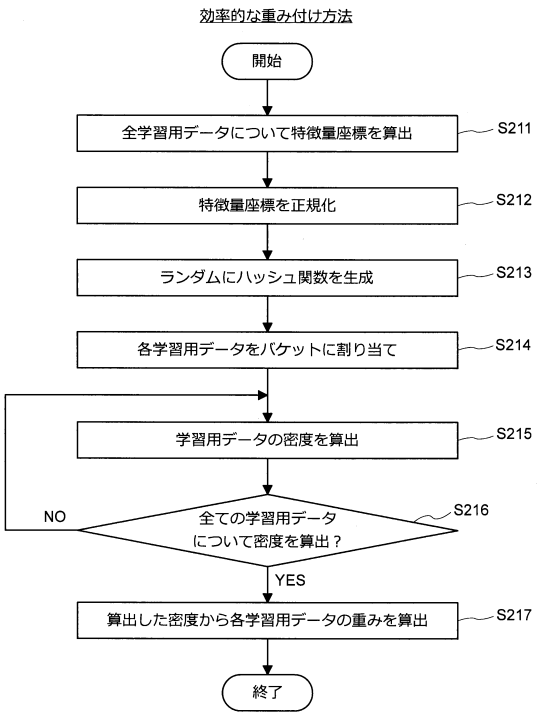
【図２２】



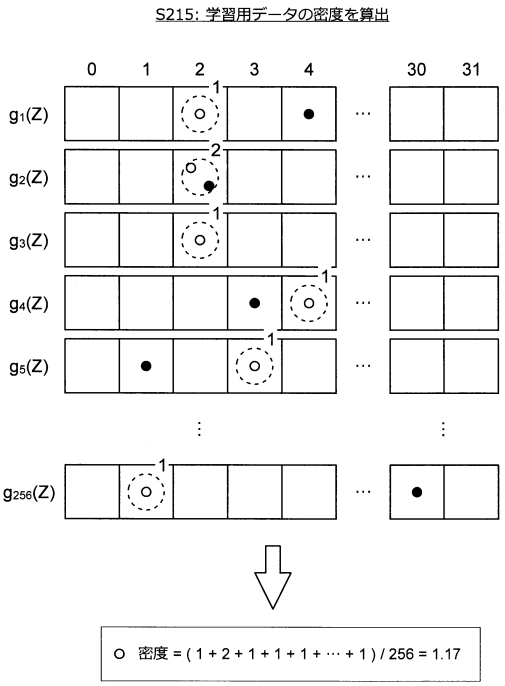
【図２３】



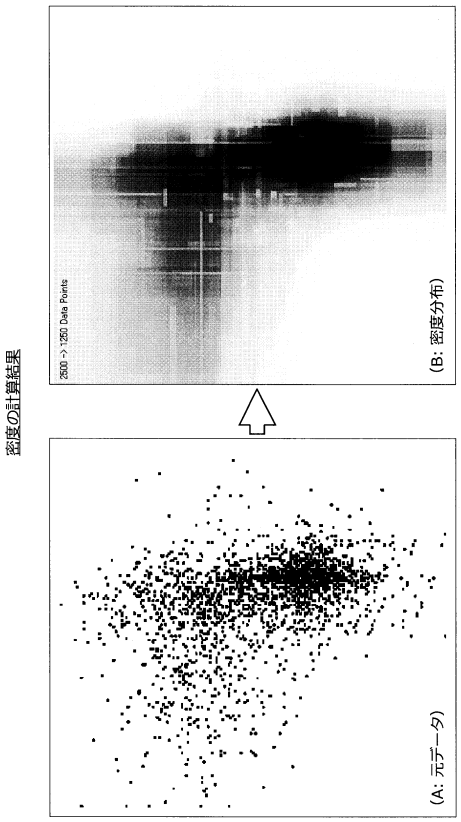
【図２４】



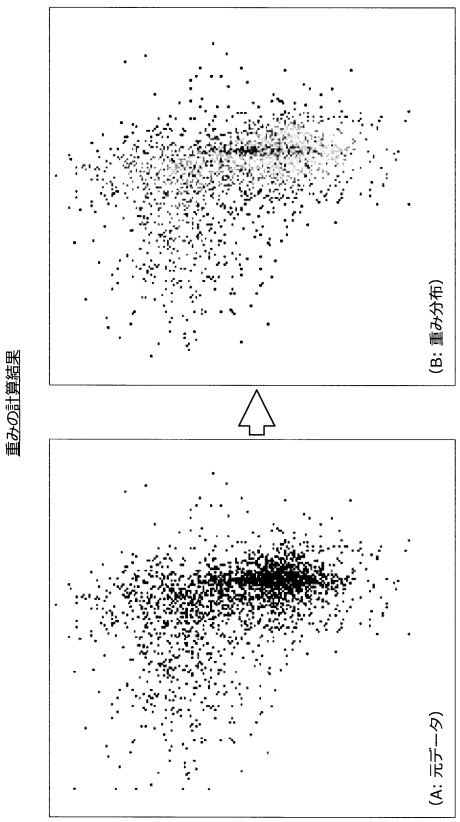
【図２５】



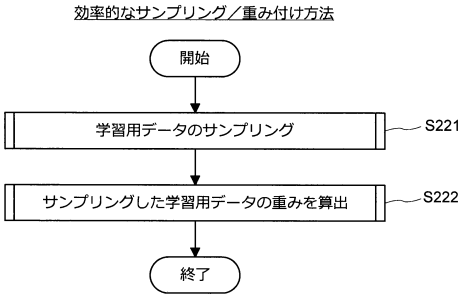
【図２６】



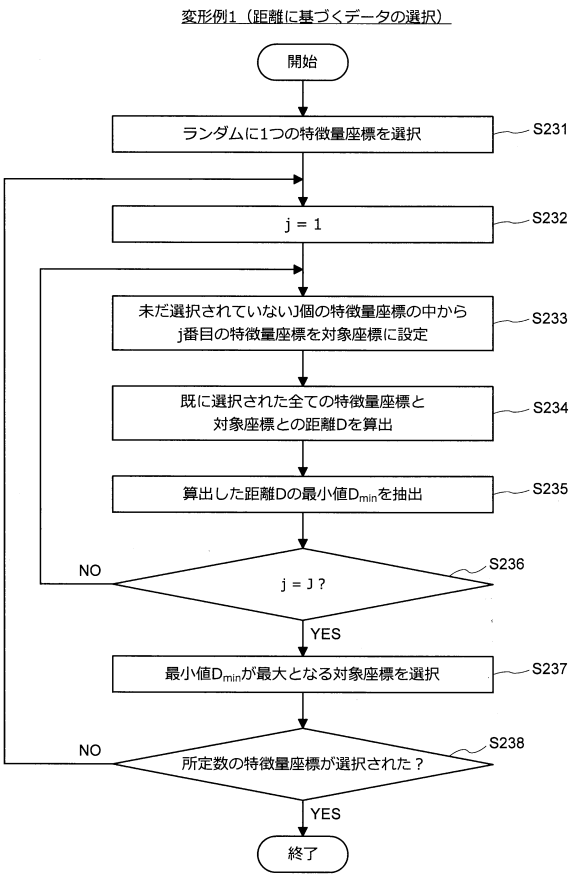
【図２７】



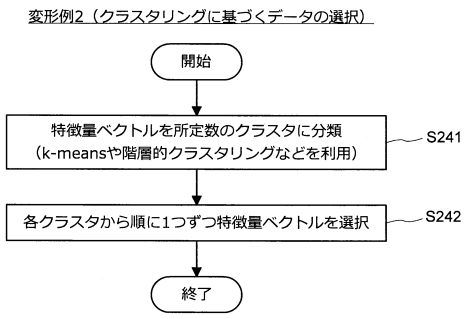
【図２８】



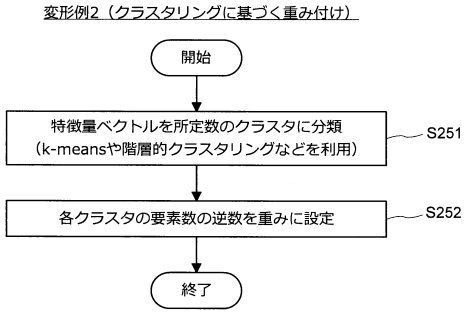
【図２９】



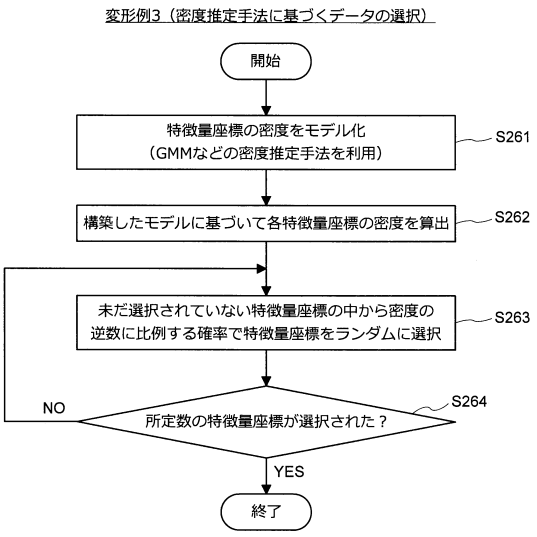
【図３０】



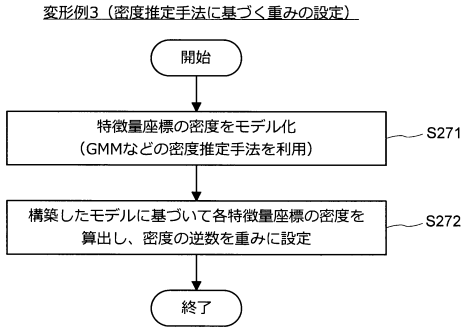
【図３１】



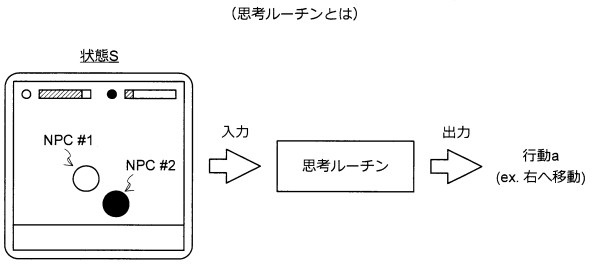
【図３２】



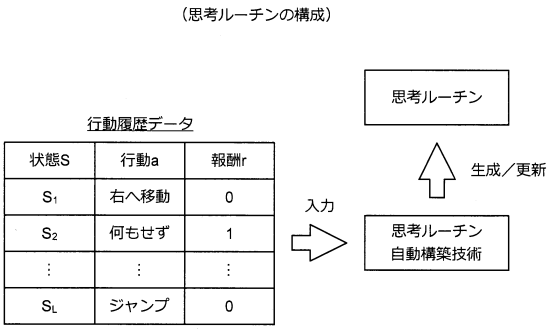
【図３３】



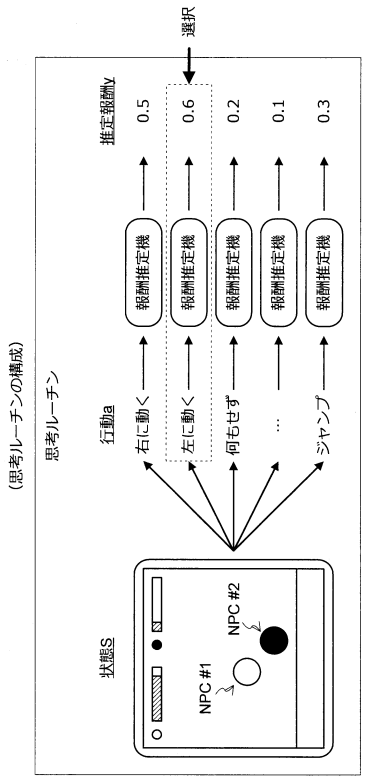
【図３４】



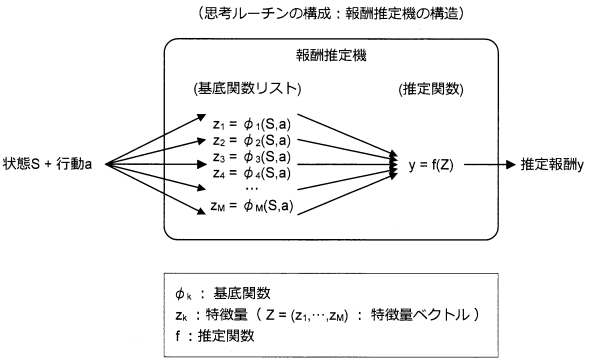
【図３５】



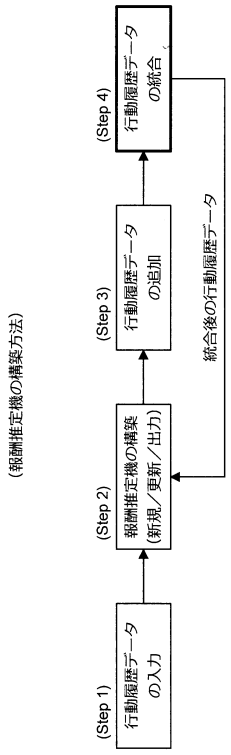
【図３６】



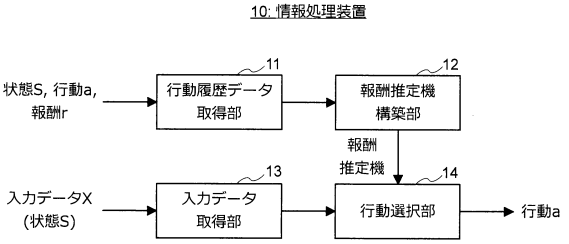
【図３７】



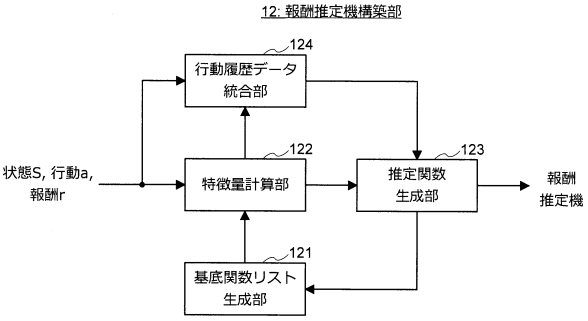
【図３８】



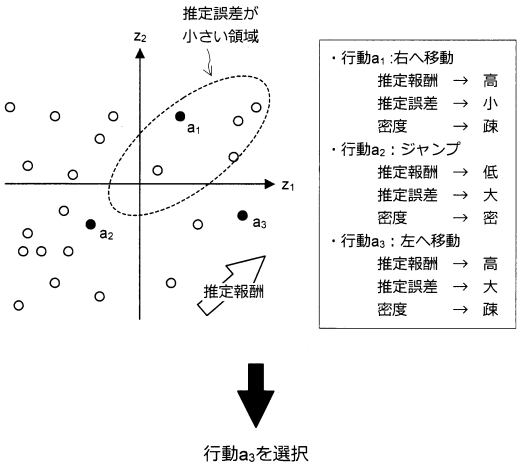
【図３９】



【図４０】



【図４１】



【図４２】

（効率的な報酬推定機の構築方法：行動選択時に考慮するスコアの種類）

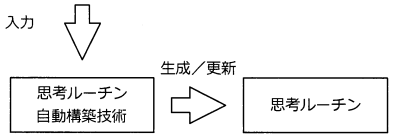
報酬スコア	推定報酬の高さを表すスコア
未知スコア	特徴量空間における密度の大きさを表すスコア
誤差スコア	推定誤差の大きさを表すスコア

【図４３】

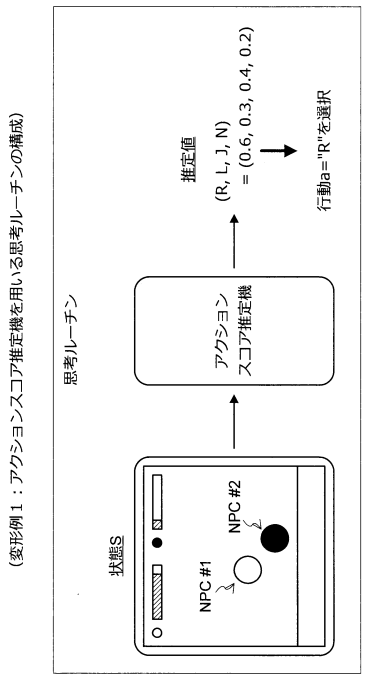
（変形例１：アクションスコア推定機を用いる思考ルーチン）

行動履歴データ

状態S	行動a	報酬r	アクションスコア (R,L,N,J)
S_1	右へ移動(a="R")	0	(0, 1, 1, 1)
S_2	左へ移動(a="L")	1	(0, 1, 0, 0)
S_3	何もせず(a="N")	0	(1, 1, 0, 1)
S_4	ジャンプ(a="J")	1	(0, 0, 0, 1)



【図４４】

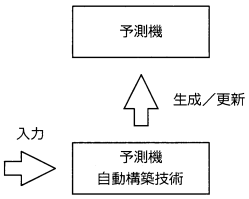


【図４５】

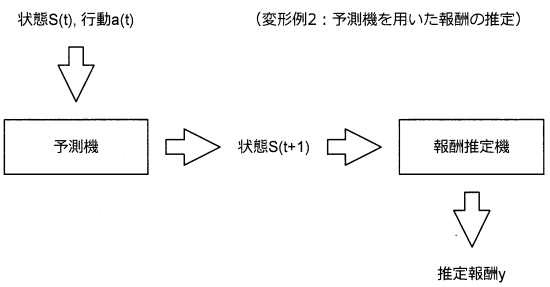
（変形例2：予測機を用いた報酬の推定）

行動履歴データ

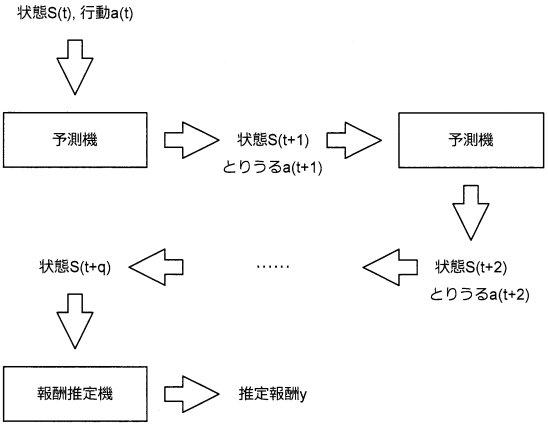
時刻t	状態S	行動a	報酬r
t_1	S_1	右へ移動	0
t_2	S_2	何もせず	1
⋮	⋮	⋮	⋮
t_L	S_L	ジャンプ	0



【図４６】

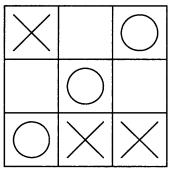


【図４７】



【図４８】

（「三目並べ」への応用）

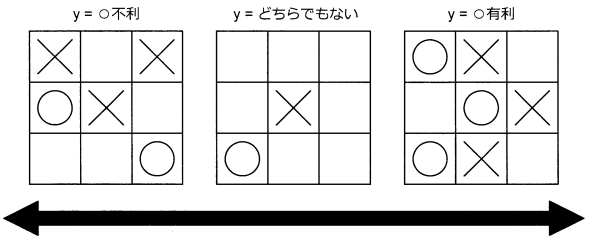


<三目並べのルール>
・互いに交互に手を打つ
・先に3つ1列に並べた方が勝ち

■盤面＝S、手=a

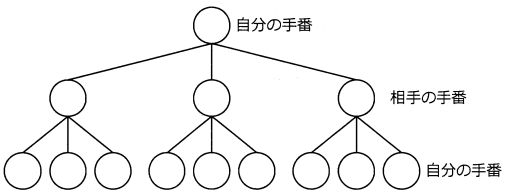
【図４９】

（「三目並べ」への応用）



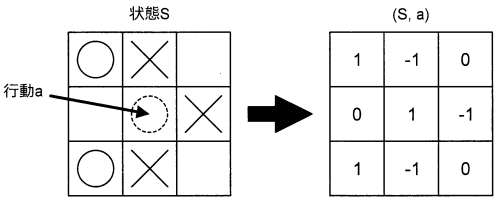
【図５０】

（「三目並べ」への応用）



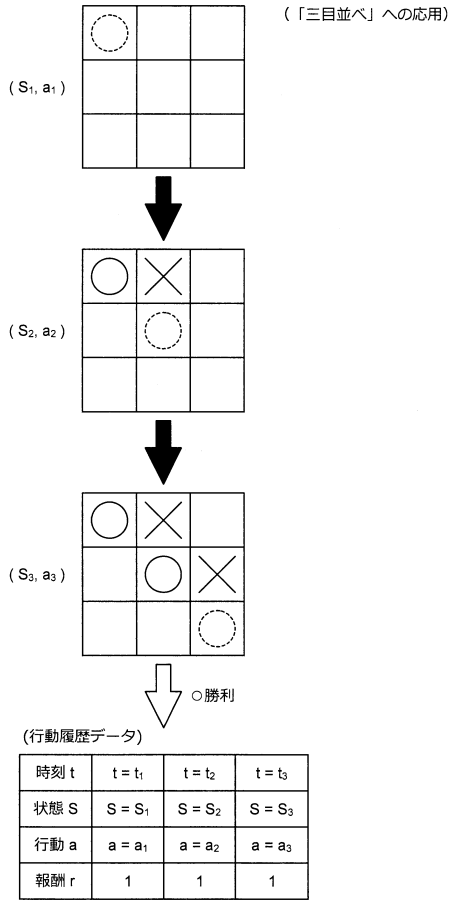
【図５１】

（「三目並べ」への応用）

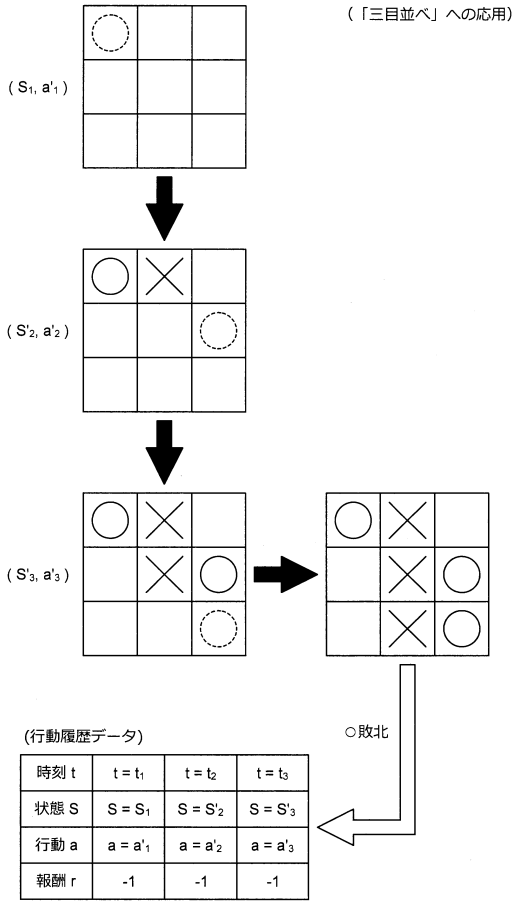


・状態S、行動aを3×3のマトリックスで表現
→ 状態Sは、自分の手番となった時点の盤面
→ 自分の手を反映した盤面が(S, a)
・○(自分)を"1"、×(相手)を"-1"、空白を"0"で表現

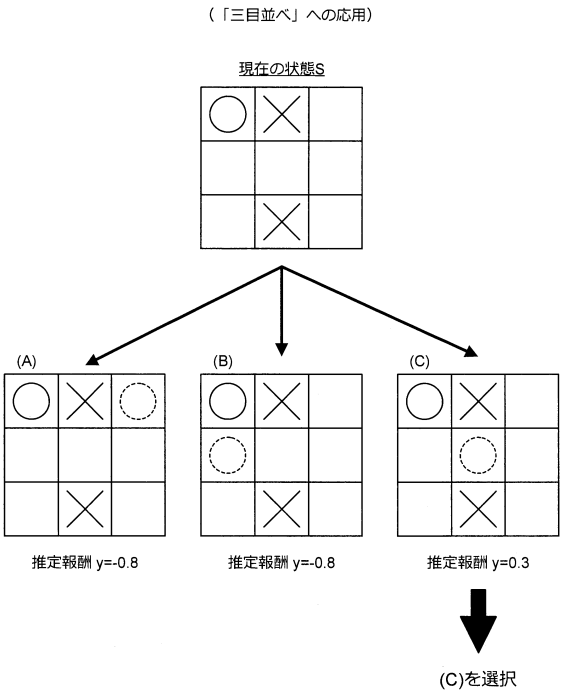
【図５２】



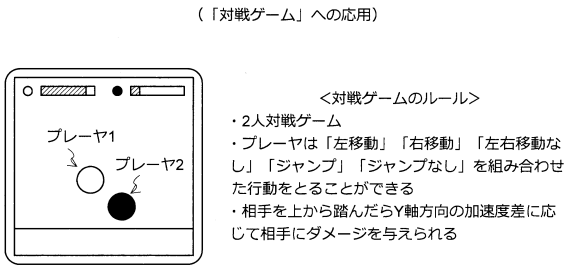
【図５３】



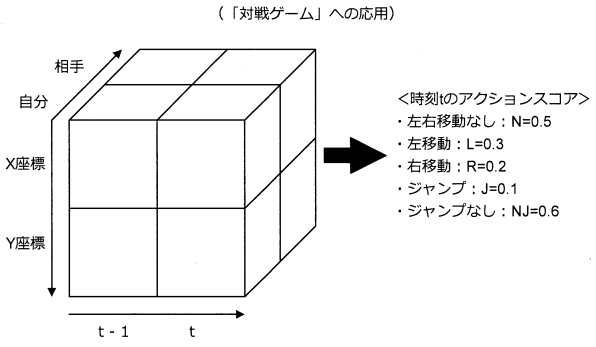
【図５４】



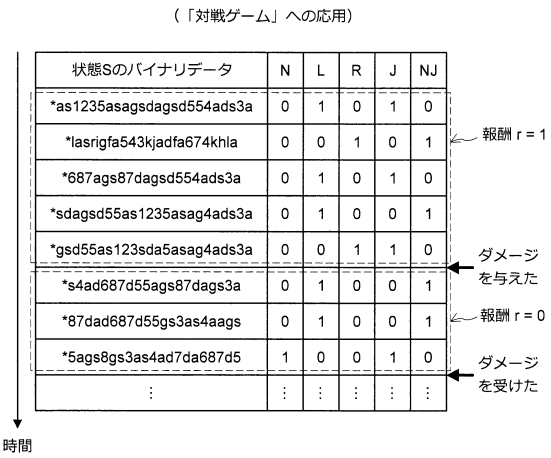
【図５５】



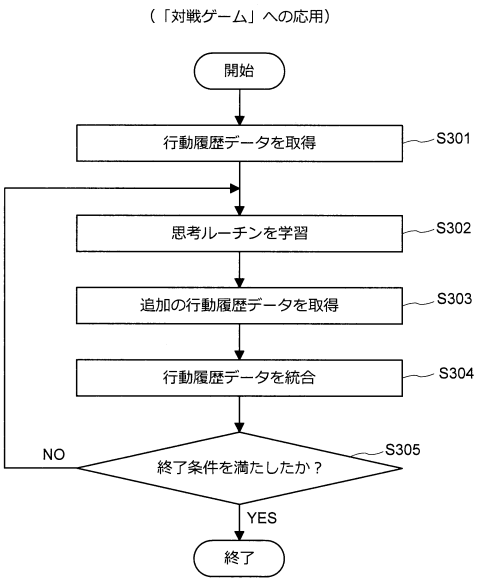
【図５６】



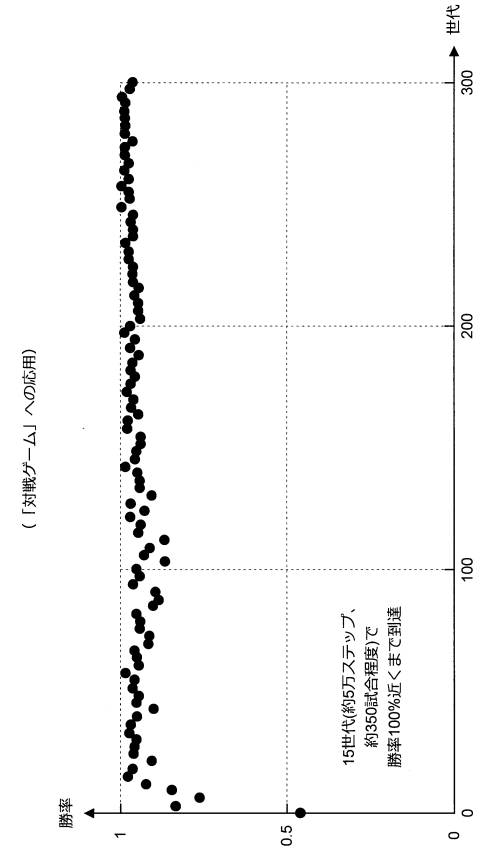
【図５７】



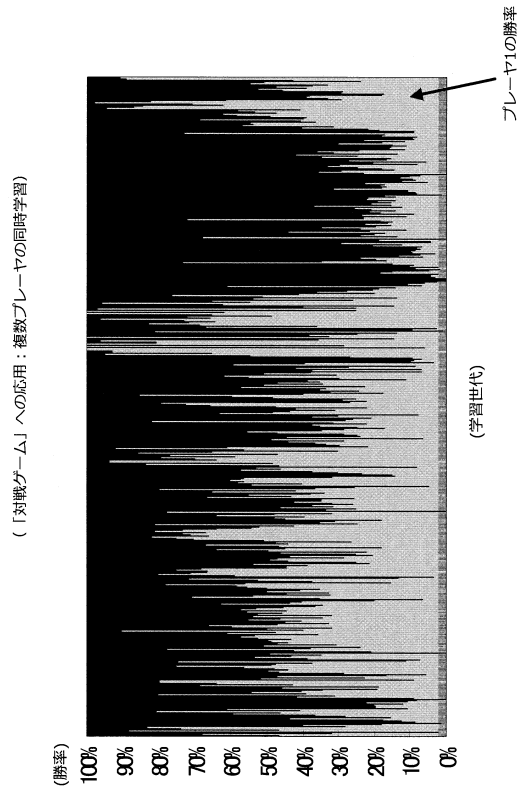
【図５８】



【図５９】



【図６０】



【図６１】

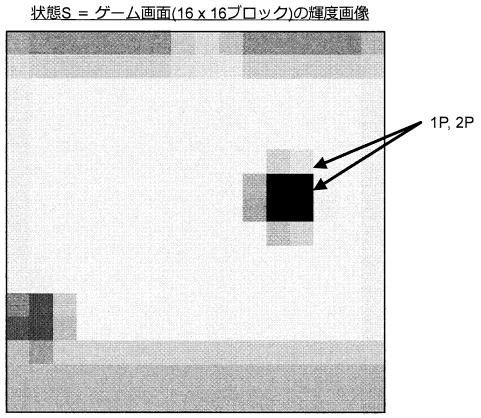
（「対戦ゲーム」への応用：複数プレーヤの同時学習）

学習後1000試合の評価結果

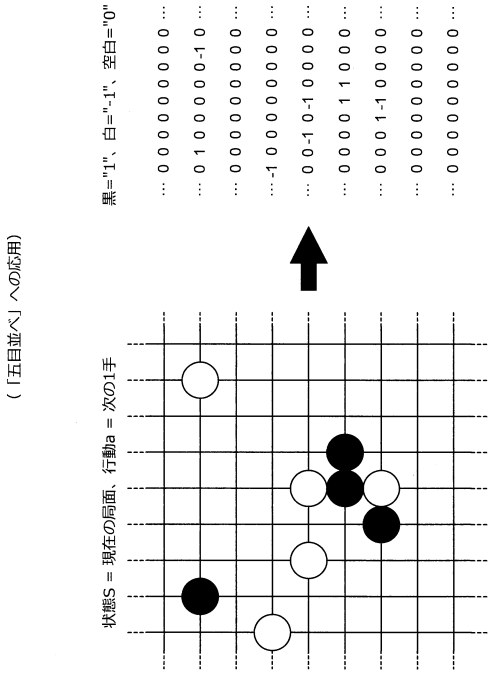
	対戦勝率	ランダム相手
1P	90.6 %	92.4 %
2P	9.4 %	91.5 %

【図６２】

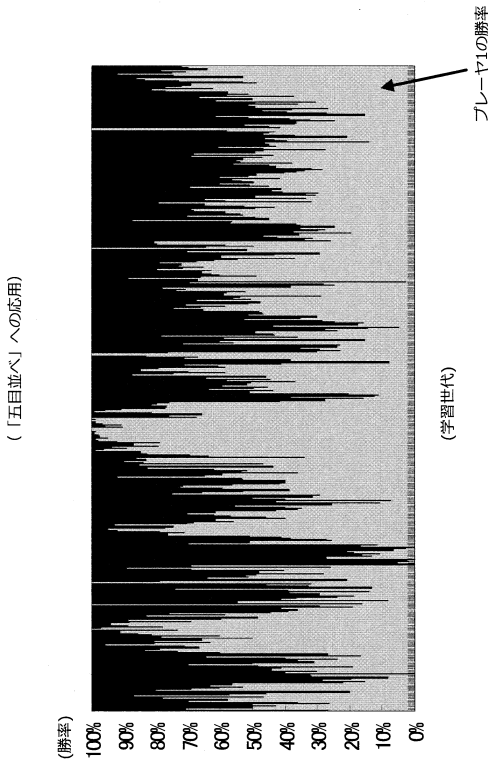
（「対戦ゲーム」への応用）



【図６３】

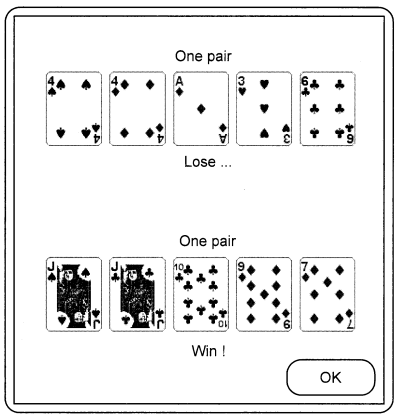


【図６４】



【図６５】

（「ポーカー」への応用）

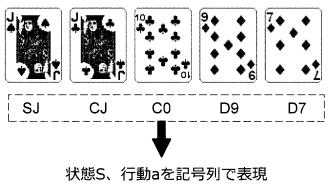


<ポーカーのルール>
・5枚のカードを配る
・捨てるカードを選択
・役の強い方が勝ち

捨てるカードを決める
思考ルーチンを構築

【図６６】

（「ポーカー」への応用）



状態S、行動aを記号列で表現

【図６７】

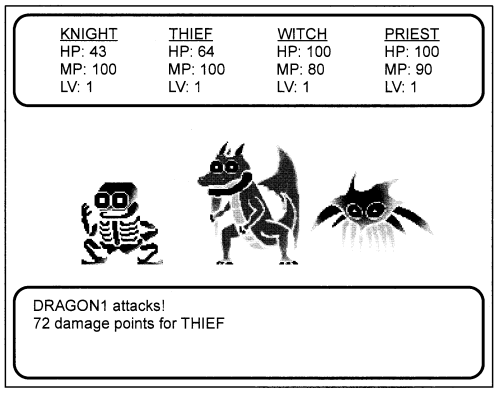
（「ポーカー」への応用）

行動履歴データ

状態S：行動aを表す記号列	報酬r
D7SJHKCJS9：SJ HK CJ S9	1
H5DJH0D4DQ：D4	1
S6CQSJSKH0：CQ SK	1
CQHQDAD9D5：HQ D9	1
C2D2H4HQDA：D2 H4	0
…	…

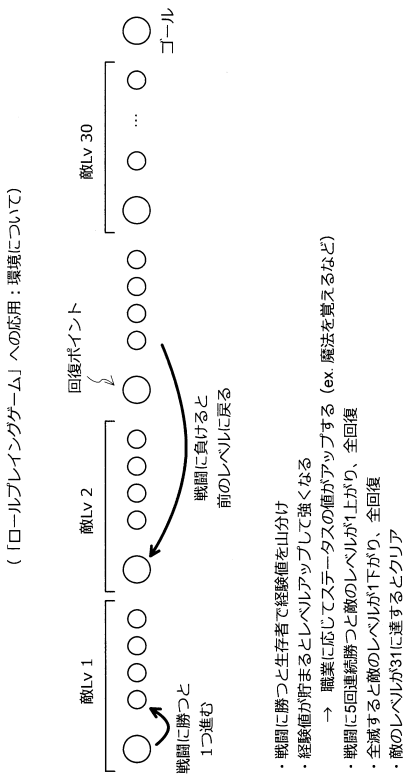
【図６８】

（「ロールプレイングゲーム」への応用）



<RPGルール>
・4人パーティ　　・敵3~5匹
・状態S = プレーヤに提供される情報（ex. ステータス、敵の種類など）
・行動a = コマンド
（ex. 戦う→ターゲット、魔法→魔法の種類→ターゲット）

【図６９】



【図７０】

（「ロールプレイングゲーム」への応用：職業について）

	ステータス上昇率（倍）					魔法を覚えるLv				魔法効果（倍）		
職業	最大HP	最大MP	攻撃力	防御力	素早さ	Heal	Fire	Ice	Heal	Fire	Ice
KNIGHT	1.06	1.00	1.06	1.05	1.04				1	1	1
THIEF	1.05	1.00	1.05	1.05	1.06				1	1	1
WITCH	1.04	1.07	1.04	1.04	1.05		0	5	1	1	1
PRIEST	1.05	1.05	1.04	1.05	1.05	0			1	1	1
SLIME	1.05	1.05	1.05	1.05	1.05	10	10	15	1	1	1
UNDEAD	1.05	1.05	1.05	1.06	1.04			15	0	1.5	0.5
DRAGON	1.06	1.05	1.06	1.05	1.05		5	10	1	0.5	1.5
MACHINE	1.05	1.00	1.05	1.05	1.06				1	0	0

【図７１】

（「ロールプレイングゲーム」への応用：状態Sに含まれる味方情報）

	Lv	職業							
		KNIGHT	THIEF	WITCH	PRIEST	SLIME	UNDEAD	DRAGON	MACHINE
味方1				当てはまるところは"1"、それ以外は"0"					
味方2									
味方3									

	HP	最大HP	MP	最大MP	攻撃力	防御力	素早さ

【図７２】

（「ロールプレイングゲーム」への応用：状態Sに含まれる敵情報）

	Lv	職業								累積ダメージ
		KNIGHT	THIEF	WITCH	PRIEST	SLIME	UNDEAD	DRAGON	MACHINE	
敵1										
敵2										
敵3										

【図７３】

（「ロールプレイングゲーム」への応用：行動a（生存している味方））

	行動者	行動対象	行動の種類			
			攻撃	Heal	Fire	Ice
味方1		対象に"1"、それ以外は"0"				
味方2						
味方3						

【図７４】

（「ロールプレイングゲーム」への応用：行動a（敵））

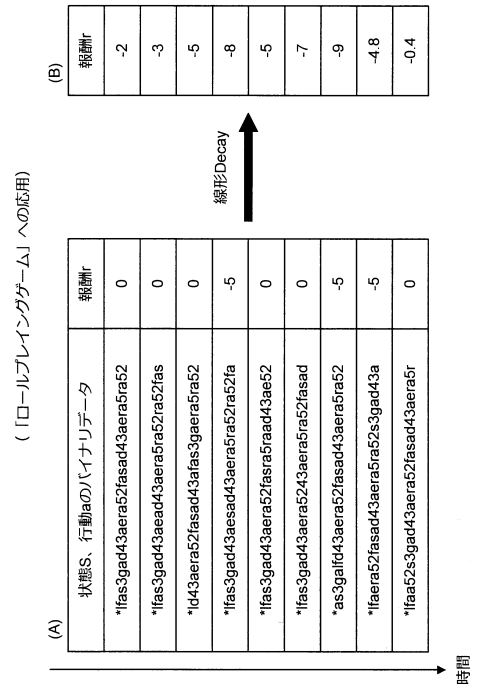
	行動対象
敵1	
敵2	
敵3	

【図７５】

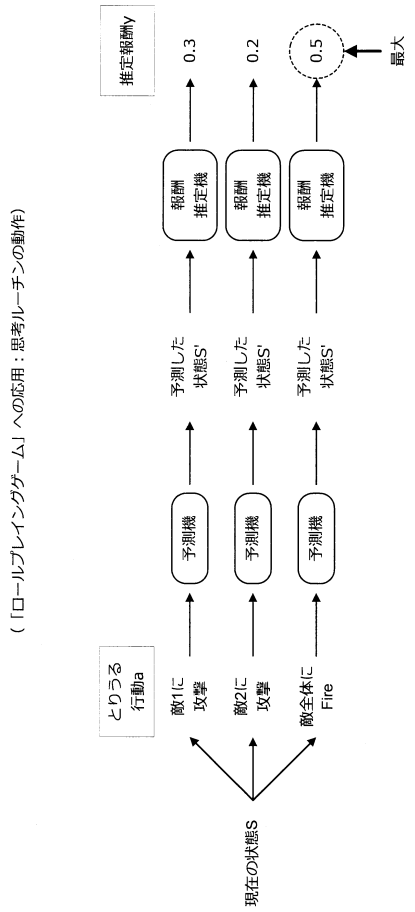
（「ロールプレイングゲーム」への応用）

時刻	状態S	行動a	報酬r
0	状態S_0	敵1を攻撃 (a_0)	0
1	状態S_1	味方1にHeal (a_1)	-5 （味方がやられた）
2	状態S_2	敵2を攻撃 (a_2)	0
3	状態S_3	敵全体にFire (a_3)	1 （敵を倒した）

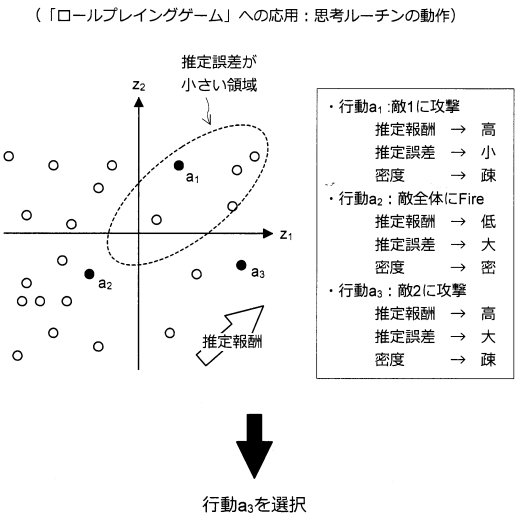
【図７６】



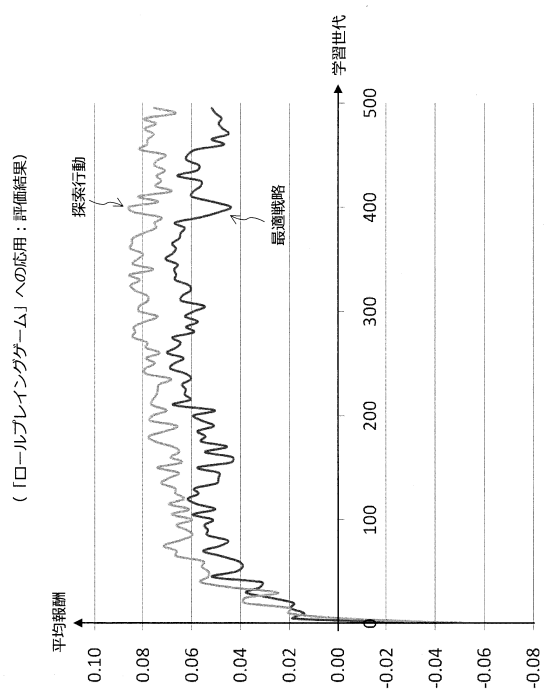
【図７７】



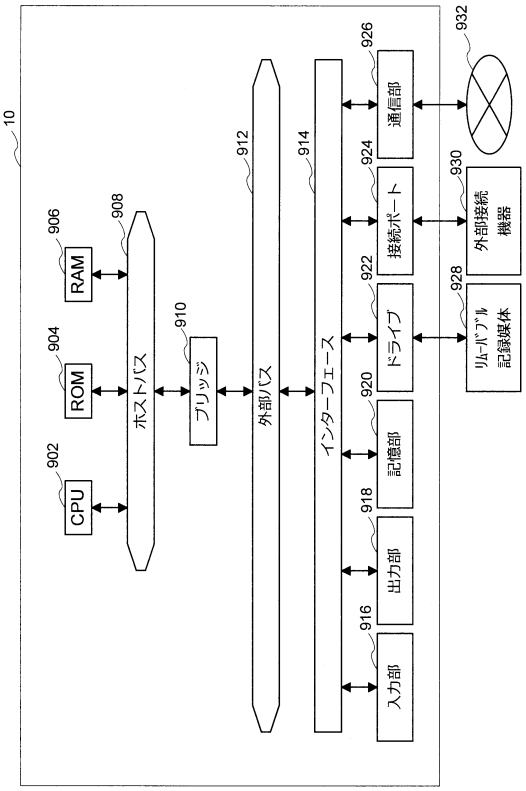
【図７８】



【図７９】



【図８０】


10
902
CPU
904
ROM
906
RAM
908
ホストバス
910
ブリッジ
912
外部バス
914
インターフェース
916
入力部
918
出力部
920
記憶部
922
ドライブ
924
接続ポート
926
通信部
928
リムーバブル記録媒体
930
外部接続機器
932

フロントページの続き

審査官　多賀　実

(56)参考文献　特開２０１３－０５８０９５（ＪＰ，Ａ）
木村　元　外１名，"ロボットの強化学習における状態－行動空間の汎化"，日本ロボット学会誌，社団法人日本ロボット学会，２００４年　3月１５日，第２２巻，第2号，ｐｐ．１６１－１６４
堀之内　剛史　外２名，"遺伝的プログラミングを用いたゲームの局面評価関数の学習"，電子情報通信学会技術研究報告，社団法人電子情報通信学会，１９９７年　3月１８日，第９６巻，第５９４号，ｐｐ．１７－２４
藤井　叙人　外１名，"戦略型トレーディングカードゲームのための戦略獲得手法"，情報処理学会論文誌，社団法人情報処理学会，２００９年１２月１５日，第５０巻，第１２号，ｐｐ．２７９６－２８０６
生天目　章　外２名，"エージェント間の相互作用：望ましい関係性の創発"，計測と制御，社団法人計測自動制御学会，２００５年１２月１０日，第４４巻，第１２号，ｐｐ．８６５－８７４
Pieter Abbeel 外１名，"Apprenticeship learning via inverse reinforcement learning"，ICML '04 Proceedings of the twenty-first international conference on Machine learning，米国，ACM，２００４年，ｐｐ．１－８

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｎ　３／００－９９／００
Ａ６３Ｆ１３／００－１３／９８