

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7009971号
(P7009971)

(45)発行日 令和4年1月26日(2022.1.26)

(24)登録日 令和4年1月17日(2022.1.17)

(51)国際特許分類 F I
G 0 6 F 9/48 (2006.01) G 0 6 F 9/48 3 0 0 H

請求項の数 8 (全27頁)

(21)出願番号	特願2017-239656(P2017-239656)	(73)特許権者	000004237 日本電気株式会社 東京都港区芝五丁目7番1号
(22)出願日	平成29年12月14日(2017.12.14)	(74)代理人	100103894 弁理士 家入 健
(65)公開番号	特開2019-106120(P2019-106120 A)	(72)発明者	三塩 武徳 東京都港区芝五丁目7番1号 日本電気 株式会社内
(43)公開日	令和1年6月27日(2019.6.27)	審査官	坂庭 剛史
審査請求日	令和2年11月9日(2020.11.9)		

最終頁に続く

(54)【発明の名称】 プロセススケジューリング装置およびプロセススケジューリング方法

(57)【特許請求の範囲】

【請求項1】

プロセスが有する少なくとも1つの特徴を用いて、複数のプロセスを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも1つの特徴を用いて複数生成するパターン生成部と、

前記分類パターンに含まれるプロセスタイプの各々に重み付けを行う重み変数の集合である重み変数セットを複数生成する変数生成部と、

前記分類パターンと前記重み変数セットとの組み合わせ毎に、予め定められた性能項目に対する評価値を決定する評価部と、

前記決定された評価値に基づいて、前記複数のプロセスに対する分類パターンおよび重み変数セットを決定する決定部と、を備える、プロセススケジューリング装置。

【請求項2】

前記パターン生成部は、所定の条件を満たすまで、前記決定された評価値に基づいて新たな複数の前記分類パターンを繰り返し生成する、請求項1に記載のプロセススケジューリング装置。

【請求項3】

前記変数生成部は、所定の条件を満たすまで、前記決定された評価値に基づいて新たな複数の前記重み変数セットを繰り返し生成する、請求項1又は2に記載のプロセススケジューリング装置。

【請求項4】

前記分類パターンと前記重み変数セットとの組み合わせ毎に、前記分類パターンに含まれるプロセスタイプ毎の数と前記重み変数セットとを用いて、プロセスタイプ毎の優先度を算出する算出部をさらに備え、

前記評価部は、前記分類パターンと前記重み変数セットとの組み合わせ毎に、前記算出されたプロセスタイプ毎の優先度に基づいて、前記複数のプロセスを実行し、前記評価値を決定する、請求項 1 ~ 3 のいずれか 1 項に記載のプロセススケジューリング装置。

【請求項 5】

前記複数のプロセスを表すプロセス群を複数生成するプロセス生成部をさらに備え、前記パターン生成部は、前記複数のプロセス群の各々に対して、前記複数の分類パターンを適用し、

前記変数生成部は、前記プロセス群と前記分類パターンとの組み合わせ毎に、前記複数の重み変数セットを生成し、

前記評価部は、前記プロセス群と前記分類パターンと前記重み変数セットとの組み合わせ毎に、前記評価値を決定し、各プロセス群の前記評価値の総和を、前記分類パターンと前記重み変数セットとの組み合わせに対する評価値と決定し、

前記決定部は、前記決定された評価値に基づいて、前記複数のプロセス群に対する分類パターンおよび重み変数セットを決定する、請求項 1 ~ 4 のいずれか 1 項に記載のプロセススケジューリング装置。

【請求項 6】

前記性能項目は、スループット、消費電力および割り当てられたリソースの平等性の少なくとも 1 つを含む、請求項 1 ~ 5 のいずれか 1 項に記載のプロセススケジューリング装置。

【請求項 7】

前記プロセスが有する少なくとも 1 つの特徴は、フォアグラウンドプロセスまたはバックグラウンドプロセスであること、プロセス実行順、プロセス発生順、過去に割り当てられたプロセスの CPU 時間、プロセス名、ユーザ情報およびプロセスに入力された入力情報のうち、少なくとも 1 つを含む、請求項 1 ~ 6 のいずれか 1 項に記載のプロセススケジューリング装置。

【請求項 8】

プロセスが有する少なくとも 1 つの特徴を用いて、複数のプロセスを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも 1 つの特徴を用いて複数生成することと、

前記分類パターンに含まれるプロセスタイプの各々に重み付けを行う重み変数の集合である重み変数セットを複数生成することと、

前記分類パターンと前記重み変数セットとの組み合わせ毎に、予め定められた性能項目に対する評価値を決定することと、

前記決定された評価値に基づいて、前記複数のプロセスに対する分類パターンおよび重み変数セットを決定することと、を含むプロセススケジューリング方法。

【発明の詳細な説明】

【技術分野】

【0001】

本開示は、プロセススケジューリング装置、プロセス実行装置およびプロセススケジューリング方法に関する。

【背景技術】

【0002】

典型的なコンピュータシステムでは、タスクを実行するアプリケーションが 1 または複数のプロセスを生成し、CPU (Central Processing Unit) が、生成されたプロセスを実行する。プロセスを実行するために、コンピュータシステムのリソースを必要とする。しかし、一般的に、割り当てられるリソースは限られているため、同時に全てのプロセスを実行することは出来ない。

【0003】

10

20

30

40

50

そのため、典型的なコンピュータシステムにおいては、複数のプロセスが共有のリソースを交互に利用することにより、限られたリソースを用いて、より多くのプロセスを実行するマルチタスキングを実現している。これにより、コンピュータシステムにおいて、複数のプロセスを同時に処理するようにみせている。

【0004】

ここで、より多くのプロセスを処理するためには、各プロセスをどの順番で実行するかを決定するプロセススケジューラが必要になる。プロセススケジューラは、プロセス実行時の性能項目値が高くなるように、複数のプロセスの実行順序を決定する。そのため、プロセススケジューラをどのように設計するかにより、プロセス実行時の性能項目値が左右される。そこで、実行されるプロセスの最適な優先度を決定する技術が検討されている（例えば、特許文献1）。

10

【0005】

特許文献1には、システム内時間を入力とし、遅延コストまたは優先順位を出力とする遅延コスト関数を定義し、多層ニューラルネットワークによって実施されることにより、プロセス実行優先度を決定する技術が開示されている。

【先行技術文献】

【特許文献】

【0006】

【文献】特開平7-175669号公報

【発明の概要】

20

【発明が解決しようとする課題】

【0007】

特許文献1においては、遅延コスト関数は、システム内時間を入力とする関数となっている。換言すると、特許文献1においては、システム内時間であるプロセスが有する1つの特徴に基づいてプロセス実行優先度を決定している。しかしながら、プロセスが有する特徴は1つでは無いことから、特許文献1に開示された技術では、最適なプロセス実行優先度となっていないとは限らない。また、特許文献1には、遅延コスト関数を、複数の変数の関数とした場合についても言及しているが、具体的な内容が開示されていない。したがって、プロセスが有する複数の特徴を用いたプロセススケジューラを提供することが望まれる。

30

【0008】

本開示の目的は、このような問題を解決するためになされたものであり、プロセスが有する複数の特徴を用いたプロセススケジューリングを可能とする、プロセススケジューリング装置、プロセス実行装置およびプロセススケジューリング方法を提供することである。

【課題を解決するための手段】

【0009】

第1の態様にかかるプロセススケジューリング装置は、プロセスが有する少なくとも1つの特徴を用いて、複数のプロセスを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも1つの特徴を用いて複数生成するパターン生成部と、前記複数の分類パターンの各々に対して、予め定められた性能項目に対する評価値を決定する評価部と、前記決定された評価値に基づいて、前記複数のプロセスに対する分類パターンを決定する決定部と、を備える。

40

【0010】

第2の態様にかかるプロセス実行装置は、プロセスが有する少なくとも1つの特徴を用いて、複数の第1のプロセスを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも1つの特徴を用いて複数生成し、前記複数の分類パターンの各々に対して、予め定められた性能項目に対する評価値を決定し、前記決定された評価値に基づいて決定された分類パターンを用いて、アプリ

50

ケーションプログラムが生成する複数の第 2 のプロセスを実行するプロセス実行部を備える。

【 0 0 1 1 】

第 3 の態様にかかるプロセススケジューリング方法は、プロセスが有する少なくとも 1 つの特徴を用いて、複数のプロセスを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも 1 つの特徴を用いて複数生成することと、前記複数の分類パターンの各々に対して、予め定められた性能項目に対する評価値を決定することと、前記決定された評価値に基づいて、前記複数のプロセスに対する分類パターンを決定することと、を含む。

10

【発明の効果】

【 0 0 1 2 】

上述した態様によれば、プロセスが有する複数の特徴を用いたプロセススケジューリングが可能となる。

【図面の簡単な説明】

【 0 0 1 3 】

【図 1】実施の形態 1 にかかるプロセススケジューリング装置の構成例を示すブロック図である。

【図 2】実施の形態 2 にかかるプロセス実行システムの構成例を示すブロック図である。

20

【図 3】学習ケースを説明するための図である。

【図 4】モデリングおよびモデリングパターンを説明するための図である。

【図 5】実施の形態 2 にかかるプロセススケジューリング装置の全体動作を説明するフローチャートである。

【図 6】最適重み変数セットの決定動作について説明するフローチャートである。

【図 7】最適モデリングパターンおよび最適重み変数セットの決定動作について説明するフローチャートである。

【図 8】実施の形態 2 にかかるプロセス実行装置の動作例を説明するフローチャートである。

【図 9】実施の形態 2 の変形例にかかるプロセススケジューリング装置 2 0 の構成例を示すブロック図である。

30

【図 10】実施の形態 2 の変形例にかかるプロセス実行装置の構成例を示すブロック図である。

【図 11】他の実施の形態にかかるプロセススケジューリング装置およびプロセス実行装置の構成例を示す図である。

【発明を実施するための形態】

【 0 0 1 4 】

以下、図面を参照しつつ、本開示の実施の形態について説明する。なお、実施の形態において、同一の要素には、同一の符号を付し、重複する説明は省略される。

(実施の形態 1)

40

まず、図 1 を用いて、実施の形態 1 にかかるプロセススケジューリング装置 1 0 について説明する。図 1 は、実施の形態 1 にかかるプロセススケジューリング装置の構成例を示すブロック図である。

【 0 0 1 5 】

プロセススケジューリング装置 1 0 は、例えば、サーバ、パーソナルコンピュータ等の情報処理装置である。プロセススケジューリング装置 1 0 は、複数のプロセスの実行時の優先度を決定する情報処理装置である。プロセススケジューリング装置 1 0 は、パターン生成部 1 1 と、評価部 1 2 と、決定部 1 3 とを備える。

【 0 0 1 6 】

パターン生成部 1 1 は、プロセスが有する少なくとも 1 つの特徴を用いて、複数のプロセ

50

スを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも1つの特徴を用いて複数生成する。換言すると、パターン生成部11は、分類パターンを生成する際に用いる特徴の全てが重複する分類パターンを生成しないように複数の分類パターンを生成する。

【0017】

プロセスが有する特徴は、フォアグラウンドプロセスまたはバックグラウンドプロセスであること、プロセス実行順、プロセス発生順であってもよい。もしくは、プロセスが有する特徴は、過去に割り当てられたプロセスのCPU時間、プロセス名、ユーザ情報およびプロセスに入力された入力情報であってもよい。

【0018】

評価部12は、パターン生成部11が生成した複数の分類パターンの各々に対して、予め定められた性能項目に対する評価値を決定する。評価部12は、パターン生成部11が生成した複数の分類パターンに含まれるプロセスタイプを実行することによって、性能項目に対する評価値を決定する。

【0019】

性能項目は、スループット、消費電力、割り当てられたリソースの平等性であってもよい。もしくは、性能項目は、上記のうちの少なくとも1つを含む組み合わせであってもよい。

【0020】

性能項目がスループット、消費電力、リソース割り当て率の少なくとも1つを含む性能項目である場合、評価部12は、各性能項目の評価値の総和を評価値と決定してもよい。

【0021】

決定部13は、評価部12によって決定された評価値に基づいて、複数のプロセスに対する分類パターンを決定する。

【0022】

以上説明した様に、パターン生成部11が、プロセスが有する複数の特徴を用いて、複数のプロセスを複数のプロセスタイプに分類する分類パターンを、複数生成する。評価部12は、パターン生成部11が生成した複数の分類パターンの各々に対して、性能項目に対する評価値を決定し、決定部13は、決定された評価値に基づいて、分類パターンを決定する。プロセスを実行する場合、決定された分類パターンを用いてプロセスを実行することにより最適なプロセススケジューリングを行うことが可能となる。したがって、実施の形態1にかかるプロセススケジューリング装置10によれば、プロセスが有する複数の特徴を用いたプロセススケジューリングが可能となる。

【0023】

(実施の形態2)

続いて、実施の形態2について説明する。実施の形態2は、実施の形態1を詳細にした実施の形態である。具体的には、実施の形態1においては、複数の分類パターンに対する評価値を決定して、決定された評価値に基づいて分類パターンを決定した。本実施の形態においては、複数のプロセスを表すプロセス群を複数生成する。さらに、複数の分類パターンにより、プロセスをプロセスタイプに分類し、分類されたプロセスタイプの各々に重み付けを行う重み変数セットを導入する。またさらに、最適な分類パターンと、最適な重み変数セットとを、機械学習によって決定する。本実施の形態の詳細を以下に説明する。

【0024】

<プロセス実行システムの構成例>

まず、図2を用いて、実施の形態2にかかるプロセス実行システム100の構成例について説明する。図2は、実施の形態2にかかるプロセス実行システムの構成例を示すブロック図である。

【0025】

プロセス実行システム100は、プロセスを実行する装置と、プロセスを実行する装置において生成されたプロセスをスケジューリングする装置と、を含む情報処理システムである。プロセス実行システム100は、プロセススケジューリング装置20と、プロセス実

10

20

30

40

50

行装置 30 とを備える。

【0026】

プロセススケジューリング装置 20 は、実施の形態 1 にかかるプロセススケジューリング装置 10 に対応する情報処理装置である。プロセススケジューリング装置 20 は、プロセス実行装置 30 において生成されたプロセスをスケジューリングする情報処理装置である。なお、説明を行う上で便宜的に、プロセス実行装置 30 において生成されたプロセスを「実プロセス」と称して記載をすることがある。

【0027】

プロセス実行装置 30 は、プロセス実行装置 30 にインストールされているアプリケーションプログラムが生成する実プロセスを実行する情報処理装置である。プロセス実行装置 30 は、例えば、サーバ、パーソナルコンピュータ等の情報処理装置であってもよい。

10

【0028】

<プロセススケジューリング装置の構成例>

続いて、プロセススケジューリング装置 20 の構成例について説明する。図 2 に示す様に、プロセススケジューリング装置 20 は、プロセス生成部 21、パターン生成部 22、変数生成部 23、記憶部 24、算出部 25、評価部 26、決定部 27 および送信部 28 を備える。

【0029】

プロセス生成部 21 は、プロセス実行装置 30 において生成される実プロセスの実行順序を決定するための複数の評価用プロセスを生成する。つまり、プロセス生成部 21 は、プロセス実行装置 30 において生成されると想定される複数の仮想プロセスを生成する。

20

【0030】

プロセス生成部 21 は、プロセス実行装置 30 において生成される実プロセス数に応じて、評価用のプロセスを生成してもよい。例えば、プロセス実行装置 30 においてインストールされているアプリケーションプログラムが 1 万プロセスを生成すると予め分かっている場合、プロセス生成部 21 は、1 万プロセスを生成してもよい。もしくは、プロセス生成部 21 は、プロセス実行システム 100 の管理者または運用者が、任意に決めた数の評価用プロセスを生成してもよい。

【0031】

プロセス生成部 21 は、生成した複数の評価用プロセスのうち、ランダムに 2 つ以上の評価用プロセスを抽出して、評価用のプロセスの集合を表す評価用プロセス群を複数生成する。プロセス生成部 21 は、プロセス実行装置 30 において、同時に実行される可能性がある評価用プロセスの集合である評価用プロセス群を複数生成する。つまり、プロセススケジューリング装置 20 は、プロセス実行装置 30 において生成されると想定される複数の仮想プロセスのうち、同時に実行されると想定される仮想プロセスの集合を複数生成して評価を行う。なお、以降の説明では、プロセス生成部 21 が生成する評価用プロセス群を「学習ケース」と称して記載することがある。また、説明を行う上で便宜的に、プロセス生成部 21 は、複数の学習ケース a_1, a_2, \dots, a_m ($m: 1$ 以上の整数) を生成したと仮定して説明を行う。

30

【0032】

プロセス生成部 21 は、生成した学習ケース a_1, a_2, \dots, a_m を、記憶部 24 の学習ケース保存領域 241 に記憶する。

40

【0033】

ここで、図 3 を用いて、学習ケースの一例について説明する。図 3 は、学習ケースを説明するための図である。図 3 に示す様に、例えば、プロセス生成部 21 がプロセス 1 ~ 5 の評価用プロセスを生成したとする。プロセス生成部 21 は、生成したプロセス 1 ~ 5 のうち、2 つ以上の評価用プロセスを抽出して学習ケースを生成する。

【0034】

図 3 に示す一例では、プロセス 1 とプロセス 2 とを抽出して生成した学習ケースを a_1 とし、プロセス 1 とプロセス 3 とを抽出して生成した学習ケースを a_2 としたことを示して

50

いる。また、学習ケースは、2つのプロセスのみに限られないため、図3の学習ケースa₅のように3つのプロセスによって生成された学習ケースが存在してもよい。学習ケースは、多ければ多いほど最適な評価結果を導けるため、プロセス数が4つにより生成された学習ケースが存在してもよいし、プロセス数が5つにより生成された学習ケースが存在してもよい。

【0035】

図2に戻り、パターン生成部22について説明する。パターン生成部22は、実施の形態1にかかるパターン生成部11に対応する。パターン生成部22は、プロセスが有する少なくとも1つの特徴を用いて、複数のプロセスを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも1つの特徴を用いて複数生成する。換言すると、パターン生成部11は、分類パターンを生成する際に用いる特徴の全てが重複する分類パターンを生成しないように複数の分類パターンを生成する。

10

【0036】

なお、以降の説明では、複数のプロセスを複数のプロセスタイプに分類することを「モデリング」と称して記載することがある。また、パターン生成部22が生成した複数の分類パターンを「モデリングパターン」と称して記載することがある。また、説明を行う上で便宜的に、パターン生成部22は、複数のモデリングパターン b_1, b_2, \dots, b_n ($n: 1$ 以上の整数)を生成したと仮定して説明を行う。なお、 m と n は、同じ整数であってもよいし、異なる整数であってもよい。

【0037】

パターン生成部22は、生成したモデリングパターン b_1, b_2, \dots, b_n を、記憶部24のモデリングパターン保存領域242に記憶する。

20

【0038】

ここで、図4を用いて、モデリングおよびモデリングパターンについて説明する。図4は、モデリングおよびモデリングパターンを説明するための図である。まず、図4の上図を用いてモデリングについて説明する。図4の上図には、プロセス1~3を含む、ある学習ケースが示されている。プロセス1~3は2つの特徴を有しており、1つ目の特徴は、フォアグラウンドプロセスまたはバックグラウンドプロセスのいずれであるかを示す属性情報であるとする。また、2つ目の特徴は、プロセスの生成順であるとする。プロセス1は、フォアグラウンドのプロセスであって、1番目に生成されたプロセスであることを示している。プロセス2は、バックグラウンドのプロセスであって、2番目に生成されたプロセスであることを示している。プロセス3は、フォアグラウンドのプロセスであって、3番目に生成されたプロセスであることを示している。

30

【0039】

この状態で、2つの特徴のうち、少なくとも1つの特徴を用いて、プロセスをプロセスタイプに分類することをモデリングと言う。図4の上図では、フォアグラウンドプロセスまたはバックグラウンドプロセスのいずれであるかを示す属性情報を用いて、プロセスをプロセスタイプに分類した場合を示している。つまり、フォアグラウンドプロセスであるプロセス1および3はプロセスタイプTYPE 1に分類され、バックグラウンドプロセスであるプロセス2はプロセスタイプTYPE 2に分類されることを示している。このように、複数のプロセスを複数のプロセスタイプに分類することをモデリングと言う。

40

【0040】

次に、モデリングパターンについて説明する。図4の下図は、2つ目の特徴であるプロセスの生成順を用いてモデリングを行ったことを示している。図4の下図では、1番目に生成されるプロセスをプロセスタイプTYPE 1とし、2番目以降に生成されるプロセスをプロセスタイプTYPE 2とするモデリングが行われていることを示している。ここで、図4の上図、下図のように、異なる特徴を用いてモデリングされた分類パターン(パターンXまたはパターンY)をそれぞれモデリングパターンとする。

【0041】

図2に戻り説明を続ける。パターン生成部22は、後述する評価部26が決定した評価値

50

に基づいて、所定の条件を満たすまで、新たな複数のモデリングパターンを繰り返し生成する。詳細は動作例において説明を行うが、パターン生成部 2 2 は、モデリングパターンを個体とし、評価部 2 6 が決定した評価値を適応度とした、遺伝的アルゴリズムを適用して、新たな複数のモデリングパターンを繰り返し生成する。

【 0 0 4 2 】

所定の条件は、遺伝的アルゴリズムを適用して、学習を行う全体の経過時間が所定の閾値を超過した場合であってもよい。もしくは、所定の条件は、遺伝的アルゴリズムを適用して世代交代をしたときの所定の世代数となった場合であってもよい。もしくは、所定の条件は、遺伝的アルゴリズムを適用して生成された複数のモデリングパターンに対する評価部 2 6 が決定した評価値が全て同じ値となる場合であってもよい。

10

【 0 0 4 3 】

パターン生成部 2 2 は、新たな複数のモデリングパターンを生成した場合、記憶部 2 4 のモデリングパターン保存領域 2 4 2 に記憶されているモデリングパターンを更新する。

【 0 0 4 4 】

また、パターン生成部 2 2 は、学習ケース a_1, a_2, \dots, a_m の各々に、モデリングパターン b_1, b_2, \dots, b_n を適用する。なお、以降の説明では、学習ケースにモデリングパターンを適用して生成されたものを「学習モデル」と称して記載することがある。

【 0 0 4 5 】

ここで、学習ケース a_i ($i = 0 \sim m$ の整数) に対して、モデリングパターン b_j ($j = 0 \sim n$ の整数) を適用して生成された学習モデルを c_{ij} とすると、学習モデル c_{ij} は、 $c_{ij} = a_i b_j$ として表すこととする。つまり、パターン生成部 2 2 は、 $c_{11} = a_1 b_1, c_{12} = a_1 b_2, \dots, c_{1n} = a_1 b_n, c_{21} = a_2 b_1, c_{22} = a_2 b_2, \dots, c_{2n} = a_2 b_n, \dots, c_{m1} = a_m b_1, c_{m2} = a_m b_2, c_{mn} = a_m b_n$ を生成する。

20

【 0 0 4 6 】

パターン生成部 2 2 が生成した学習モデル $c_{11} \sim c_{mn}$ を記憶部 2 4 の学習モデル保存領域 2 4 3 に記憶する。

【 0 0 4 7 】

ここで、学習モデル $c_{ij} = a_i b_j$ は、学習ケース a_i に含まれている評価用プロセスが、モデリングパターン b_j によってプロセスタイプに分類されている。学習モデル c_{ij} に含まれているプロセスタイプ毎の数を学習モデル c_{ij} の特徴とし、プロセスタイプ毎にいくつ含まれているかを特徴量とする。

30

【 0 0 4 8 】

例えば、学習モデル c_{ij} が、プロセスタイプ $TYPE_1$ と $TYPE_2$ とを含んでいるとすると、プロセスタイプ $TYPE_1$ の特徴量は m_1 で表され、プロセスタイプ $TYPE_2$ の特徴量は m_2 で表される。一般化して表すと、プロセスタイプ $TYPE_f$ の特徴量は、 m_f として定義出来る。

【 0 0 4 9 】

次に、変数生成部 2 3 について説明する。変数生成部 2 3 は、各学習モデルに含まれるプロセスタイプの各々に重み付けを行う重み変数の集合である重み変数セットを複数生成する。変数生成部 2 3 が生成する重み変数セットは、プロセスタイプ毎の優先度を決定する関数を構成する変数の集合である。

40

【 0 0 5 0 】

ここで、プロセスタイプ毎の優先度を決定する関数について説明を行った上で、変数生成部 2 3 が生成する重み変数セットについて説明する。

プロセスタイプ毎の優先度を決定する関数は、以下の数式 (1) のように定義される。

【 数 1 】

50

$$n_a = \frac{d_a}{\sum_{k=1}^q d_k m_k} \dots(1)$$

ここで、 n_a ：プロセスタイプTYPE aの優先度、 d_a ：プロセスタイプTYPE aの重み変数、 d_k ：プロセスタイプTYPE kの重み変数 m_k ：プロセスタイプTYPE kの特徴（プロセスタイプTYPE kの数）、 q ：1～プロセスタイプ数。

【0051】

変数生成部23は、上記数式(1)において、プロセスタイプ毎の重み変数の集合である $d_1 = (d_{11}, d_{12}, \dots, d_{1q})$ を重み変数セットとして生成する。また、変数生成部23は、複数の重み変数セットをランダムに生成する。変数生成部23が、 d_1, d_2, \dots, d_o (o ：1以上の整数)を生成したとする。具体的には、各重み変数セットは、 $d_1 = (d_{11}, d_{12}, \dots, d_{1q}), d_2 = (d_{21}, d_{22}, \dots, d_{2q}), \dots, d_o = (d_{o1}, d_{o2}, \dots, d_{oq})$ である。

【0052】

なお、適用するモデリングパターンによって、学習モデルに含まれるプロセスタイプ数が異なる場合も想定される。具体的には、ある学習モデルの場合、プロセスタイプを2つ(TYPE 1およびTYPE 2)含み、別の学習モデルの場合、プロセスタイプを3つ(TYPE 1、TYPE 2およびTYPE 3)含むことが想定される。そのため、変数生成部23は、重み変数セットを生成する際に、全ての学習モデルに含まれるプロセスタイプ数を確認し、プロセスタイプ数が最も多い学習モデルのプロセスタイプ数に応じた重み変数を集合とする重み変数セットを生成する。

【0053】

また、変数生成部23は、後述する評価部26が決定した評価値に基づいて、所定の条件を満たすまで、新たな複数の重み変数セットを繰り返し生成する。詳細は動作例において説明を行うが、変数生成部23は、重み変数セットを個体とし、評価部26が決定した評価値を適応度として、遺伝的アルゴリズムを適用して、新たな複数の重み変数セットを繰り返し生成する。

【0054】

所定の条件は、遺伝的アルゴリズムを適用して、学習を行う全体の経過時間が所定の閾値を超過した場合であってもよい。もしくは、所定の条件は、遺伝的アルゴリズムを適用して世代交代をしたときの所定の世代数となった場合であってもよい。もしくは、所定の条件は、遺伝的アルゴリズムを適用して生成された複数の重み変数セットに対する評価部26が決定した評価値が全て同じ値となる場合であってもよい。

【0055】

変数生成部23は、新たな複数の重み変数セットを生成した場合、記憶部24の重み変数セット保存領域244に記憶されている重み変数セットを更新する。

【0056】

次に、記憶部24について説明する。記憶部24には、学習ケース保存領域241、モデリングパターン保存領域242、学習モデル保存領域243および重み変数セット保存領域244が設けられている。

【0057】

学習ケース保存領域241は、プロセス生成部21が生成した学習ケースが保存される領域である。

モデリングパターン保存領域242は、パターン生成部22が生成したモデリングパターンが保存される領域である。

学習モデル保存領域243は、パターン生成部22が生成した学習モデルが保存される領域である。

10

20

30

40

50

重み変数セット保存領域 2 4 4 は、変数生成部 2 3 が生成した重み変数セットが保存される領域である。

【 0 0 5 8 】

次に、算出部 2 5 について説明する。算出部 2 5 は、学習モデルと重み変数セットとの組み合わせ毎に、学習モデルに含まれるプロセスタイプ毎の特徴量、つまり、プロセスタイプ毎の数と、前記重み変数セットとを用いて、プロセスタイプ毎の優先度を算出する。具体的には、算出部 2 5 は、学習モデルと重み変数セットとの組み合わせ毎に、学習モデルに含まれるプロセスタイプ毎の特徴量を求め、重み変数セットを用いて、上記数式 (1) からプロセスタイプ毎の優先度を算出する。

【 0 0 5 9 】

次に、評価部 2 6 について説明する。評価部 2 6 は、実施の形態 1 における評価部 1 2 に対応する。評価部 2 6 は、学習モデルと重み変数セットとの組み合わせ毎に、算出部 2 5 が算出したプロセスタイプ毎の優先度に基づいて、学習モデルに含まれるプロセス (プロセスタイプ) を実行し、予め定められた性能項目の評価値を決定する。具体的には、評価部 2 6 は、算出されたプロセスタイプ毎の優先度が高いプロセスタイプから実行し、性能項目の評価値を決定する。

【 0 0 6 0 】

性能項目は、例えば、スループット、消費電力、割り当てられたリソースの平等性であってもよい。もしくは、予め定められた性能項目は、例えば、スループットおよび消費電力などのように複数の性能項目の組み合わせであってもよい。

【 0 0 6 1 】

性能項目がスループットである場合、プロセスタイプ毎の優先度に基づいて、学習モデルに含まれるプロセス (プロセスタイプ) を実行したときの実行開始から全プロセス (全プロセスタイプ) の終了までの時間を計測する。そして、その計測時間に応じて評価値を決定する。評価値は、例えば、計測されたスループットを所定の閾値と比較し、所定の閾値以上の場合、評価値を 1 として、所定の閾値未満である場合、評価値を 0 としてもよい。もしくは、評価値は、計測されたスループットを所定の閾値と比較し、所定の閾値との差分に応じた評価値としてもよい。

【 0 0 6 2 】

性能項目が消費電力である場合、プロセスタイプ毎の優先度に基づいて、学習モデルに含まれるプロセス (プロセスタイプ) を実行したときの消費電力を監視し、消費電力の大小に応じて評価値を決定する。評価値は、例えば、計測された消費電力を所定の閾値と比較し、所定の閾値未満の場合、評価値を 1 として、所定の閾値以上である場合、評価値を 0 としてもよい。もしくは、評価値は、計測された消費電力を所定の閾値と比較し、所定の閾値との差分に応じた評価値としてもよい。なお、消費電力は小さい程、良い性能を發揮していると考えられるため、計測された消費電力の逆数を用いて評価値を決定してもよい。

【 0 0 6 3 】

性能項目が、割り当てられたリソースの平等性である場合、プロセスタイプ毎の優先度に基づいて、学習モデルに含まれるプロセス (プロセスタイプ) を実行したときの各プロセスに割り当てられたリソースを監視する。そして、割り当てられたリソースのプロセス間のばらつき (分散) に基づいて評価値を決定する。評価値は、例えば、割り当てられたリソースのプロセス間のばらつきを所定の閾値と比較し、所定の閾値未満の場合、評価値を 1 として、所定の閾値以上である場合、評価値を 0 としてもよい。もしくは、評価値は、割り当てられたリソースのプロセス間のばらつきを所定の閾値と比較し、所定の閾値との差分に応じた評価値としてもよい。なお、割り当てられたリソースのプロセス間のばらつきは小さい程、良い性能を發揮していると考えられるため、割り当てられたリソースのプロセス間のばらつきの逆数を用いて評価値を決定してもよい。

【 0 0 6 4 】

性能項目が、例えば、スループットおよび消費電力のように、複数の性能項目の組み合わせである場合、各性能項目に対して、上記のように決定した評価値を総和することにより

10

20

30

40

50

決定してもよい。なお、説明を行う上で便宜的に、性能項目は、スループットとして以降の説明を行う。

【 0 0 6 5 】

詳細は後述するが、プロセススケジューリング装置 2 0 は、最適なモデリングパターンと最適な重み変数セットとを決定することを目的とする。そのため、評価部 2 6 は、モデリングパターンと重み変数セットとの組み合わせ毎に評価値を決定する。ここで、各学習モデルは、学習ケースのそれぞれに、モデリングパターンを適用して生成されている。つまり、同一のモデリングパターンによって生成された学習モデルは、学習ケース分存在する。そのため、モデリングパターンを評価する際、重み変数セットを評価する際、各学習ケースの評価値を決定して、決定した評価値を総和した値を、モデリングパターンに対する評価値、重み変数セットに対する評価値として決定する。

10

【 0 0 6 6 】

次に、決定部 2 7 について説明する。決定部 2 7 は、実施の形態 1 にかかる決定部 1 3 に対応する。決定部 2 7 は、評価部 2 6 において決定された評価値に基づいて、最適なモデリングパターンと最適な重み変数セットとを決定する。パターン生成部 2 2 および変数生成部 2 3 は、所定の条件を満たすまで、新たな複数のモデリングパターンおよび新たな複数の重み変数セットを繰り返し生成する。決定部 2 7 は、所定の条件を満たした場合、最も高い評価値に対応する、モデリングパターンおよび重み変数セットを決定する。なお、説明を行う上で便宜的に、最も良い評価値のモデリングパターンを「最適モデリングパターン」と称して記載し、最も良い評価値の重み変数セットを「最適重み変数セット」と称して記載することがある。

20

【 0 0 6 7 】

次に、送信部 2 8 について説明する。送信部 2 8 は、最適モデリングパターンと、最適重み変数セットとをプロセス実行装置 3 0 に送信する。

【 0 0 6 8 】

< プロセス実行装置の構成例 >

続いて、プロセス実行装置 3 0 について説明する。プロセス実行装置 3 0 は、受信部 3 1、記憶部 3 2 およびプロセス実行部 3 3 を備える。

【 0 0 6 9 】

受信部 3 1 は、最適モデリングパターンと、最適重み変数セットとを送信部 2 8 から受信する。受信部 3 1 は、最適モデリングパターンと、最適重み変数セットとを受信すると、記憶部 3 2 に記憶する。

30

【 0 0 7 0 】

記憶部 3 2 は、モデリングパターン保存領域 3 2 1 と、重み変数セット保存領域 3 2 2 と、を備える。

モデリングパターン保存領域 3 2 1 は、受信部 3 1 が受信した最適モデリングパターンが保存される保存領域である。

重み変数セット保存領域 3 2 2 は、受信部 3 1 が受信した最適重み変数セットが保存される保存領域である。

【 0 0 7 1 】

次に、プロセス実行部 3 3 について説明する。プロセス実行部 3 3 は、プロセス実行装置にインストールされたアプリケーションプログラムが生成した複数の実プロセスを実行する。具体的には、プロセス実行部 3 3 は、複数の実プロセスを最適モデリングパターンに基づいて、複数のプロセスタイプに分類する。そして、プロセス実行部 3 3 は、プロセスタイプ毎の数を算出し、最適重み変数セットを用いて、上記数式 (1) に代入してプロセスタイプ毎の優先度を算出する。プロセス実行部 3 3 は、算出されたプロセスタイプ毎の優先度が高いプロセスタイプに分類された実プロセスから実行する。ただし、同じ優先度のプロセスタイプに対応する実プロセスが複数存在する場合は、プロセス実行部 3 3 は、各プロセスに既に割り振ったリソースの量が少ないものを先に実行する。優先度が等しく、かつ、既に割り振ったリソースの量も等しい場合は、プロセス実行部 3 3 は、先に生成

40

50

されたプロセスから順に実行する。

【0072】

< プロセススケジューリング装置の動作例 >

続いて、図5～図7を用いて、プロセススケジューリング装置20の動作例を説明する。まず、図5を用いて、プロセススケジューリング装置20の全体動作を説明する。図5は、実施の形態2にかかるプロセススケジューリング装置の全体動作を説明するフローチャートである。

【0073】

プロセス生成部21は、複数の学習ケースを生成する(ステップS1)。具体的には、プロセス生成部21は、複数の評価用プロセスを生成する。そして、プロセス生成部21は、生成した複数の評価用プロセスのうち、ランダムに2つ以上の評価用プロセスを抽出して、学習ケースを複数生成する。プロセス生成部21は、複数の学習ケース a_1, a_2, \dots, a_m を生成したとする。プロセス生成部21は、生成した複数の学習ケース a_1, a_2, \dots, a_m を学習ケース保存領域241に記憶する。

10

【0074】

次に、パターン生成部22は、複数のモデリングパターンを生成する(ステップS2)。具体的には、パターン生成部22は、プロセスが有する複数の特徴のうち、それぞれ異なる少なくとも1つの特徴をランダムに抽出して、複数のプロセスを複数のプロセスタイプに分類する分類パターンを表すモデリングパターンを複数生成する。パターン生成部22は、複数のモデリングパターン b_1, b_2, \dots, b_n を生成したとする。なお、 m と n は、同じ整数であってもよいし、異なる整数であってもよい。パターン生成部22は、生成した複数のモデリングパターン b_1, b_2, \dots, b_n をモデリングパターン保存領域242に記憶する。

20

【0075】

次に、パターン生成部22は、複数の学習モデルを生成する(ステップS3)。具体的には、パターン生成部22は、学習ケース a_1, a_2, \dots, a_m の各々に、モデリングパターン b_1, b_2, \dots, b_n を適用して、複数の学習ケースを生成する。パターン生成部22は、 $c_{11} = a_1 b_1, c_{12} = a_1 b_2, \dots, c_{1n} = a_1 b_n, c_{21} = a_2 b_1, c_{22} = a_2 b_2, \dots, c_{2n} = a_2 b_n, \dots, c_{m1} = a_m b_1, c_{m2} = a_m b_2, c_{mn} = a_m b_n$ を生成したとする。パターン生成部22は、生成した学習モデル $c_{11}, c_{12}, \dots, c_{1n}, c_{21}, c_{22}, \dots, c_{2n}, \dots, c_{m1}, c_{m2}, c_{mn}$ を学習モデル保存領域243に記憶する。

30

【0076】

次に、変数生成部23は、複数の重み変数セットを生成する(ステップS4)。具体的には、変数生成部23は、複数の学習モデルのうち、最も多いプロセスタイプ数の学習モデルを特定する。変数生成部23は、特定したプロセスタイプ数分(最も多いプロセスタイプ数の学習モデルのプロセスタイプ数分)の重み変数の集合を1つの重み変数セットとして、複数の重み変数セットをランダムに生成する。変数生成部23は、 d_1, d_2, \dots, d_o を生成したとする。なお、各重み変数セットは、 $d_1 = (d_{11}, d_{12}, \dots, d_{1q}), d_2 = (d_{21}, d_{22}, \dots, d_{2q}), \dots, d_o = (d_{o1}, d_{o2}, \dots, d_{oq})$ となっている。変数生成部23は、生成した複数の重み変数セットを重み変数セット保存領域244に記憶する。

40

【0077】

次に、最適重み変数セットと最適モデリングパターンとを決定する(ステップS5)。詳細は後述するが、ステップS5は、具体的には、図6に示す動作例と、図7に示す動作例と、を含む動作となっている。また、ステップS5は、図6の動作例を実施した後に、図7の動作例を実施することにより構成される。具体的には、図6の動作例を実施することにより、ステップS2において生成した各モデリングパターンに対する最適重み変数セットを決定する。そして、図7の動作例を実施することにより、最適モデリングパターンと最適重み変数セットとを決定する。

50

【 0 0 7 8 】

次に、送信部 2 8 は、決定部 2 7 が決定した最適重み変数セットと最適モデリングパターンとをプロセス実行装置 3 0 に送信する（ステップ S 6）。

【 0 0 7 9 】

続いて、図 6 を用いて、最適重み変数セットの決定動作について説明する。図 6 は、最適重み変数セットの決定動作について説明するフローチャートである。上述したように、図 6 に示すフローチャートは、図 5 におけるステップ S 5 において実施される動作例である。

【 0 0 8 0 】

図 6 に示す様に、ステップ S 1 1 ~ S 1 8 は、モデリングパターン数分繰り返し行われる。ステップ S 1 2 ~ S 1 5 は、モデリングパターン数分かつ重み変数セット数分、繰り返し行われる。ステップ S 1 2 ~ S 1 4 は、モデリングパターン数分かつ重み変数セット数分かつ学習ケース数分、繰り返し行われる。

10

【 0 0 8 1 】

まず、算出部 2 5 は、複数の学習モデルのうち、モデリングパターン s により生成された学習モデルを抽出する（ステップ S 1 1）。ここで、抽出された学習モデルは c_{1s} 、 c_{2s} 、 \dots 、 c_{ms} である。

【 0 0 8 2 】

次に、算出部 2 5 は、抽出した学習モデル c_{1s} 、 c_{2s} 、 \dots 、 c_{ms} のそれぞれに対して、プロセスタイプ毎の優先度を算出する（ステップ S 1 2）。算出部 2 5 は、抽出した学習モデル c_{1s} 、 c_{2s} 、 \dots 、 c_{ms} のそれぞれに対して、プロセスタイプ毎の数を求め、プロセスタイプ毎の数と、重み変数セットとを、上述した数式（1）に代入することによりプロセスタイプ毎の優先度を算出する。

20

【 0 0 8 3 】

次に、評価部 2 6 は、ステップ S 1 2 において算出されたプロセスタイプ毎の優先度を用いて、処理対象の学習モデルを実行し（ステップ S 1 3）、処理対象の学習モデルの性能項目の評価値を決定する（ステップ S 1 4）。評価部 2 6 は、プロセス実行時の実行開始から全プロセス（全プロセスタイプ）の終了までの時間を計測し、その計測時間に応じて、スループットによる評価値を決定する。

【 0 0 8 4 】

次に、評価部 2 6 は、抽出した学習モデルに対して評価値が決定されると、各評価値の総和を、重み変数セット t に対する評価値と決定する（ステップ S 1 5）。

30

【 0 0 8 5 】

ここで、上記のステップ S 1 2 ~ S 1 5 の処理内容について、具体例を用いて説明する。例えば、学習ケース a_1 、 a_2 が存在し、プロセスをプロセスタイプ $TYPE_1$ と $TYPE_2$ との 2 つに分類するモデリングパターン b_1 が存在すると仮定する。そうすると、学習モデル $c_{11} = a_1 b_1$ と $c_{21} = a_2 b_1$ とが生成される。学習モデル c_{11} は、プロセスタイプ $TYPE_1$ が m_{11} 個、 $TYPE_2$ が m_{12} 個を含んでいたとする。学習モデル c_{21} は、プロセスタイプ $TYPE_1$ が m_{21} 個、 $TYPE_2$ が m_{22} 個を含んでいたとする。さらに、重み変数セット $d_1 = (d_{11}, d_{12})$ 、 $d_2 = (d_{21}, d_{22})$ が生成されているとする。

40

【 0 0 8 6 】

重み変数セット d_1 の評価値は、上記数式（1）に、 m_{11} と m_{12} と、 $d_1 = (d_{11}, d_{12})$ と、を代入し、プロセスタイプ $TYPE_1$ と $TYPE_2$ との優先度を算出する。算出した優先度に基づいて、学習モデル $c_{11} = a_1 b_1$ を実行し、プロセス実行時の実行開始から全プロセス（全プロセスタイプ）の終了までの時間を計測する。そして、その計測時間に応じて、スループットによる評価値を決定する。同様に、 $c_{21} = a_2 b_1$ に対しても評価値を決定する。算出された c_{11} の評価値と c_{21} の評価値との和が重み変数セット d_1 に対する評価値として決定される。重み変数セット d_2 に対しても上記と同様のことを行い、算出された c_{11} の評価値と c_{21} の評価値との和が重み変数セット d_2 に対する評価値として決定される。

50

【 0 0 8 7 】

全ての重み変数セットに対して評価値が決定されると、終了条件となったかを判定する（ステップ S 1 6）。終了条件は、学習を行う全体の経過時間が所定の閾値を超過した場合であるとする。つまり、図 6 のステップ S 1 1 からの処理の経過時間が所定の閾値を超過した場合（ステップ S 1 6 の Y E S）、決定部 2 7 は、決定した評価値の中から最も高い評価値に対応する重み変数セットを最適重み変数セットと決定する（ステップ S 1 7）。

【 0 0 8 8 】

一方、図 6 のステップ S 1 1 からの処理の経過時間が所定の閾値を超過していない場合（ステップ S 1 6 の N O）、変数生成部 2 3 は、新たな複数の重み変数セットを生成する（ステップ S 1 8）。

10

【 0 0 8 9 】

具体的には、変数生成部 2 3 は、遺伝的アルゴリズムを用いて、新たな複数の重み変数セットを生成する。この際、遺伝的アルゴリズムにおける、初期集団は図 5 のステップ S 4 において生成した複数の重み変数セットとなる。また、個体を重み変数セットとし、ステップ S 1 5 において決定された評価値を適応度とする。終了条件は、学習を行う全体の経過時間が所定の閾値を超過したこととする。つまり、終了条件を満たした場合は、図 6 のステップ S 1 1 開始からの経過時間が所定の閾値を超過した場合である。

【 0 0 9 0 】

選択は、例えば、複数の重み変数セットのうち、評価値が高い方から一定量選び、次の世代に残すことにより行う。交叉は、選択によって残った重み変数セットのうち、2 つを選択して、各重み変数セットの平均を取ることにより行う。例えば、重み変数セットが $d_1 = (d_{11}, d_{12})$ 、 $d_2 = (d_{21}, d_{22})$ である場合、交叉により、 $d_3 = ((d_{11} + d_{21}) / 2, (d_{12} + d_{22}) / 2)$ が生成される。

20

【 0 0 9 1 】

突然変異は、図 5 のステップ S 4 において生成した重み変数セットのように、完全にランダムな重み変数セットを低確率で生成することにより行う。なお、選択、交叉および突然変異については、まず、選択を行い、その後に、交叉および突然変異を行うようにする。以上のような条件である遺伝的アルゴリズムを適用して、新たな複数の重み変数セットを生成する。

【 0 0 9 2 】

ステップ S 1 8 において、新たな複数の重み変数セットを生成すると、ステップ S 1 1 以降の処理を、終了条件（所定の条件）を満たすまで繰り返し行う。終了条件を満たした場合、決定部 2 7 は、決定した評価値の中から最も高い評価値に対応する重み変数セットを最適重み変数セットと決定する（ステップ S 1 7）。

30

【 0 0 9 3 】

続いて、図 7 を用いて、最適モデリングパターンおよび最適重み変数セットの決定動作について説明する。図 7 は、最適モデリングパターンおよび最適重み変数セットの決定動作について説明するフローチャートである。上述したように、図 7 に示す動作例は、図 5 のステップ S 5 において行われる動作であって、図 6 に示した動作の後に行われる。なお、前提として、図 6 が実行されると、各モデリングパターンに対して、最適重み変数セットが決定されている。

40

【 0 0 9 4 】

図 7 に示す様に、ステップ S 2 1 ~ S 2 5 は、モデリングパターン数分繰り返し行われる。ステップ S 2 2 ~ S 2 4 は、モデリングパターン数分かつ学習ケース分、繰り返し行われる。

【 0 0 9 5 】

まず、算出部 2 5 は、複数の学習モデルのうち、モデリングパターン v により生成された学習モデルを抽出する（ステップ S 2 1）。ここで、抽出された学習モデルは $c_{1v}, c_{2v}, \dots, c_{mv}$ である。

【 0 0 9 6 】

50

次に、算出部 25 は、抽出した学習モデル c_{1v} , c_{2v} , ..., c_{mv} のそれぞれに対して、プロセスタイプ毎の優先度を算出する (ステップ S 22)。算出部 25 は、抽出した学習モデル c_{1v} , c_{2v} , ..., c_{mv} のそれぞれに対して、プロセスタイプ毎の数を求め、求めたプロセスタイプ毎の数と、重み変数セットとを、上述した数式 (1) に代入してプロセスタイプ毎の優先度を算出する。

【0097】

次に、評価部 26 は、ステップ S 22 において算出されたプロセスタイプ毎の優先度を用いて、処理対象の学習モデルを実行し (ステップ S 23)、処理対象の学習モデルの性能項目の評価値を決定する (ステップ S 24)。評価値の決定については、図 6 のステップ S 14 と同様とする。

10

【0098】

次に、評価部 26 は、抽出した学習モデルに対して評価値が決定されると、各評価値の総和を、モデリングパターン v に対する評価値と決定する (ステップ S 25)。全てのモデリングパターンに対して評価値が決定されると、終了条件となったかを判定する (ステップ S 26)。終了条件は、学習を行う全体の経過時間が所定の閾値を超過した場合であるとする。つまり、図 7 のステップ S 21 からの処理の経過時間が所定の閾値を超過した場合 (ステップ S 26 の YES)、決定部 27 は、決定した評価値の中から最も高い評価値に対応するモデリングパターンおよび重み変数セットを決定する。決定部 27 は、決定したモデリングパターンおよび重み変数セットを、最適モデリングパターンおよび最適重み変数セットと決定する (ステップ S 27)。

20

【0099】

一方、図 7 のステップ S 21 からの処理の経過時間が所定の閾値を超過していない場合 (ステップ S 26 の NO)、パターン生成部 22 は、新たな複数のモデリングパターンを生成する (ステップ S 28)。

【0100】

具体的には、パターン生成部 22 は、遺伝的アルゴリズムを用いて、新たな複数のモデリングパターンを生成する。この際、遺伝的アルゴリズムにおける、初期集団は図 5 のステップ S 2 において生成した複数のモデリングパターンとなる。また、個体をモデリングパターンとし、ステップ S 25 において決定された評価値を適応度とする。終了条件は、学習を行う全体の経過時間が所定の閾値を超過したこととする。つまり、終了条件を満たした場合とは、図 7 のステップ S 21 開始からの経過時間が所定の閾値を超過した場合である。

30

【0101】

選択は、例えば、複数のモデリングパターンの評価値が高い方から一定量選び、次の世代に残すことにより行う。交叉は、選択によって残った重み変数セットのうち、2つを選択して、各重み変数セットの平均を取ることにより行う。例えば、モデリングパターン b_1 と b_2 とが、プロセスの特徴 y と z を元に、3種類のプロセスをプロセスタイプに分類するモデリングであると仮定する。また、モデリングパターン b_1 と b_2 の関数が以下のようになっていたとする。

【数 2】

40

$$\text{モデリンパターン } b_1 = \begin{cases} \text{Process } x = \text{TypeA} & (y + 3z \geq 4) \\ \text{Process } x = \text{TypeB} & (y + 3z \geq 1) \\ \text{Process } x = \text{TypeC} & (y + 3z < 1) \end{cases}$$

$$\text{モデリンパターン } b_2 = \begin{cases} \text{Process } x = \text{TypeA} & (3y + 5z \geq 8) \\ \text{Process } x = \text{TypeB} & (3y + 5z \geq 5) \\ \text{Process } x = \text{TypeC} & (3y + 5z < 5) \end{cases}$$

10

この場合、交叉によって新たに生成されるモデリングパターンを b_3 とすると、以下のよう関数が生成される。

【数 3】

$$\text{モデリンパターン } b_3 = \begin{cases} \text{Process } x = \text{TypeA} & \left(\left(\frac{1+3}{2} \right) y + \left(\frac{3+5}{2} \right) z \geq \left(\frac{4+8}{2} \right) \right) \\ \text{Process } x = \text{TypeB} & \left(\left(\frac{1+3}{2} \right) y + \left(\frac{3+5}{2} \right) z \geq \left(\frac{1+5}{2} \right) \right) \\ \text{Process } x = \text{TypeC} & \left(\left(\frac{1+3}{2} \right) y + \left(\frac{3+5}{2} \right) z \geq \left(\frac{1+5}{2} \right) \right) \end{cases}$$

20

【0102】

突然変異は、図5のステップS2において生成したモデリングパターンのように、ランダムなモデリングパターンを低確率で生成することにより行う。なお、選択、交叉および突然変異については、まず、選択を行い、その後に、交叉および突然変異を行うようにする。以上のような条件である遺伝的アルゴリズムを適用して、新たな複数のモデリングパターンを生成する。

30

【0103】

ステップS28において、新たな複数のモデリングパターンを生成すると、新たな複数のモデリングパターンを学習ケース a_1, a_2, \dots, a_m に適用して、新たな複数の学習モデルを生成する(ステップS29)。ステップS29で行うことは、図5のステップS3と同じである。

【0104】

次に、新たな複数の学習モデルに対して最適重み変数セットを決定する(ステップS30)。新たな学習モデルが生成された場合は、当然ながら既に存在する重み変数セットのいずれかが、新たな学習モデルに対する最適重み変数セットになるとは限らない。そのため、ステップS30において、新たな学習モデルに対して、最適重み変数セットを決定する。ステップS30において実施される内容は、図6に示した動作と同じである。つまり、ステップS30においても、図6が実施され、最適重み変数セットを決定する。なお、図6においては、全ての学習モデルに対して実施されるが、ステップS30においては、新たなモデリングパターンにより生成された新たな学習モデルに対してのみ、図6の動作が実施される。

40

【0105】

以降、ステップS21以降の処理を、終了条件(所定の条件)を満たすまで繰り返し行う。終了条件を満たした場合、決定部27は、決定した評価値の中から最も高い評価値に対応するモデリングパターンおよび重み変数セットを決定する。決定部27は、決定したモ

50

デリングパターンおよび重み変数セットを、最適モデリングパターンおよび最適重み変数セットと決定する（ステップS27）。

【0106】

<プロセス実行装置の動作例>

続いて、図8を用いて、プロセス実行装置30の動作例について説明する。図8は、実施の形態2にかかるプロセス実行装置の動作例を説明するフローチャートである。

【0107】

まず、受信部31は、最適モデリングパターンおよび最適重み変数セットをプロセススケジューリング装置20の送信部28から受信する（ステップS31）。受信部31は、最適モデリングパターンおよび最適重み変数セットを、それぞれモデリングパターン保存領域321および重み変数セット保存領域322に記憶する。

10

【0108】

次に、プロセス実行部33は、プロセス実行装置30にインストールされているアプリケーションプログラムが生成した実プロセスを、最適モデリングパターンを用いてプロセスタイプに分類する（ステップS32）。

【0109】

次に、プロセス実行部33は、最適重み変数セットを用いて、プロセスタイプ毎の優先度を算出する（ステップS33）。プロセス実行部33は、ステップS32において分類されたプロセスタイプ毎の数を求め、求めたプロセスタイプ毎の数と最適重み変数セットとを、上述した数式(1)に代入して、プロセスタイプ毎の優先度を算出する。

20

【0110】

次に、プロセス実行部33は、算出したプロセスタイプ毎の優先度に基づいて、実プロセスを実行する（ステップS33）。プロセス実行部33は、複数の実プロセスを最適モデリングパターンに基づいて、複数のプロセスタイプに分類している。また、プロセス実行部33は、プロセスタイプ毎の優先度を算出しているため、分類したプロセスタイプを、プロセスタイプ毎の優先度が高いプロセスタイプから実行する。同じ優先度のプロセスタイプが複数存在する場合は、プロセス実行部33は、各プロセスに既に割り振ったリソースの量が少ないプロセスを先に実行する。優先度が等しく、かつ、既に割り振ったリソースの量も等しい場合は、プロセス実行部33は、先に生成されたプロセスから順に実行する。

30

【0111】

続いて、実施の形態2の効果について説明する。1つ目は、プロセスが有する複数の特徴を用いたプロセススケジューリングが可能となる。実施の形態2においても、パターン生成部22が、プロセスが有する複数の特徴を用いて、複数のモデリングパターンを生成する。評価部26は、複数のモデリングパターンの各々に対して、評価値を決定し、決定部27は、決定された評価値に基づいて、最適な分類パターンを決定する。したがって、実施の形態2にかかるプロセススケジューリング装置10によれば、プロセスが有する複数の特徴を用いたプロセススケジューリングが可能となる。

【0112】

2つ目は、プロセスが有する特徴を用いたモデリングを行うことにより、プロセスが有する特徴を入力とした機械学習を行う場合と比較して情報量を削減することが可能となる。最適なプロセススケジューリングとなる解を導くために、機械学習を用いることが考えられる。機械学習を行う際に、プロセスが有する複数の特徴を、そのまま入力値とすることも想定される。しかしながら、この場合、プロセスの実行優先度は絶対的なものではなく、他に存在するプロセスにより決定される相対的なものとなる。つまり、全てのプロセスに関する情報も機械学習の入力情報とする必要が生じてしまう。しかし、各プロセスの特徴だけでなく、他のプロセスの特徴も考慮した機械学習を行う場合、全てのプロセスの特徴情報を保持しなければならないため、情報量が膨大になるという課題が生じてしまう。そこで、本実施の形態では、複数の特徴を用いて、プロセスをプロセスタイプに分類することによって、適切に情報量を削減することが可能となる。また、このようにすることで

40

50

、容易に、かつ有用な学習結果を得ることが出来る。つまり、最適なモデリングパターンおよび最適な重み変数セットを容易に求めることが出来、最適なプロセススケジューリングを行うことが可能となる。

【0113】

3つ目は、プロセスが有する複数の特徴のうち、最適な特徴を決定して、最適なプロセススケジューリングとすることが可能となる。最適なプロセススケジューリングとなるような特徴を決定する場合、人間が1つ1つの特徴に対して評価を行うことも想定される。しかしながら、例えば、ある特徴は、プロセススケジューリングに関係が無いとして評価を行わないと言う場合も考えられる。本実施の形態にかかるプロセススケジューリング装置20は、プロセスが有する複数の特徴を用いて、最適なモデリングパターンと最適な重み変数セットとを決定する。そのため、プロセスが有する複数の特徴を反映したプロセススケジューリングを可能とする。つまり、人為的に発見することの難しい特徴を用いたプロセススケジューリングを行うことが可能となる。したがって、本実施の形態にかかるプロセススケジューリング装置20によれば、人為的に発見することの難しい特徴を用いた最適なプロセススケジューリングとなる関数を発見できるという利点がある。

10

【0114】

4つ目は、機械学習に適したプロセスのモデリングパターンを導出する手法を確立できる。関連する技術では、機械学習に適したプロセスをプロセスタイプにモデリングすることは検討されていない。また、複数のモデリングパターンを生成することも検討されていない。仮に、人間が機械学習に適したプロセスのモデリングパターンを作成することも想定されるが、人間が作成するのは容易ではない。一方、上述した本実施の形態によれば、複数のモデリングパターンによりモデル化された学習モデルを学習することより、任意の機械学習手法に適したモデリングパターンを導出することが可能となる。これにより人間がモデリングパターンを作成する必要がなくなるという効果を有する。

20

【0115】

5つ目は、最適な重み変数セットを導出することが可能となるので、より最適なプロセススケジューリングを可能とする。本実施の形態によれば、最適なモデリングパターンだけでなく、最適なモデリングパターンにより分類された各プロセスタイプに対して最適な重み付けを行うことが可能となる。つまり、本実施の形態によれば、最適なモデリングパターンと、最適な重み変数セットと、によって、最適なバランスとなるプロセススケジューリングとすることが可能となる。上述したように、人間が最適なモデリングパターンを決定することは容易ではないが、仮に、人間が最適なモデリングパターンを作成出来たとしても、各プロセスタイプを最適に重み付けする変数を決定することも容易ではない。本実施の形態によれば、最適なモデリングパターンだけでなく、最適なモデリングパターンにより分類された各プロセスタイプに対して最適な重み付けを行うことが可能となる。

30

【0116】

(変形例)

上述した実施の形態において、以下のような変形を施してもよい。

<1> 上述した実施の形態では、プロセススケジューリング装置20において決定した最適モデリングパターンおよび最適重み変数セットを用いて、プロセス実行装置30が実プロセスを実行する構成として説明した。例えば、上述した実施の形態2を変形して、プロセススケジューリング装置20が最適モデリングパターンおよび最適重み変数セットを用いて、実プロセスを実行するようにしてもよい。

40

【0117】

図9は、実施の形態2の変形例にかかるプロセススケジューリング装置20の構成例を示すブロック図である。この場合、図9に示す様に、プロセススケジューリング装置20において、実施の形態2における送信部28を設けなくてよい。また、プロセススケジューリング装置20において、プロセス実行部29を設け、実施の形態2におけるプロセス実行部33と同様の構成とすればよい。さらに、モデリングパターン保存領域242および重み変数セット保存領域244において、最適モデリングパターンおよび最適重み変数セ

50

ットが特定出来るように変更すればよい。このようにすることで、プロセス実行部 29 は、最適モデリングパターンおよび最適重み変数セットを特定することができ、実施の形態 2 と同様の効果を得ることが可能となる。

【0118】

< 2 > 上述した実施の形態 2 において、プロセス実行装置 30 は、プロセススケジューリング装置 20 から決定した最適モデリングパターンと最適重み変数セットとを受信した。例えば、プロセス実行装置 30 の記憶部 32 に、予め最適モデリングパターンと最適重み変数セットとを記憶しておいてもよい。つまり、受信部 31 を設けない構成であってもよい。図 10 は、実施の形態 2 の変形例にかかるプロセス実行装置の構成例を示すブロック図である。図 10 に示す様に、プロセス実行装置 30 は、記憶部 32 と、プロセス実行部 33 とを備える構成であってもよい。

10

【0119】

(他の実施の形態)

図 11 は、上述した実施の形態において説明したプロセススケジューリング装置 10、20 (以下、プロセススケジューリング装置 10 等と称する) およびプロセス実行装置 30 の構成例を示すブロック図である。図 11 を参照すると、プロセススケジューリング装置 10 等およびプロセス実行装置 30 は、ネットワーク・インターフェース 1201、プロセッサ 1202、及びメモリ 1203 を含む。ネットワーク・インターフェース 1201 は、通信システムを構成する他のネットワークノード装置と通信するために使用される。ネットワーク・インターフェース 1201 は、例えば、IEEE 802.3 series に準拠したネットワークインターフェースカード (NIC) を含んでもよい。

20

【0120】

プロセッサ 1202 は、メモリ 1203 からソフトウェア (コンピュータプログラム) を読み出して実行することで、上述の実施形態においてフローチャートを用いて説明されたプロセススケジューリング装置 10 等およびプロセス実行装置 30 の処理を行う。プロセッサ 1202 は、例えば、マイクロプロセッサ、MPU (Micro Processing Unit)、又は CPU (Central Processing Unit) であってもよい。プロセッサ 1202 は、複数のプロセッサを含んでもよい。

【0121】

メモリ 1203 は、揮発性メモリ及び不揮発性メモリの組み合わせによって構成される。メモリ 1203 は、プロセッサ 1202 から離れて配置されたストレージを含んでもよい。この場合、プロセッサ 1202 は、図示されていない I/O インターフェースを介してメモリ 1203 にアクセスしてもよい。

30

【0122】

図 11 の例では、メモリ 1203 は、ソフトウェアモジュール群を格納するために使用される。プロセッサ 1202 は、これらのソフトウェアモジュール群をメモリ 1203 から読み出して実行することで、上述の実施形態において説明されたプロセススケジューリング装置 10 等およびプロセス実行装置 30 の処理を行うことができる。

【0123】

図 11 を用いて説明したように、プロセススケジューリング装置 10 等およびプロセス実行装置 30 が有するプロセッサの各々は、図面を用いて説明されたアルゴリズムをコンピュータに行わせるための命令群を含む 1 または複数のプログラムを実行する。

40

【0124】

上述の例において、プログラムは、様々なタイプの非一時的なコンピュータ可読媒体 (non-transitory computer readable medium) を用いて格納され、コンピュータに供給することができる。非一時的なコンピュータ可読媒体は、様々なタイプの実体のある記録媒体 (tangible storage medium) を含む。非一時的なコンピュータ可読媒体の例は、磁気記録媒体 (例えばフレキシブルディスク、磁気テープ、ハードディスクドライブ)、光磁気記録媒体 (例えば光磁気ディスク) を含む。さらに、非一時的なコンピュータ可読媒体の例は、CD-ROM (Read Only Memory)、CD-R、CD-R/W を含む。さ

50

らに、非一時的なコンピュータ可読媒体の例は、半導体メモリを含む。半導体メモリは、例えば、マスクROM、PROM (Programmable ROM)、EPROM (Erasable PROM)、フラッシュROM、RAM (Random Access Memory)を含む。また、プログラムは、様々なタイプの一時的なコンピュータ可読媒体 (transitory computer readable medium) によってコンピュータに供給されてもよい。一時的なコンピュータ可読媒体の例は、電気信号、光信号、及び電磁波を含む。一時的なコンピュータ可読媒体は、電線及び光ファイバ等の有線通信路、又は無線通信路を介して、プログラムをコンピュータに供給できる。

【0125】

なお、本開示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。また、本開示は、それぞれの実施の形態を適宜組み合わせ実施されてもよい。

10

【0126】

また、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。

(付記1)

プロセスが有する少なくとも1つの特徴を用いて、複数のプロセスを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも1つの特徴を用いて複数生成するパターン生成部と、

前記複数の分類パターンの各々に対して、予め定められた性能項目に対する評価値を決定する評価部と、

20

前記決定された評価値に基づいて、前記複数のプロセスに対する分類パターンを決定する決定部と、を備えるプロセススケジューリング装置。

【0127】

(付記2)

前記パターン生成部は、所定の条件を満たすまで、前記決定された評価値に基づいて新たな複数の前記分類パターンを繰り返し生成する、付記1に記載のプロセススケジューリング装置。

【0128】

(付記3)

前記分類パターンに含まれるプロセスタイプの各々に重み付けを行う重み変数の集合である重み変数セットを複数生成する変数生成部をさらに備え、

前記評価部は、前記分類パターンと前記重み変数セットとの組み合わせ毎に、前記評価値を決定し、

前記決定部は、前記決定された評価値に基づいて、前記複数のプロセスに対する分類パターンおよび重み変数セットを決定する、付記1または2に記載のプロセススケジューリング装置。

30

【0129】

(付記4)

前記変数生成部は、所定の条件を満たすまで、前記決定された評価値に基づいて新たな複数の前記重み変数セットを繰り返し生成する、付記3に記載のプロセススケジューリング装置。

40

【0130】

(付記5)

前記分類パターンと前記重み変数セットとの組み合わせ毎に、前記分類パターンに含まれるプロセスタイプ毎の数と前記重み変数セットとを用いて、プロセスタイプ毎の優先度を算出する算出部をさらに備え、

前記評価部は、前記分類パターンと前記重み変数セットとの組み合わせ毎に、前記算出されたプロセスタイプ毎の優先度に基づいて、前記複数のプロセスを実行し、前記評価値を決定する、付記3または4のいずれか1項に記載のプロセススケジューリング装置。

50

【 0 1 3 1 】

(付記 6)

前記複数のプロセスを表すプロセス群を複数生成するプロセス生成部をさらに備え、前記パターン生成部は、前記複数のプロセス群の各々に対して、前記複数の分類パターンを適用し、

前記変数生成部は、前記プロセス群と前記分類パターンとの組み合わせ毎に、前記複数の重み変数セットを生成し、

前記評価部は、前記プロセス群と前記分類パターンと前記重み変数セットとの組み合わせ毎に、前記評価値を決定し、各プロセス群の前記評価値の総和を、前記分類パターンと前記重み変数セットとの組み合わせに対する評価値と決定し、

前記決定部は、前記決定された評価値に基づいて、前記複数のプロセス群に対する分類パターンおよび重み変数セットを決定する、付記 3 ~ 5 のいずれか 1 項に記載のプロセススケジューリング装置。

10

【 0 1 3 2 】

(付記 7)

前記パターン生成部は、遺伝的アルゴリズムを適用して、新たな複数の前記分類パターンを生成する、付記 2 に記載のプロセススケジューリング装置。

【 0 1 3 3 】

(付記 8)

前記変数生成部は、遺伝的アルゴリズムを適用して、新たな複数の前記重み変数セットを生成する、付記 4 に記載のプロセススケジューリング装置。

20

【 0 1 3 4 】

(付記 9)

前記性能項目は、スループット、消費電力および割り当てられたリソースの平等性の少なくとも 1 つを含む、付記 1 ~ 8 のいずれか 1 項に記載のプロセススケジューリング装置。

【 0 1 3 5 】

(付記 10)

前記プロセスが有する少なくとも 1 つの特徴は、フォアグラウンドプロセスまたはバックグラウンドプロセスであること、プロセス実行順、プロセス発生順、過去に割り当てられたプロセスの CPU 時間、プロセス名、ユーザ情報およびプロセスに入力された入力情報のうち、少なくとも 1 つを含む、付記 1 ~ 9 のいずれか 1 項に記載のプロセススケジューリング装置。

30

【 0 1 3 6 】

(付記 11)

前記決定部が決定した分類パターンおよび重み変数セットの少なくとも 1 つに基づいて、アプリケーションプログラムが生成する複数のプロセスを実行するプロセス実行部をさらに備える、付記 1 ~ 10 のいずれか 1 項に記載のプロセススケジューリング装置。

【 0 1 3 7 】

(付記 12)

プロセスが有する少なくとも 1 つの特徴を用いて、複数の第 1 のプロセスを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも 1 つの特徴を用いて複数生成し、前記複数の分類パターンの各々に対して、予め定められた性能項目に対する評価値を決定し、前記決定された評価値に基づいて決定された分類パターンを用いて、アプリケーションプログラムが生成する複数の第 2 のプロセスを実行するプロセス実行部を備える、プロセス実行装置。

40

【 0 1 3 8 】

(付記 13)

前記プロセス実行部は、前記決定された分類パターンを用いて、前記複数の第 2 のプロセスを複数のプロセスタイプに分類し、前記分類されたプロセスタイプ毎の優先度を算出し、前記算出されたプロセスタイプ毎の優先度に基づいて、前記複数の第 2 のプロセスを実

50

行する、付記 1 2 に記載のプロセス実行装置。

【 0 1 3 9 】

(付記 1 4)

プロセスが有する少なくとも 1 つの特徴を用いて、複数のプロセスを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも 1 つの特徴を用いて複数生成することと、

前記複数の分類パターンの各々に対して、予め定められた性能項目に対する評価値を決定することと、

前記決定された評価値に基づいて、前記複数のプロセスに対する分類パターンを決定することと、を含むプロセススケジューリング方法。

10

【 0 1 4 0 】

(付記 1 5)

プロセスが有する少なくとも 1 つの特徴を用いて、複数のプロセスを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも 1 つの特徴を用いて複数生成することと、

前記複数の分類パターンの各々に対して、予め定められた性能項目に対する評価値を決定することと、

前記決定された評価値に基づいて、前記複数のプロセスに対する分類パターンを決定することと、をコンピュータに実行させるプログラム。

【 0 1 4 1 】

(付記 1 6)

前記分類パターンに含まれるプロセスタイプの各々に重み付けを行う重み変数の集合である重み変数セットを複数生成することをさらに含み、

前記分類パターンと前記重み変数セットとの組み合わせ毎に、前記評価値を決定し、

前記決定された評価値に基づいて、前記複数のプロセスに対する分類パターンおよび重み変数セットを決定する、付記 1 5 に記載のプログラム。

20

【 0 1 4 2 】

(付記 1 7)

プロセス実行時のプロセス優先度を決定するプロセススケジューリング装置と、

アプリケーションプログラムが生成した実プロセスを実行するプロセス実行装置と、を備え、

30

前記プロセススケジューリング装置は、

プロセスが有する少なくとも 1 つの特徴を用いて、複数のプロセスを複数のプロセスタイプに分類する分類パターンを、それぞれ異なる少なくとも 1 つの特徴を用いて複数生成するパターン生成部と、

前記複数の分類パターンの各々に対して、予め定められた性能項目に対する評価値を決定する評価部と、

前記決定された評価値に基づいて、前記複数のプロセスに対する分類パターンを決定する決定部と、

前記決定された分類パターンを前記プロセス実行装置に送信する送信部と、を備え、

40

前記プロセス実行装置は、

前記決定された分類パターンを前記プロセススケジューリング装置から受信する受信部と、

前記受信した分類パターンに基づいて、アプリケーションプログラムが生成する複数のプロセスを実行するプロセス実行部と、を備える、プロセス実行システム。

【 0 1 4 3 】

(付記 1 8)

前記プロセススケジューリング装置は、

前記分類パターンに含まれるプロセスタイプの各々に重み付けを行う重み変数の集合である重み変数セットを複数生成する変数生成部をさらに備え、

前記評価部は、前記分類パターンと前記重み変数セットとの組み合わせ毎に、前記評価値

50

を決定し、

前記決定部は、前記決定された評価値に基づいて、前記複数のプロセスに対する分類パターンおよび重み変数セットを決定し、

前記送信部は、前記決定された分類パターンおよび重み変数セットを送信し、

前記受信部は、前記決定された分類パターンおよび重み変数セットを受信し、

前記プロセス実行部は、前記受信した分類パターンおよび重み変数セットに基づいて、前記アプリケーションプログラムが生成する複数のプロセスを実行する、付記 17 に記載のプロセス実行システム。

【符号の説明】

【 0 1 4 4 】

10、20 プロセススケジューリング装置

11、22 パターン生成部

12、26 評価部

13、27 決定部

21 プロセス生成部

23 変数生成部

24、32 記憶部

25 算出部

28 送信部

29、33 プロセス実行部

30 プロセス実行装置

31 受信部

33 プロセス実行部

100 プロセス実行システム

10

20

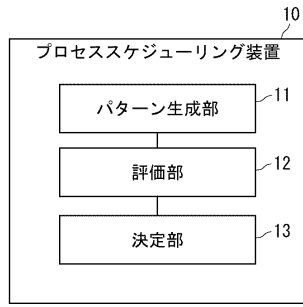
30

40

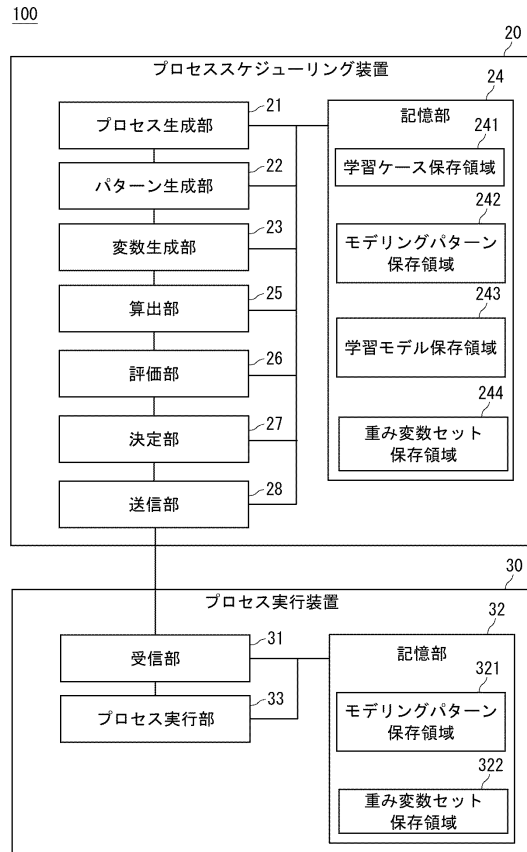
50

【図面】

【図1】



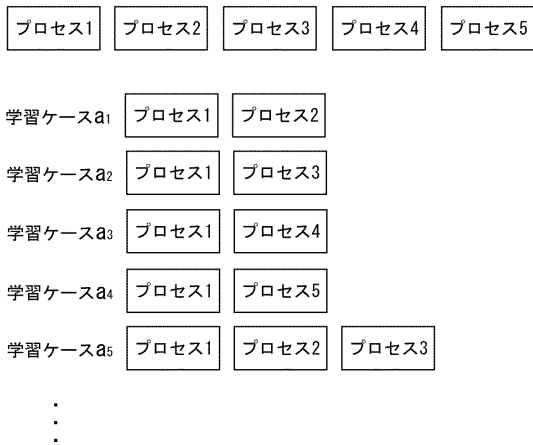
【図2】



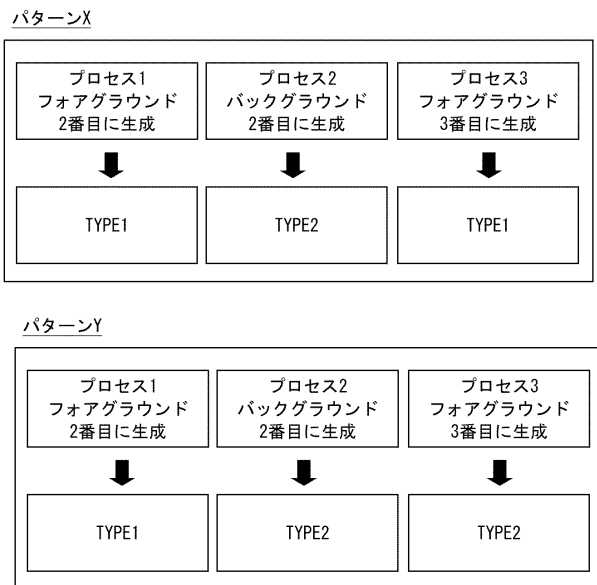
10

20

【図3】



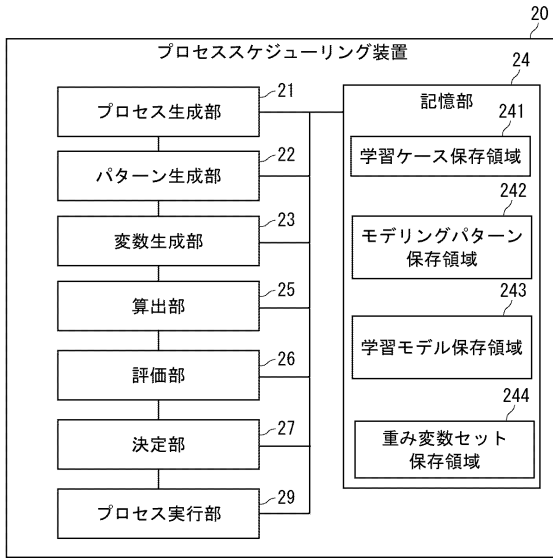
【図4】



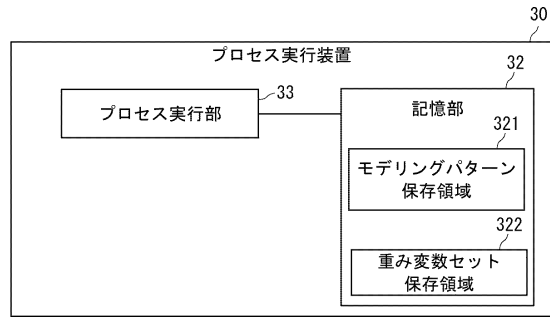
30

40

【図 9】

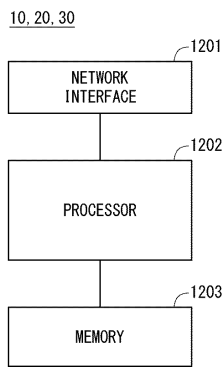


【図 10】



10

【図 11】



20

30

40

50

フロントページの続き

- (56)参考文献 国際公開第2010/055719(WO, A1)
特表2012-517041(JP, A)
特開2010-257056(JP, A)
特開2016-139171(JP, A)
特開2009-020692(JP, A)
特開2004-326486(JP, A)
- (58)調査した分野 (Int.Cl., DB名)
G06F 9/48