

(19)日本国特許庁(JP)

**(12)公開特許公報(A)**(11)公開番号  
特開2024-70362  
(P2024-70362A)

(43)公開日 令和6年5月23日(2024.5.23)

(51)国際特許分類

G 0 6 F 9/50 (2006.01)

F I

G 0 6 F

9/50

1 5 0 Z

審査請求 未請求 請求項の数 18 O L (全29頁)

(21)出願番号 特願2022-180798(P2022-180798)  
 (22)出願日 令和4年11月11日(2022.11.11)

(71)出願人 515130201  
 株式会社 Preferred Networks  
 東京都千代田区大手町1丁目6番1号  
 大手町ビル  
 100120031  
 弁理士 宮嶋 学  
 100118876  
 弁理士 鈴木 順生  
 100202429  
 弁理士 石原 信人  
 川口 順央  
 東京都千代田区大手町1丁目6番1号  
 大手町ビル 株式会社 Preferred Networks 内

最終頁に続く

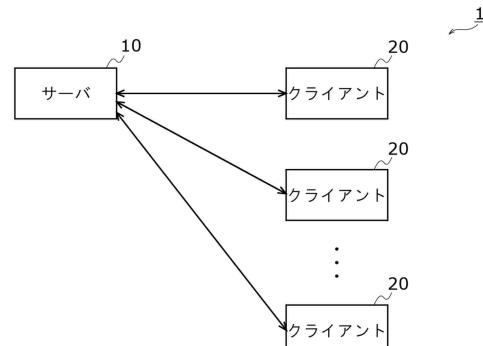
(54)【発明の名称】 情報処理装置システム

**(57)【要約】**

**【課題】**入力データに対するリソースの割り当てを最適化する。

**【解決手段】**情報処理システムは、少なくとも1つのメモリと、少なくとも1つのプロセッサと、を備える。前記少なくとも1つのプロセッサは、ユーザの要求を受信し、前記ユーザの利用条件に基づいて、前記要求の処理可否を判定し、処理可能と判定された前記要求に基づいて、ニューラルネットワークを用いた処理を実行し、前記利用条件は、前記ニューラルネットワークの処理対象の構造に関する条件を含む。

【選択図】図1



10

**【特許請求の範囲】****【請求項 1】**

少なくとも 1 つのメモリと、

少なくとも 1 つのプロセッサと、を備え、

前記少なくとも 1 つのプロセッサは、

ユーザの要求を受信し、

前記ユーザの利用条件に基づいて、前記要求の処理可否を判定し、

処理可能と判定された前記要求に基づいて、ニューラルネットワークを用いた処理を実行し、

前記利用条件は、前記ニューラルネットワークの処理対象の構造に関する条件を含む

10

、  
情報処理システム。

**【請求項 2】**

前記利用条件は、少なくとも前記処理対象の原子数に関する条件又は前記処理対象の原子の近傍数に関する条件のいずれかを含む、

請求項 1 に記載の情報処理システム。

**【請求項 3】**

前記ユーザの利用条件における原子数は、前記ユーザより安価なプランで前記ニューラルネットワークを用いたサービスを契約している他のユーザの利用条件における原子数以上であり、

前記ユーザの利用条件における原子の近傍数は、前記他のユーザの利用条件における原子の近傍数以上である、

請求項 2 に記載の情報処理システム。

**【請求項 4】**

前記少なくとも 1 つのプロセッサは、

前記要求に基づいて、前記ニューラルネットワークに入力する情報を構築し、

構築された前記情報と前記利用条件とに基づいて、前記要求の処理可否を判定する、  
請求項 1 に記載の情報処理システム。

**【請求項 5】**

構築された前記情報はグラフである、

30

請求項 4 に記載の情報処理システム。

**【請求項 6】**

前記ニューラルネットワークは、グラフニューラルネットワークである、

請求項 1 に記載の情報処理システム。

**【請求項 7】**

前記ニューラルネットワークは、NNP(Neural Network Potential)のモデルである、

請求項 1 に記載の情報処理システム。

**【請求項 8】**

前記ユーザの利用条件は、前記ニューラルネットワークの種類に基づいて設定される、

40

請求項 1 に記載の情報処理システム。

**【請求項 9】**

前記少なくとも 1 つのプロセッサは、

前記ユーザから複数の要求を受信し、

前記複数の要求を前記少なくとも 1 つのメモリに記憶し、

構造に関する情報に基づいて、前記少なくとも 1 つのメモリに記憶された前記複数の要求のうち、前記ニューラルネットワークを用いた処理を実行する要求の量を制御する、

請求項 1 乃至請求項 8 のいずれか一項に記載の情報処理システム。

**【請求項 10】**

前記構造に関する情報は、少なくとも原子数に関する情報又は原子の近傍数に関する情

50

報のいずれかを含む、

請求項 9 に記載の情報処理システム。

【請求項 1 1】

少なくとも 1 つのメモリと、

少なくとも 1 つのプロセッサと、を備え、

前記少なくとも 1 つのプロセッサは、

ユーザから複数の要求を受信し、

前記複数の要求を前記少なくとも 1 つのメモリに記憶し、

前記少なくとも 1 つのメモリに記憶された前記複数の要求のうち、ニューラルネットワークを用いた処理を実行する要求の量を制御し、

前記要求の量は、構造に関する情報に基づいて制御される、

情報処理システム。

【請求項 1 2】

前記構造に関する情報は、少なくとも原子数に関する情報又は原子の近傍数に関する情報のいずれかを含む、

請求項 1 1 に記載の情報処理システム。

【請求項 1 3】

前記量は、所定時間に前記ユーザに追加される第 1 トークン量、前記ユーザが保有可能な第 2 トークン量又は所定時間に前記ユーザが消費可能な第 3 トークン量のいずれか一つを用いて制御される、

請求項 1 1 に記載の情報処理システム。

【請求項 1 4】

前記第 1 トークン量は、前記構造に関する情報に基づく値であり、

前記第 2 トークン量は、前記構造に関する情報に基づく値であり、

前記第 3 トークン量は、前記構造に関する情報に基づく値である、

請求項 1 3 に記載の情報処理システム。

【請求項 1 5】

前記構造に関する情報は、少なくとも原子数に関する情報又は原子の近傍数に関する情報のいずれかを含む、

請求項 1 4 に記載の情報処理システム。

【請求項 1 6】

前記ユーザに設定された前記第 1 トークン量は、前記ユーザより安価なプランで前記ニューラルネットワークを用いたサービスを契約している他のユーザに設定された第 1 トークン量以上であり、

前記ユーザに設定された前記第 2 トークン量は、前記他のユーザに設定された第 2 トークン量以上であり、

前記ユーザに設定された前記第 3 トークン量は、前記他のユーザに設定された第 3 トークン量以上である、

請求項 1 4 に記載の情報処理システム。

【請求項 1 7】

前記ニューラルネットワークは、グラフニューラルネットワークである、

請求項 1 0 に記載の情報処理システム。

【請求項 1 8】

前記グラフニューラルネットワークは、NNP(Neural Network Potential)のモデルである、

請求項 1 0 に記載の情報処理システム。

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

本開示は、情報処理装置システムに関する。

10

20

30

40

50

**【背景技術】****【0002】**

SaaS (Software as a Service) は、インターネット回線を介してクラウドサーバに実装されているソフトウェアを用いた種々のサービスを提供する手段として広く展開されてきている。SaaS の一例として、クラウド上に GPU (Graphics Processing Unit) 等のアクセラレータを含むクラスタを演算サーバとして備え、この GPU を用いた高速演算をユーザに利用させるサービスがある。

**【0003】**

このような GPU クラスタにおいて複数のクライアントからの処理を受け付けるサービスを行う場合、優先度の設定やユーザごとに入力できるデータ量によって処理を制御することで、安定し、かつ、ユーザが払う料金に応じた適切なリソースの割り振りをすることが望ましい。しかしながら、SaaS におけるサービスは多岐に亘るため、どのようなリソースの割り当てをするかは提供するサービスの内容に応じた考慮が必要である。

**【先行技術文献】****【非特許文献】****【0004】**

【非特許文献1】MTLANTIS, [インターネット], <https://matlantis.com/>, 2022年11月10日確認

**【発明の概要】****【発明が解決しようとする課題】****【0005】**

本開示の実施形態が解決しようとする限定されない課題の一つは、ニューラルネットワークを用いた演算を行う場合における適切なリソース割り当て手法の提供である。

**【課題を解決するための手段】****【0006】**

一実施形態によれば、情報処理システムは、少なくとも1つのメモリと、少なくとも1つのプロセッサと、を備える。前記少なくとも1つのプロセッサは、ユーザの要求を受信し、前記ユーザの利用条件に基づいて、前記要求の処理可否を判定し、処理可能と判定された前記要求に基づいて、ニューラルネットワークを用いた処理を実行し、前記利用条件は、前記ニューラルネットワークの処理対象の構造に関する条件を含む。

**【0007】**

一実施形態によれば、情報処理装置は、処理回路を備える。前記処理回路は、クライアントから送信される入力データにおける原子数について、ユーザごとに設定される流量に基づいた処理速度で前記入力データを処理する、処理回路。

**【図面の簡単な説明】****【0008】**

【図1】一実施形態に係る情報処理システムの一例を模式的に示す図。

【図2】一実施形態に係る情報処理システムの処理を示すフローチャート。

【図3】一実施形態に係る情報処理システムの一例を模式的に示す図。

【図4】一実施形態に係る情報処理システムの処理を示すフローチャート。

【図5】一実施形態に係る流量制御の一例を模式的に示す図。

【図6】一実施形態に係るそれぞれの情報処理装置の一例を模式的に示すブロック図。

**【発明を実施するための形態】****【0009】**

本開示の実施形態により解決しようとする課題は、上記に示したもののはか、さらに限定されないいくつかの例として、実施形態において記載した効果に対応する課題、とすることもできる。すなわち、本開示の実施形態の説明において記載された効果のうち任意の少なくとも1つに対応する課題を本開示における解決しようとする課題とすることができる。

**【0010】**

10

20

30

40

50

以下、図面を参照して本発明の実施形態について説明する。図面及び実施形態の説明は一例として示すものであり、本発明を限定するものではない。また、実施形態の説明において提示する数値は、例として示すものであり、本開示における形態を提示されている数値に限定するものではない。

#### 【0011】

##### (第1実施形態)

図1は、一実施形態に係る情報処理システムの概略の一例を模式的に示す図である。情報処理システム1は、サーバ10と、複数のクライアント20と、を備える。情報処理システム1は、クライアント20から入力された演算要求によってサーバ10において所定の演算を実行し、クライアント20へと演算結果を出力するシステムである。  
10 サーバ10とクライアント20は、例えば、インターネットといったネットワークを介して接続される。

#### 【0012】

本開示においては、限定されない一例として、情報処理システム1は、グラフを入力が可能なニューラルネットワークの処理を実行し、ユーザに出力する。より具体的な一例として、原子情報に基づくグラフを入力すると、少なくともエネルギー又は力のいずれかを出力するNNP(Neural Network Potential)を実行するシステムについて説明する。

#### 【0013】

ここで、原子情報は、処理対象の複数の原子に関する情報を含んでいればよく、例えば、各原子の種類および位置に関する情報を含む。原子の位置に関する情報としては、原子の位置を座標によって直接的に示す情報や、原子間の相対位置を直接的に又は間接的に示す情報等が挙げられる。また、原子の位置に関する情報は、原子間の距離、角度、二面角等によって原子間の位置関係を表現する情報であってもよい。原子情報は、原子の種類および位置の情報の他に、電荷に関する情報、原子の結合に関する情報、周期境界条件、Cellのサイズ等の情報を含んでもよい。グラフは、複数の原子の情報を含み、原子の情報をノードとし原子間の接続をエッジとするグラフであってもよい。  
20

#### 【0014】

NNP以外の処理においても、本開示と同等に、処理対象となるデータ、及び、演算途中におけるデータが占めるメモリ領域が大きくなるような演算に対して、本開示における形態を適用してもよい。例えば、適用対象は、原子情報から他の物性値(エネルギー又は力以外の物性値)を計算するニューラルネットワークを用いたシステムであってもよい。  
30

#### 【0015】

図においては、1つのサーバ10が複数のクライアント20と接続されているが、複数のサーバ10が備えられてもよいし、タイミングによっては1つのクライアント20が接続されていてもよい。

#### 【0016】

サーバ10は、クライアント20からの演算要求(クライアント20を使用するユーザからの演算要求)により所定の演算を実行し、クライアント20へと返す情報処理装置である。サーバ10は、例えば、GPUをアクセラレータとして備えることができる。この他、サーバ10は、高速並列演算が可能である任意のプロセッサを備えてもよい。サーバ10は、クラウド上に存在する演算器(例えばGPU)のクラスタを備えてもよい。  
40

#### 【0017】

図1においては省略しているが、この他にゲートウェイサーバ、プロキシサーバ、又は、ロードバランサといった直接的に演算の高速化に関与しない他の構成を備え、演算するサーバに各種要求を適切に振り分けるサーバ構成であってもよい。

#### 【0018】

サーバ10は、一例として、NNPを実行するためのニューラルネットワークを形成し、このニューラルネットワークに原子情報に関するデータを順伝播させることで、エネ  
50

ルギーの値を取得する。また、サーバ 10 は、このエネルギーの値を逆伝播して原子に関する力を取得することもできる。これらに限定されず、サーバ 10 は、NNP を用いて取得可能な任意の物性値を演算して取得してもよい。

#### 【 0 0 1 9 】

サーバ 10 に展開される NNP に係るニューラルネットワークは、種々の入力グラフデータに対応することが可能なニューラルネットワークである。すなわち、サーバ 10 は、このニューラルネットワークにクライアント 20 から入力される種々の原子情報に対応するグラフを入力することで、適切な物性値を取得することが可能である。サーバ 10 は、複数種類のニューラルネットワークを備え、ユーザからの演算要求や処理対象に基づいて、使用するニューラルネットワークを切り替えてよい。

10

#### 【 0 0 2 0 】

上記においては、クライアント 20 からグラフデータを送信するましたが、これに限定されるものではない。クライアント 20 からは処理対象の原子情報を取得し、サーバ 10 側においてグラフデータを生成する形態であってもよい。この他、サーバ 10 とクライアント 20 との間にグラフデータを生成するための異なるサーバが備えられていてもよい。

#### 【 0 0 2 1 】

サーバ 10 は、取得した物性値をクライアント 20 へと出力する。また、サーバ 10 は、クライアント 20 に出力するのではなく、クラウド上のファイルサーバといった任意のストレージに出力をしてもよい。この場合、クライアント 20 に演算が完了した旨の通知をすることもできる。

20

#### 【 0 0 2 2 】

情報処理システム 1 が複数の GPU をサーバ 10 として備える場合、同一の機能、性能を有する GPU によって構成されてもよいし、異なる機能、性能を有する GPU が含まれる構成であってもよい。

#### 【 0 0 2 3 】

クライアント 20 は、ユーザが入出力インターフェースとして用いる情報処理装置である。サーバ 10 は、クライアント 20 を介して入力されたユーザの演算要求に基づいて演算を実行し、演算結果をクライアント 20 へと出力する。

30

#### 【 0 0 2 4 】

上述したように、サーバ 10 において実行する演算の一例は、NNP を用いた物性値の取得に関する演算である。この場合、クライアント 20 は、物性値を測定したい原子情報を受け付け、この受け付けた情報をサーバ 10 に送信して演算を要求することで、ユーザが望む演算を実行させることができる。このように、ユーザは、自らが必要とする、又は、自らの研究対象である原子情報をグラフ化して入力することで、エネルギー、力等の物性値を取得することができる。

#### 【 0 0 2 5 】

上述したように、このグラフ化は、クライアント 20 以外の装置（サーバ 10 等）において実行されるものであってもよい。

40

#### 【 0 0 2 6 】

図2は、一実施形態に係る情報処理システムの処理の一例を示すフローチャートである。左側のフローは、クライアント 20 における処理を示し、右側のフローは、サーバ 10 における処理を示す。

#### 【 0 0 2 7 】

クライアント 20 は、ユーザから処理対象に関する情報（入力データ）を受け付ける（S100）。この入力データは、例えば、物性値を取得したい処理対象（分子、複数の原子等）に関する情報である。一例として、入力データは原子情報である。原子情報は、例えば、原子の種類及び 3 次元空間における位置を示す座標のデータを含むことができる。また、入力データは、処理対象の環境に関するデータ、処理対象の境界条件等の演算に関するデータを含んでもよい。

50

## 【0028】

クライアント 20 は、受け付けた入力データが第 1 条件を満たすか否かを判定する (S102)。この第 1 条件は、入力された原子情報で表される処理対象の原子数で判定される条件であってもよい。情報処理システム 1 は、例えば、ユーザにより入力可能な原子数の最大値を設定してもよく、クライアント 20 は、入力された原子情報で表される対象対象の原子数（入力原子数）を読み取ることにより、S102 の判定を実行する。

## 【0029】

第 1 条件を満たしていない場合、例えば、入力原子数が最大原子数を超えている場合 (S102: NO)、クライアント 20 は、データの入力を待機する状態へと戻ってもよい。別の例として、クライアント 20 は、処理を完了してもよい。クライアント 20 は、これらの処理と併せて、第 1 条件を満たさない旨をディスプレイ等の出力インターフェースを介してユーザに通知してもよい。10

## 【0030】

第 1 条件を満たしている場合、例えば、入力原子数が最大原子数以下である場合 (S102: YES)、クライアント 20 は、受け付けた入力データの処理リクエストをサーバ 10 へと送信する (S104)。クライアント 20 は、この演算要求とともに、入力データを併せてサーバ 10 へと送信することができる。別の例として、クライアント 20 は、サーバ 10 がアクセス可能であるストレージ等のファイルサーバに入力データを送信し、サーバ 10 が適切なタイミングでこの入力データを取得する形態としてもよい。ここで、クライアント 20 からサーバ 10 に送信される演算要求に入力データが含まれてもよいし、受信した演算要求に基づいてサーバ 10 がファイルサーバ等から入力データを取得してもよい。20

## 【0031】

クライアント 20 からの演算要求を受信 (S204) したサーバ 10 は、入力データが第 2 条件を満たすか否かを判定する (S206)。この第 2 条件は、処理対象に含まれる複数の原子の近傍の原子数（近傍数）で判定される条件であってもよい。近傍の原子は、入力された原子において近傍とみなす原子を示す。近傍数は、例えば、入力された原子（ノード）について、演算において相互作用を考慮するべきノードへと接続する関係（エッジ）の総数とすることができる。情報処理システム 1 は、第 1 条件と同様に、ユーザごとに第 2 条件、より具体的には演算可能な近傍数の最大値を設定することができる。30

## 【0032】

サーバ 10 は、例えば、着目しているノードに対して近傍であるノードまでの距離（限定されない一例として、4 ~ 10 といった距離）を設定し、GPU による演算によりそれぞれのノードから設定された距離内に存在するノードの数を取得し、この取得したノードの数の総数をエッジの数、すなわち、近傍数として取得することができる。サーバ 10 は、取得した近傍数が最大近傍数を超えていないか否かにより、S206 の判定を実行する。

## 【0033】

このように、第 2 条件には、クライアント 20 側で演算すると時間が掛かる演算であっても、サーバ 10 側のアクセラレータを用いることで高速に演算することが可能な判定条件を設定することができる。これに対して、第 1 条件は、クライアント 20 側で演算してもそれほどコストが掛からない条件とすることができます。第 1 条件には、例えば、演算要求の送信、回答の受信、サーバ 10 における待ち時間等を考慮するとクライアント 20 で処理した方がよりコストが低い判定条件を設定することができる。40

## 【0034】

第 2 条件を用いた判定は、上記のように S102 においてクライアント 20 で実行されてもよいが、クライアント 20 側で実行すると時間コスト及び演算コストが高くなる可能性があるため、サーバ 10 で NNP に関する演算の前処理（例えば、GNN: Graph Neural Network に関する処理）として取得されることが望ましい。また、サーバ 10 側で判定することにより、高速にエッジ数の探索を実行することが可能になるとともに、50

サーバ 10 側で NNP を構成するニューラルネットワークの仕様が変わったり、距離の設定が変わったりした場合にも、適切に判定をすることが可能になる。

#### 【 0 0 3 5 】

入力データが第 2 条件を満たさない場合 (S206: NO) 、サーバ 10 は、クライアント 20 に対してエラーを送信する (S208)。サーバ 10 は、エラーを送信するとともに、クライアント 20 からの演算要求の受信待機状態へと遷移することができる。

#### 【 0 0 3 6 】

クライアント 20 は、サーバ 10 がエラーを送信した場合には、当該エラーを受信する (S108)。

#### 【 0 0 3 7 】

クライアント 20 は、エラーを受信した場合 (S110: YES) 、サーバ 10 において演算が実行されなかつたと判定し、入力データの受け付け待機状態へと遷移してもよい。

#### 【 0 0 3 8 】

サーバ 10 は、S208 におけるエラーを送信する場合には、入力データの近傍数が最大近傍数を超えた旨のメッセージ（エラーコード）を併せてクライアント 20 へと送信してもよい。クライアント 20 は、このメッセージを参照することで、エラーの原因が第 2 条件を満たさなかつたことであることを取得し、ユーザに通知することができる。

#### 【 0 0 3 9 】

また、サーバ 10 は、このメッセージにユーザが利用可能な最大近傍数及び入力データから推論を行う場合の近傍数を送信してもよい。ユーザは、入力データに対する近傍数を知ることで、結果を取得するための適切な入力データを生成しやすくなる。

#### 【 0 0 4 0 】

クライアント 20 は、エラーを受信しない場合 (S110: NO) 、入力データに対する演算がサーバ 10 において実行されていると判定し、サーバ 10 が結果を出力するのを待機する状態へと遷移する。

#### 【 0 0 4 1 】

別の実装として、クライアント 20 は、S104 の後、サーバ 10 からの結果受信状態となつてもよい。この場合、サーバ 10 から結果を受信した後に、結果がエラーであれば入力データの受け付け待機状態に戻つてもよい。エラーではなく特性値を結果として受信した場合には、クライアント 20 は、適切に結果の受信後、処理を完了したり、入力データの受け付け待機状態に戻つたりしてもよい。また、クライアント 20 は、連続したデータをキューとして入力している場合には、次の演算要求をサーバ 10 に送信してもよい。この場合、クライアント 20 は、キューに入力する前に第 1 条件を満たすかの判定 (S102) を実行してもよい。

#### 【 0 0 4 2 】

第 2 条件を満たす場合 (S206: YES) 、サーバ 10 は、入力データに基づいた NNP の演算を実行し、クライアント 20 から要求された物性値を取得する (S212)。S208 におけるエラーの送信は、S206 における判定のみならず、S212 における演算の実行中に発生したエラーに対して実行することもできる。この場合、クライアント 20 は、実行中に発生したエラーと、その原因等をサーバ 10 から受信することもできる。

#### 【 0 0 4 3 】

サーバ 10 は、演算が完了した後に、演算結果である物性値をクライアント 20 へと送信し (S214) 、処理を完了する。この後、サーバ 10 は、新たな演算要求の待機状態へと遷移してもよい。

#### 【 0 0 4 4 】

このように、情報処理システム 1 は、クライアント 20 が受け付けた入力データである原子情報に基づいてサーバ 10 へと演算要求をし、サーバ 10 において処理された演算結果をクライアント 20 へと出力する SaaS に係るシステムを形成する。

#### 【 0 0 4 5 】

第 1 条件及び第 2 条件は、上述したようにユーザごとに設定することができる。情

10

20

30

40

50

報処理システム 1 は、例えば、ユーザごとに設定されているプラン (SaaS型サービスの利用プラン) にしたがって、第 1 条件及び第 2 条件を設定することができる。

#### 【 0 0 4 6 】

第 1 条件及び第 2 条件は、例えば、ユーザが支払った対価により、設定することができる。情報処理システム 1 は、例えば、高い対価を払っているユーザに対しては、最大原子数、最大近傍数の双方を大きく設定し、それよりも低い対価を払っているユーザに対しては、最大原子数、最大近傍数の少なくとも一方をより小さく設定することができる。

#### 【 0 0 4 7 】

以上のように、本実施形態によれば、情報処理システム 1 は、システムの安定化及び高い対価を支払っているユーザに対して有利な条件で演算を実行するシステムを形成することができる。ユーザにとっても、性能と自らが支払う対価のバランスを考えた上で、適切な対価を支払うことが可能となる。SaaS で実現しているサービスは、計算資源のコストの多くを GPU、GPU メモリといったリソースを用いる演算が占める処理に対して展開されることがある。本開示における形態によれば、各ユーザに設定された利用条件に基づく適切なリソース割り当てを実現することができる。

#### 【 0 0 4 8 】

ユーザは、例えば、値段の異なる複数のプランから受けるサービスを選択することができる。情報処理システム 1 は、限定されない一例として、安価なプラン 1 として、最大原子数を 10,000、最大近傍数を 100,000 とし、中程度のプラン 2 として、最大原子数を 20,000、最大近傍数を 200,000 とし、高価なプラン 3 として、最大原子数を 40,000、最大近傍数を 400,000 等とすることができる。

#### 【 0 0 4 9 】

上記では最大原子数、最大近傍数の双方が支払う対価により上がっているがこれに限定されるものではない。例えば、プラン 2 の最大原子数を 10,000、最大近傍数を 200,000 とすることもできるし、別の例として、最大原子数を 20,000、最大近傍数を 100,000 等とすることもできる。すなわち、最大原子数又は最大近傍数のいずれか一方を、異なるプランと同じ値にすることを排除するものではない。

#### 【 0 0 5 0 】

なお、情報処理システム 1 はさらに、プランごとに最大メモリ使用量を設定してもよい。上記において一例としてあげた NNP の処理は、最大原子数、最大近傍数の他に実行中に必要となるメモリ容量の確保が重要である。このため、サーバ 10 は、実行中ににおける OOM (Out of Memory) のエラーを回避する必要がある。これに鑑みて情報処理システム 1 は、利用できるメモリ容量をプラン決定の要素とすることができる。もちろん、情報処理システム 1 は、これに併せて同時に並列演算することが可能なスレッド数を制限として加えてよい。

#### 【 0 0 5 1 】

上記の実施形態は、以下のような限定されない変形例で実装されてもよい。もちろん、上記の形態及び以下の変形例以外において同じ処理を実現できる他の構成を排除するものではない。

#### 【 0 0 5 2 】

情報処理システム 1 は、本実施形態の例として、第 1 条件及び第 2 条件による判定をした後に演算を実行している。これらの条件の判定は、クライアント 20、サーバ 10、又は、次の実施形態で詳しく説明するクライアント 20 とサーバ 10 間に備えられるゲートウェイのいずれにおいてそれぞれ判定されてもよい。

#### 【 0 0 5 3 】

一例として、サーバ 10 側が第 1 条件及び第 2 条件の双方を判定してもよい。サーバ 10 は、ユーザが選択しているプランに応じて、第 1 条件及び第 2 条件を取得する。上記と同様に、クライアント 20 が演算要求を送信した後、サーバ 10 は、第 1 条件の判定をし、第 1 条件を満たす場合に、演算の前処理として第 2 条件の判定をして

10

20

30

40

50

もよい。

#### 【0054】

サーバ10が1又は複数のクライアント20からの演算タスクをキューで管理する場合、サーバ10は、エンキュー前に第1条件の判定をし、判定結果をクライアント20に送信してもよい。このタイミングで判定することで、クライアント20は、第1条件が満たされていない旨を早いタイミングでユーザに提示することができる。尤も、サーバ10がデキューされた後のタイミングで第1条件の判定をしてよい。

#### 【0055】

一例として、ゲートウェイが第1条件又は第2条件のうち少なくとも一方の判定をしてよい。上記と同様に、第1条件の判定は、複数のタスクをキューで処理する場合には、キューに入る前、又は、キューから取り出した後のいずれのタイミングであってもよい。クライアント20への通知は、判定後の任意のタイミングで実行することができる。

10

#### 【0056】

ゲートウェイが第1条件判定後につづいて第2条件を判定する場合、又は、クライアント20が第1条件を判定し、その後にゲートウェイが第2条件を判定する場合、ゲートウェイは、第2条件を判定するために実行した前処理の結果をサーバ10に送信することができる。ゲートウェイは、エンキュー前に第2条件の判定をしてよいが、第2条件の判定のためにコストが掛かるため、デキュー後に判定をする、又は、待ち行列の先頭のタスクの第2条件を判定する、といった方法を取ることもできる。ゲートウェイは、サーバ10からユーザが選択しているプランの情報を取得することで、第1条件又は第2条件のうちゲートウェイが実行する判定条件を適切に取得して処理することができる。

20

#### 【0057】

一例として、クライアント20が第1条件又は第2条件の判定をする場合、クライアント20は、事前にサーバ10が記憶しているユーザの選択しているプランの情報を、サーバ10から取得することができる。プランの情報を参照することで、クライアント20は、適切な条件の判定を実行することができる。

30

#### 【0058】

一例として、サーバ10が第1条件の判定をすることができる。第1条件の判定をする場合には、サーバ10において記憶されているユーザが選択しているプランを参照することで適切な判定をすることができる。これは、第2条件の判定についても同様である。

30

#### 【0059】

一例として、ユーザとして設定される単位は、個人（自然人）であってもよいし、団体（自然人の団体又は法人）であってもよい。法人の団体は、例えば、企業、学校等の団体であってもよい。

40

#### 【0060】

一例として、プランの設定は、取得対象となる物性値の種類により設定されてもよい。また、プランの設定は、取得対象となる物性値の個数により設定されてもよい。ユーザは、自らが取得したい物性値、又は、自らが取得したい物性値の個数により適切なプランを選択することができる。

#### 【0061】

一例として、プランの設定は、サーバ10に複数備えられるモデルのうち、いずれのモデルを用いるかで設定されてもよい。ユーザは、サーバ10に実装されているモデルから自らが利用するモデルを考慮してプランを選択することができる。

40

#### 【0062】

一例として、サーバ10（又はゲートウェイ）において実行する前処理は、クライアント20からの演算要求に基づいてニューラルネットワークに入力するグラフを構築、又は、グラフを再構築する処理を含んでいてもよい。ユーザは、処理対象の原子情報をクラ

50

イアント 20 が指定するフォーマットで記述し、サーバ 10 でこのフォーマットにしたがった原子情報からグラフを構築することができる。

#### 【0063】

また、ユーザは、サーバ 10 が処理するグラフの構築とは異なるフォーマットのグラフ情報を入力してもよい。サーバ 10 は、フォーマットの変換をすることで、グラフを適切にニューラルネットワークに入力するフォーマットへと再構築することができる。この場合、第 1 条件の判定については、クライアント 20 が当該変換前のフォーマットから原子数を抽出してもよいし、上記に变形例として説明したように、ゲートウェイ又はサーバ 10 において適切なフォーマットに変換後に原子数を取得して第 1 条件の判定をすることができる。

10

#### 【0064】

一例として、サーバ 10 (又はゲートウェイ) において実行する前処理は、グラフを構築した後に、構築されたグラフに基づいてサーバ側で第 1 条件及び第 2 条件の判定を実行することを含んでもよい。

#### 【0065】

構築とは、例えば、上記の GNN においては、ユーザの要求に基づいて GNN に入力されるグラフを構築することを意味することができる。また、構築とは、限定されない他のニューラルネットワークの例においては、ユーザの要求に基づいて利用するニューラルネットワークのモデルに入力する情報を構築することを意味することができる。

20

#### 【0066】

##### (第 2 実施形態)

情報処理システム 1 は、複数のユーザ（複数のクライアント 20）からの演算要求を受けることも可能である。この場合、サーバ 10 の一構成として、キューを制御する情報処理装置が備えられてもよい。複数のクライアント 20 からの演算要求は、キューにより制御される。もちろん、同一のクライアント 20 からの複数の演算要求も、このキューにより制御することもできる。本実施形態においては、この制御について説明する。

#### 【0067】

図3は、一実施形態に係る情報処理システムの概略の一例を模式的に示す図である。本実施形態に係る情報処理システム 1 は、前述の第 1 実施形態に係る情報処理システム 1 に加えて、クライアント 20 からサーバ 10 へのトラフィックを制御するゲートウェイ 30 をさらに備える。

30

#### 【0068】

ゲートウェイ 30 は、クライアント 20 から送信される演算要求について、サーバ 10 において処理をする流量を制御する情報処理装置である。ゲートウェイ 30 は、ユーザごとに設定されている流量に基づいてサーバ 10 による演算実行を制御する。ゲートウェイ 30 は、プロシキサーバであってもよい。

30

#### 【0069】

流量は、前述の実施形態の最大原子数及び最大近傍数と同様に、ユーザごとに設定されているプランにより決定される。この流量の制限をすることで、ユーザに設定されているプランごとに、不平等となる処理をなくすように制御する。流量は、処理速度、所定時間（単位時間を含む）あたりの処理量（使用する演算リソース）等を表すものでもよい。

40

#### 【0070】

ゲートウェイ 30 は、種々の流量制限のアルゴリズムを用いて、サーバ 10 で使用する演算リソースを制限することができる。なお、サーバ 10 は、処理した結果については、ゲートウェイ 30 を介さずに、直接的にクライアント 20 へと出力する構成であってもよい。

#### 【0071】

図4は、一実施形態に係る情報処理システム 1 の処理を示すフローチャートである。

#### 【0072】

クライアント 20 は、ユーザから処理対象に関する情報（入力データ）を受け付ける

50

(S120) と、当該入力データに対する処理をサーバ 10 が実行するようにゲートウェイ 30 へと演算要求を送信する (S122)。クライアント 20 は、演算要求を送信する前に、前述の実施形態におけるユーザごとに設定されている第 1 条件における判定を実行してもよい。

#### 【0073】

ゲートウェイ 30 は、クライアント 20 が送信した演算要求を受信する (S322)。ゲートウェイ 30 は、キューを準備しておき、受信した演算要求をエンキューし、原子数に基づく流量制御により演算要求をデキューしてもよい。

#### 【0074】

ゲートウェイ 30 は、流量制御に基づいて制御されたデータ（流量制限に基づく演算要求）をサーバ 10 へと送信する (S324)。流量制御の例については、後述にて詳しく説明する。10

#### 【0075】

サーバ 10 は、ゲートウェイ 30 を介して入力データを受信する (S224)。前述の実施形態と同様に、入力データ自体は、情報処理システム 1 内に備えられるストレージを介して取得してもよい。ここで、ゲートウェイ 30 からサーバ 10 に送信される演算要求に入力データが含まれてもよいし、受信した演算要求に基づいてサーバ 10 がストレージ等から入力データを取得してもよい。

#### 【0076】

サーバ 10 は、エラーが発生した場合には (S226: YES)、クライアント 20 へとエラーを送信する。クライアント 20 は、サーバ 10 からエラーが送信された場合には、当該エラーを受信する (S126)。サーバ 10 は、例えば、前述の実施形態におけるユーザごとに設定されている第 2 条件における判定を実行して、エラーがあるか否かを判定してもよい。20

#### 【0077】

サーバ 10 は、演算を実行する (S228)。前述した実施形態と同様に、サーバ 10 は、NNP に係るニューラルネットワークにクライアント 20 から送信された原子情報に基づくグラフを入力することで物性値を取得する演算を実行する。実行中にエラーが発生した場合は、サーバ 10 は、クライアント 20 にエラーを返すこともできる。

#### 【0078】

サーバ 10 は、物性値の推論後、取得した結果をクライアント 20 へと出力する (S230)。クライアント 20 は、サーバ 10 から出力された推論結果を受信し (S130)、適切にその後の処理を実行することができる。30

#### 【0079】

このように、情報処理システム 1 は、ゲートウェイ 30 を介することで、クライアント 20 から送信される入力データの流量を制御した上で、サーバ 10 に適切な処理を実行させることができる。

#### 【0080】

適切な流量の制御をすることで、ユーザごとの処理を公平に実行することができる。前述の実施形態と同様に、ユーザは、例えば、値段の異なる複数のプランから受けるサービスを選択することができる。情報処理システム 1 は、対価が高いプランになるほど、流量を大きくするように設定することで、プランごとに公平なサーバ 10 の利用をユーザに提供することが可能となる。40

#### 【0081】

例えば、前述した第 1 実施形態における第 1 条件及び第 2 条件によるプランの分け方である場合を考える。同じプランを選択しているユーザ間で、設定されている最大原子数の範囲において、大きな原子数を有する入力データを投入したユーザは、それよりも小さな原子数を有する入力データを投入したユーザよりもリソースの割り当てが多くなる可能性がある。この場合、小さな原子数を有する入力データを投入したユーザは、不公平を感じる可能性がある。

**【 0 0 8 2 】**

ここで、流量制限をして所定時間（単位時間を含む）あたりに処理することが可能な原子数を制限することにより、同じプランを選択しているユーザ間の不公平感を抑制することが可能となる。

**【 0 0 8 3 】**

次に、流量制御について例を挙げて説明する。情報処理システム 1 は、流量制限をすることで、演算リソースとして利用できる GPU、GPUメモリ をより公平にユーザに割り当てることが可能となる。ゲートウェイ 30 は、1 つのキューを管理し、このキューにクライアント 20 からの演算要求に対してユーザごとに設定されている流量制限に基づくエンキュー及びデキューを実行することで複数の GPU コアを用いた演算を実行する。10

**【 0 0 8 4 】**

ゲートウェイ 30 は、限定されない一例として、トーカンバケットアルゴリズムに基づいた流量制御をすることができる。

**【 0 0 8 5 】**

ゲートウェイ 30 は、例えば、接続されるクライアント 20 ごとにバケットを備える。このバケットには、ユーザが選択しているプランにしたがった速度でトーカンが追加される。このトーカンは、例えば、構造に関する情報（原子数等）に基づく情報であってもよい。20

**【 0 0 8 6 】**

ゲートウェイ 30 は、プランに応じて  $1 / r$  秒ごとに 1 個のトーカンをバケットに追加してもよいし、又は、プランに応じて 1 秒ごとに  $r$  個のトーカンをバケットに追加してもよい。より具体的には、(1 トーカン) = (1 原子) としてもよく、ゲートウェイ 30 は、例えば、 $1 / r$  秒ごとに原子 1 個の演算を、又は、1 秒ごとに原子  $r$  個の演算を可能とするトーカンをバケットに追加する。

**【 0 0 8 7 】**

また、ゲートウェイ 30 は、ユーザが選択しているプランにしたがってバケットの大きさを変えることもできる。バケットの大きさは、プランで指定された原子数に基づいて決定してもよい。

**【 0 0 8 8 】**

ユーザがクライアント 20 を介して演算要求を送信すると、ゲートウェイ 30 は、クライアント 20 から送信された演算要求をエンキューする。ゲートウェイ 30 は、同一のユーザからの複数の演算要求をエンキューしてもよい。ゲートウェイ 30 は、キューの先頭に保存されている演算要求で特定される処理対象の原子数と、現在バケットに保持されているトーカンに基づいた演算可能な原子数とを比較する。30

**【 0 0 8 9 】**

バケットに保持されているトーカンに基づいた原子数が処理対象の原子数以上である場合には、ゲートウェイ 30 は、当該演算要求をデキューする。入力データの原子数の方が大きい場合には、ゲートウェイ 30 は、バケットに保持される演算可能な原子数が処理対象の原子数以上となるまで演算要求を保留状態とする。

**【 0 0 9 0 】**

別の例として、ゲートウェイ 30 は、保留状態にするのではなく、トーカン量が足りない場合には、エラーをクライアント 20 へと返してもよい。クライアント 20 は、クールタイムにおいてトーカンを快復した後にリトライを要求するといった処理ができる。40

**【 0 0 9 1 】**

ゲートウェイ 30 は、接続されるクライアント 20 ごとに FIFO (First In First Out) を備えてもよく、バケットに保持されているトーカンに基づいてこの FIFO からサーバ 10 へ演算要求をするキューへと演算タスクを進める形態であってもよい。すなわち、ゲートウェイ 30 は、クライアント 20 からの演算要求を保持するクライアント 250

0ごとに備えられるキューと、サーバ10へと演算要求をする1つのキューとを管理してもよい。

#### 【0092】

サーバ10は、ゲートウェイ30がエンキューした演算要求を適切にデキューすることで、順次演算を実行する。複数のGPUを有する場合には、適切に演算要求を割り振ることにより、並列的に複数のクライアント20からの演算要求を処理することができる。

#### 【0093】

例えば、サーバ10は、ゲートウェイ30に現在処理可能である演算要求があるか否かの確認をしてもよい。サーバ10は、処理可能である演算要求がある場合には、当該演算要求と関連するデータを取得し、演算処理を開始することができる。10

#### 【0094】

このように流量制限をすることで、プランごとに1秒あたりに処理することが可能な原子数を設定することが可能となる。

#### 【0095】

具体的な例としては、流量制限がない場合、2000原子を投入したユーザAと、1000原子を投入したユーザBとが同時にこのデータに係る演算要求をした場合には、同じタイミングで演算結果を取得することになる。この結果、同じプランを選択している場合であっても、ユーザAの方が時間あたりに利用する演算リソースの量が多いにも拘わらず、ユーザBは、ユーザAと同じ対価を支払うことになる。20

#### 【0096】

一方で、上記のように流量制限をすることで、同時に上記のユーザAとユーザBがデータを投入した場合、情報処理システム1は、ユーザBの演算については1000原子分のトークンが保持されたタイミングで演算処理を実行する。このタイミングにおいて、ユーザAの演算処理は、待機状態を維持する。上記と同様に、待機状態とするのではなく、エラーを返す処理をしてもよい。エラーに対して、クライアント20は、トークンが快復するまで待ってリトライすることができる。ユーザAの演算については、2000原子分のトークンが保持されたタイミング、すなわち、ユーザBの演算待ち時間の2倍の待ち時間の後、演算処理が実行される。

#### 【0097】

この結果、時間あたりの処理する原子数を同じプランを選択したユーザごとに同等にすることができる。もちろん、ユーザBは、続けて1000原子あたりのデータを投入してもよく、この場合、オーバーヘッドを除いて、ユーザAが投入したデータに係る演算時間と、ユーザBが投入したデータに係る演算時間とを、ほぼ同じにすることができます。このように、同じプランを選択しているユーザ間の不公平感を払拭することが可能となる。30

#### 【0098】

上記の流量制御処理は、ゲートウェイ30において実行されたが、これに限定されるものではない。例えば、クライアント20側において演算要求を送信するのに用いるソフトウェア（アプリケーション）又はAPI（Application Programming Interface）により制御して、ゲートウェイ30へと適切なトークンが保持されたタイミングで演算要求をする実装とすることも可能である。この場合、ゲートウェイ30において、クライアント20が送信した演算要求が適切なトークン制御によるものであるかを確認する整合性チェックをしてもよい。このような実装とすることで、クライアント20側でゲートウェイ30へと演算要求を送信するためのキューを保持することが可能となり、ゲートウェイ30においては演算処理を低減させることができるとともに、クライアント20においては連続的に演算要求を送信する場合における未送信の要求の削除又は別の要求への置き換え等を通信の遅延等を考慮することなく実現することができる。40

#### 【0099】

上記においては、バーストを許可した上で、処理原子数のトークンがたまつたタイミン50

グで演算要求をする例を説明したが、これに限定されるものではない。通常のトークンバケットモデルのように、単位時間あたりの流量に応じたデータをゲートウェイ 30 からサーバ 10 へと送信する形態とすることもできる。

#### 【0100】

この例についての処理の概略を説明する。

#### 【0101】

図5は、一実施形態に係る流量制御の一例を説明するための図である。

#### 【0102】

情報処理システム 1 は、上記のトークンバケットに基づく手法、又は、他の流量制御をするアルゴリズムを用いることができる。利用するアルゴリズムは、(1)定期的にトークンが付与される、(2)トークンが残っていればユーザの演算要求を通す、(3)トークンが残っていない場合は、ユーザの演算要求をキューで保持する、といった処理を実現できるアルゴリズムであればよい。10

#### 【0103】

情報処理システム 1 は、ユーザが選択しているプランに応じて、(A)所定時間（単位時間を含む）にユーザに追加される第 1 トークン量、(B)ユーザが保有可能な第 2 トークン量（例えば、トークンバケットモデルであれば、バケットの大きさ）、(C)所定時間（単位時間を含む）にユーザが消費可能な第 3 トークン量、のうち、少なくとも 1 つを設定することができる。

#### 【0104】

図5に示すように、ゲートウェイ 30 は、クライアント 20 ごと（ユーザごとでもよい）に備える第 1 キュー 40 と、サーバ 10 がタスクをデキューするための第 2 キュー 42 とを備える。第 1 キュー 40 は、必須の構成ではない。別の例として、第 1 キュー 40 は、クライアント 20 に備えられてもよい。20

#### 【0105】

ゲートウェイ 30 は、仮想的なバケツ 44 にトークン 46 を所定時間ごとに 1 個、又は、単位時間ごとに所定個ずつ補給（トークンの回復）する。この回復量に関するパラメータが上記の(1)、(A)に対応する。バケツ 44 に蓄積できるトークン 46 の最大量が(B)に対応する。

#### 【0106】

第 1 キュー 40 がある場合には、第 1 キュー 40 にクライアント 20 からの演算要求が待ち行列として格納される。バケツ 44 に待ち行列の先頭の処理に必要な量のトークンが保持されている場合には、ゲートウェイ 30 は、第 1 キュー 40 から当該演算要求をデキューして、サーバ 10 が処理をするための第 2 キュー 42 にエンキューする。この処理が上記の(2)の処理に対応する。この第 2 キュー 42 にキューイングされている演算要求は、サーバ 10 が処理可能なタイミングで取得し、演算が実行される。30

#### 【0107】

第 1 キュー 40 をゲートウェイ 30 に備えない場合には、ゲートウェイ 30 は、クライアントからの演算要求を受信すると、当該演算要求を処理するための量のトークン 46 がバケツ 44 に保持されているか否かを判定する。40

#### 【0108】

十分な量のトークンがある場合には、ゲートウェイ 30 は、クライアント 20 から受信した演算要求を第 2 キュー 42 へとエンキューする。この処理が上記の(2)に対応する。

#### 【0109】

十分な量のトークンがない場合には、ゲートウェイ 30 は、クライアント 20 に流量制限により処理できない旨のエラーを返し、受信した演算要求を破棄してもよい。この場合、クライアント 20 は、トークンの回復を待って、ゲートウェイ 30 に対する要求送信のリトライをすることができる。この処理が上記の(3)に対応する。

#### 【0110】

10

20

30

40

50

十分な量のトークンがある場合であっても、上記の(C)のパラメータにより、クライアント 20 からの演算要求を許可しないことができる。例えば、単位時間あたりに使用することのできるトークンを制御することで、ゲートウェイ 30 は、この処理を実現することができる。

#### 【0111】

上記に説明したトークンは、限定されない一例として、原子数、又は、原子数に基づく値とすることができます。また、トークンは、ニューラルネットワークを用いた演算リソースに影響を与える他の値であってもよい。限定されない一例として、サーバ 10 においてグラフに関連するニューラルネットワークの処理をする場合には、トークンは、処理するグラフのノード数とすることができますし、エッジ数を考慮した値とすることもできる。その他、トークンは、サーバ 10 で用いるニューラルネットワークに基づいて、サーバ 10 における処理量又は使用するメモリ量（使用する／占有するリソース量）に応じた値とすることができます。10

#### 【0112】

いずれの例においても、情報処理システム 1 は、高額なプランを選択しているユーザのトークンに関する量が大きくなる様に設定することができる。このように設定することで、ユーザの満足度をより高く充足することが可能となる。

#### 【0113】

上記において、高額なプランにおけるトークンに関する量と、低額なプランにおけるトークンに関する量とのうち少なくとも 1 つが同じ値になることを排除するものではない。20

#### 【0114】

本実施形態においては、利用可能であるリソース（トークン量）をクライアント 20 を介してユーザに通知してもよい。この通知も、利用可能な残りの原子数、割合等を数値やグラフ表示等、ユーザにわかりやすい任意の形態とすることができる。

#### 【0115】

以上のように、本実施形態によれば、プランごとに不公平な状態が生まれない制御をすることができる。全てのクライアント 20 からの演算要求をゲートウェイ 30 においてキューで制御することで、サーバ 10 は、流量制御に基づいた適切な演算処理を実現することができる。30

#### 【0116】

この流量制限は、トークンバケットモデル以外のモデルを用いることもできる。例えば、流量制限として、リーキーバケットモデルを用いてもよい。

#### 【0117】

なお、図4においては、クライアント 20 がエラーを返すのは、サーバ 10 からの通知のみが示されているが、これに限定されるものではない。クライアント 20 は、例えば、ゲートウェイ 30 から流量制限を超えていたエラーを取得した場合に、ユーザに対してエラーを通知してもよい。クライアント 20 は、流量制限を超えていたデータをリトライする旨、ゲートウェイ 30 に通知してもよい。リトライをしている間に、トークンがバケットに追加されている可能性があるため、リトライ後に流量制限を解消できる可能性がある。クライアント 20 は、所定回数のリトライ後に処理不可のエラーをユーザに通知することもできる。例えば、ゲートウェイ 30 に送信する段階で保留中の入力データがあるような場合、クライアント 20 は、エラーを返すこともできる。40

#### 【0118】

前述したように、第 1 実施形態と第 2 実施形態は組み合わせて用いることができる。組み合わせる場合、プランごとにより柔軟な制限を加えることができる。

#### 【0119】

また、本開示における処理は、NNP だけに限定されるものではなく、例えば、GNN による演算を実行する SaaS においても同様に実装することが可能である。また、その他の SaaS の処理においても、メモリサイズに影響を与えるような演算を実行するサー

ビスに適用することもできる。

**【 0 1 2 0 】**

また、ゲートウェイ 30 は、キューを少なくとも 1 つのメモリを用いて管理することができる。

**【 0 1 2 1 】**

また、情報処理システム 1 は、少なくとも 1 つのメモリにそれぞれのユーザの利用条件、例えば、ユーザが選択しているプランと、プランに紐付けられている条件を記憶部(記憶回路)に記憶している。

**【 0 1 2 2 】**

本開示における情報処理システム 1 は、前述のそれぞれの実施形態における構成の他、適切に処理が可能な任意の構成とすることもできる。 10

**【 0 1 2 3 】**

情報処理システム 1 は、例えば、1 台以上のサーバ 10 を備える構成であってもよい。情報処理システム 1 は、この構成を用いて、前述のそれぞれの実施形態に記載した機能を実現してもよい。

**【 0 1 2 4 】**

情報処理システム 1 は、例えば、1 台以上のサーバ 10 及び 1 台以上のゲートウェイ 30 を備える構成であってもよい。情報処理システム 1 は、この構成を用いて、前述のそれぞれの実施形態に記載した機能を実現してもよい。

**【 0 1 2 5 】**

情報処理システム 1 は、例えば、1 台以上のサーバ 10 、1 台以上のゲートウェイ 30 、及び 1 台以上のクライアント 20 を備える構成であってもよい。情報処理システム 1 は、この構成を用いて、前述のそれぞれの実施形態に記載した機能を実現してもよい。 20

**【 0 1 2 6 】**

情報処理システム 1 は、例えば、1 台以上のサーバ 10 、及び 1 台以上のクライアント 20 を備える構成であってもよい。情報処理システム 1 は、この構成を用いて、前述のそれぞれの実施形態に記載した機能を実現してもよい。

**【 0 1 2 7 】**

本開示において、ユーザとは、個人、団体、又は、団体のメンバであってもよい。団体は、例えば、企業、学校であってもよい。情報処理システム 1 においてユーザは、識別子により管理されていてもよい。情報処理システム 1 において、例えば、このユーザの識別子がクライアント 20 からサーバ 10 又はゲートウェイ 30 に送信され、適切にクライアント 20 からの要求を処理することができる。 30

**【 0 1 2 8 】**

本開示において、少なくとも 1 つのプロセッサによって実行される、処理をするか否かの判定(処理可否の判定)は、少なくとも、要求の受付の可否(受付の要否)、データの受付の可否(受付の要否)、要求に基づく処理の実行の可否(実行の要否)、又は、要求がユーザに設定された利用条件に沿っているか否かの判定を含む概念である。これに対する処理可能という判定は、要求の受付が可能(受付が必要)、データの受付が可能(受付が必要)、要求に基づく処理の実行が可能(実行が必要)、又は、要求がユーザに設定された利用条件に沿っている、という判定を含む概念である。 40

**【 0 1 2 9 】**

本開示において、ニューラルネットワークを用いた処理は、限定されない一例として前述したそれぞれの実施形態に示すように NNP を用いて、少なくとも力又はエネルギーのいずれかを計算すること、処理対象に関する他の物性値等を算出することを含む。また、1 又は複数のニューラルネットワークを用いて、処理対象に関する 1 種類以上の解析値を算出することを含む。

**【 0 1 3 0 】**

また、ニューラルネットワークを用いた処理は、限定されない一例として、ニューラル 50

ネットワークを用いて処理対象の分類等を実行する処理を含むことができる。

#### 【0131】

また、ニューラルネットワークは、一例としてグラフニューラルネットワークを用いたものを説明したが、これに限定されるものではなく、CNN (Convolutional Neural Network)、MLP (Multi-Layer Perceptron) 等の任意の形態のモデルが少なくとも一部に備えられるモデルであってもよい。

#### 【0132】

本開示において、ユーザからの要求／ユーザの要求は、前述のそれぞれの実施形態で説明したようにニューラルネットワークを用いた演算要求であってもよい。また、この要求は、ニューラルネットワークでの処理対象に関する情報が含まれていてもよい。一例として、原子情報が含まれてもよいし、処理対象の構造を示す情報が含まれてもよいし、又は、処理対象の構造を間接的に示す情報が含まれてもよい。もちろん、これら以外の処理に用いる情報が含まれてもよい。

10

#### 【0133】

本開示において、利用条件は、実施形態において説明した第1条件、第2条件、及び、流量制御に関する条件の上位概念とすることができる。すなわち、この利用条件は、ユーザの契約プラン等に応じて決定される条件であればよく、例えば、実施形態において説明した第1条件、第2条件、及び、流量制御に関する条件のうちいずれか1つ以上を含む条件であればよい。また、利用条件は後述する構造に関する条件を含む。

20

#### 【0134】

本開示において、処理対象は、複数の原子、分子、結晶、化合物、物質、物体等であってもよい。また、処理対象は、複数の原子、分子、結晶、化合物、物質、物体等を間接的に表す情報であってもよい。また、処理対象は、ニューラルネットワークに入力される情報、例えば、グラフであってもよい。また、処理対象は、ニューラルネットワークを用いた解析対象であってもよい。

#### 【0135】

本開示において、構造に関する条件は、処理対象の構造に関する下記条件を含む概念であってもよい。構造に関する条件は、例えば、下記に例としてあげるいずれか1つを用いた条件であってもよいし、複数の条件を組み合わせた条件であってもよい。前述したそれぞれの実施形態にお変える各種条件のいずれか1つを下記のいずれかに挙げた条件に置き換えることもできる。

30

#### 【0136】

条件の一例は、処理対象の原子数に関する条件（例えば、トータルの原子数でもよいし所定の一部の原子数を基準にした数）であってもよい。原子数に関する条件は、原子数、原子数に基づいて決定されるグラフのノード数、原子数と同じ値を示す情報、原子数を間接的に表す情報、原子数に比例する情報といった情報であってもよい。

40

#### 【0137】

条件の一例は、処理対象の原子の近傍数に関する条件（例えば、トータルの近傍数でもよいし、所定の一部の近傍数を基準にした数）であってもよい。近傍数に関する条件は、原子の近傍数、原子の近傍数に基づいて決定されるグラフのエッジ数、原子の近傍数と同じ値を示す情報、原子の近傍数を間接的に表す情報、原子の近傍数に比例する情報といった情報であってもよい。

#### 【0138】

条件の一例は、ニューラルネットワークを用いた演算リソース (GPU、GPUメモリのリソース等) に影響を与えるパラメータであってもよい。ニューラルネットワークを用いたリソースに影響を与えるパラメータは、ニューラルネットワークの種類、処理対象に含まれる元素の種類（種類の数）、処理対象の空間的な大きさ、電子数、陽子数、中性子数、原子量、上記各情報と同じ値を示す情報、上記各種情報を間接的に表す情報、上記各情報に比例する情報等であってもよい。

#### 【0139】

50

上記では、ニューラルネットワークを一例として記載したが、他のモデル又はアルゴリズムを用いた演算にも本開示に記載の実施形態を適用することが可能である。また、これらの場合、ユーザの要求に対して演算時間、演算リソースを左右する要素でプランを設定する、といったことが可能である。

#### 【0140】

本開示において、トークン量は、それぞれのユーザが利用可能なリソースの制御に用いられる情報であってもよい。また、トークン数を直接的又は間接的に表すものであってもよい。

#### 【0141】

本開示において、要求の量は、要求の数を表してもよい。また、要求の量は、ニューラルネットワークの処理対象の量（又は数）を直接的に又は間接的に表してもよい。10

#### 【0142】

要求の量の制御の一例として、構造に関する情報が用いられてもよい。この構造に関する情報は、以下に挙げる限定されない例のうちの少なくともいずれか1つを含んでもよいし、以下に挙げる例のうち複数の組み合わせであってもよい。

#### 【0143】

構造に関する情報の例として、原子数に関する情報（例えば、原子数、原子数に基づいて決定されるグラフのノード数、原子数と同じ値を示す情報、原子数を間接的に表す情報、原子数に比例する情報等）を用いてもよい。

#### 【0144】

構造に関する情報の例として、原子の近傍数に関する情報（例えば、原子の近傍数、原子の近傍数に基づいて決定されるグラフのエッジ数、原子の近傍数と同じ値を示す情報、原子の近傍数を間接的に表す情報、原子の近傍数に比例する情報等）を用いてもよい。20

#### 【0145】

構造に関する情報の上記以外の限定されないその他の例として、元素の種類の数、電子数、陽子数、中性子数、原子量、上記各情報と同じ値を示す情報、上記各種情報を間接的に示す情報、又は、上記各情報に比例する情報を用いてもよい。また、ニューラルネットワークにおける処理対象の構造を表現する際に利用される情報を用いてもよい。さらに、ニューラルネットワークの処理対象の量（数）を直接的に又は間接的に表現可能な情報を用いてもよい。30

#### 【0146】

上記において説明した第1トークン、第2トークン、又は、第3トークンの量は、構造に関する情報を直接的又は間接的に表す値として設定されてもよい。また、この量は、構造に関する情報を用いた所定の関数によって設定されてもよいし、構造に関する情報に比例するように設定されてもよい。つまり、第1トークン、第2トークン、又は、第3トークンの量は、構造に関する情報に基づいて設定されてもよい。

#### 【0147】

それぞれのユーザの利用条件に基づいて、要求の量の制御に用いられる構造に関する情報が決定されてもよい。また、要求の量の制御に用いられる構造に関する情報が、ユーザの利用条件と紐付けられていてもよい。

#### 【0148】

それぞれのユーザの利用条件に、要求の量の制御に用いられる構造に関する情報が含まれていてもよい。例えば、高額な契約をしたユーザが低額な契約をしたユーザ以上の要求の量を処理可能となるような設定になつていればよい。

#### 【0149】

ユーザに設定された第1トークン量は、契約するプランにより制限をすることができる。例えば、あるプランを選択しているユーザの第1トークン量は、それよりも低額なプランを選択しているユーザに設定されている第1トークン量以上のトークン量とすることができる。

#### 【0150】

10

20

30

40

50

同様に、あるプランを選択しているユーザの第2トークン量は、例えば、それよりも低額なプランを選択しているユーザに設定されている第2トークン量以上とすることができる。あるプランを選択しているユーザの第3トークン量は、例えば、それよりも低額なプランを選択しているユーザに設定されている第3トークン量以上とすることができる。

#### 【0151】

ユーザの利用条件として、最大原子数を10,000、最大近傍数を100,000とする安価なプラン1、最大原子数を20,000、最大近傍数を200,000とする中程度のプラン2、最大原子数を40,000、最大近傍数を400,000とする高価なプラン3が提供されているとする。この場合、情報処理システム1は、プラン3における第1～第3トークン量が、プラン1及びプラン2における第1～第3トークン量以上になるように設定してもよい。また、ユーザの利用条件において、直接的に又は間接的に第1～第3トークン量を設定してもよい。また、第1～第3トークン量を、各プランの利用条件に含まれる他の条件に基づいて設定してもよい。また、第1実施形態における条件に基づいて、要求の量の制御に用いられる条件が設定されてもよい。

10

#### 【0152】

このような設定とすることで、情報処理システム1は、高額なプランを選択しているユーザに対して、低額なプランを選択しているユーザよりもよい演算環境を提供することが可能となる。

20

#### 【0153】

本開示における実施形態は、以下のように書き換えることもできる。

#### 【0154】

(1)

少なくとも1つのメモリと、  
少なくとも1つのプロセッサと、を備え、  
前記少なくとも1つのプロセッサは、

ユーザの要求を受信し、

前記ユーザの利用条件に基づいて、前記要求の処理可否を判定し、

処理可能と判定された前記要求に基づいて、ニューラルネットワークを用いた処理を実行し、

30

前記利用条件は、前記ニューラルネットワークの処理対象の構造に関する条件を含む、情報処理システム。

#### 【0155】

(2)

前記利用条件は、少なくとも前記処理対象の原子数に関する条件又は前記処理対象の原子の近傍数に関する条件のいずれかを含む、

(1)に記載の情報処理システム。

#### 【0156】

(3)

前記ユーザの利用条件における原子数は、前記ユーザより安価なプランで前記ニューラルネットワークを用いたサービスを契約している他のユーザの利用条件における原子数以上であり、

40

前記ユーザの利用条件における原子の近傍数は、前記他のユーザの利用条件における原子の近傍数以上である、

(2)に記載の情報処理システム。

#### 【0157】

(4)

前記少なくとも1つのプロセッサは、

前記要求に基づいて、前記ニューラルネットワークに入力する情報を構築し、

構築された前記情報と前記利用条件とに基づいて、前記要求の処理可否を判定する、

50

(1)～(3)のいずれかに記載の情報処理システム。

【0158】

(5)

構築された前記情報はグラフである、

(4)に記載の情報処理システム。

【0159】

(6)

前記ニューラルネットワークは、グラフニューラルネットワークである、

(1)～(5)のいずれかに記載の情報処理システム。

【0160】

10

(7)

前記ニューラルネットワークは、NNP(Neural Network Potential)のモデルである、

(1)～(6)のいずれかに記載の情報処理システム。

【0161】

(8)

前記ユーザの利用条件は、前記ニューラルネットワークの種類に基づいて設定される、

(1)～(7)のいずれかに記載の情報処理システム。

【0162】

20

(9)

前記少なくとも1つのプロセッサは、

前記ユーザから複数の要求を受信し、

前記複数の要求を前記少なくとも1つのメモリに記憶し、

構造に関する情報に基づいて、前記少なくとも1つのメモリに記憶された前記複数の要求のうち、前記ニューラルネットワークを用いた処理を実行する要求の量を制御する、

(1)～(8)のいずれか一項に記載の情報処理システム。

【0163】

(10)

前記構造に関する情報は、少なくとも原子数に関する情報又は原子の近傍数に関する情報のいずれかを含む、

30

(9)に記載の情報処理システム。

【0164】

(11)

少なくとも1つのメモリと、

少なくとも1つのプロセッサと、を備え、

前記少なくとも1つのプロセッサは、

ユーザから複数の要求を受信し、

前記複数の要求を前記少なくとも1つのメモリに記憶し、

前記少なくとも1つのメモリに記憶された前記複数の要求のうち、ニューラルネットワークを用いた処理を実行する要求の量を制御し、

40

前記要求の量は、構造に関する情報に基づいて制御される、

情報処理システム。

【0165】

(12)

前記構造に関する情報は、少なくとも原子数に関する情報又は原子の近傍数に関する情報のいずれかを含む、

(11)に記載の情報処理システム。

【0166】

(13)

前記量は、所定時間に前記ユーザに追加される第1トークン量、前記ユーザが保有可能

50

な第2トークン量又は所定時間に前記ユーザが消費可能な第3トークン量のいずれか一つを用いて制御される、

(11)又は(12)に記載の情報処理システム。

【0167】

(14)

前記第1トークン量は、前記構造に関する情報に基づく値であり、

前記第2トークン量は、前記構造に関する情報に基づく値であり、

前記第3トークン量は、前記構造に関する情報に基づく値である、

(13)に記載の情報処理システム。

【0168】

10

(15)

前記構造に関する情報は、少なくとも原子数に関する情報又は原子の近傍数に関する情報のいずれかを含む、

(14)に記載の情報処理システム。

【0169】

(16)

前記ユーザに設定された前記第1トークン量は、前記ユーザより安価なプランで前記ニューラルネットワークを用いたサービスを契約している他のユーザに設定された第1トークン量以上であり、

前記ユーザに設定された前記第2トークン量は、前記他のユーザに設定された第2トークン量以上であり、

前記ユーザに設定された前記第3トークン量は、前記他のユーザに設定された第3トークン量以上である、

(14)又は(15)に記載の情報処理システム。

【0170】

(17)

前記ニューラルネットワークは、グラフニューラルネットワークである、

(10)~(16)のいずれかに記載の情報処理システム。

【0171】

(18)

前記グラフニューラルネットワークは、NNP(Neural Network Potential)のモデルである、

(10)に記載の情報処理システム。

【0172】

本開示における内容は、米国仮出願63/209,419、及び当該仮出願を優先権の基礎とする国際出願PCT/JP2022/023519の全ての内容を参照し、援用している。

【0173】

前述した実施形態における各装置(情報処理装置)の一部又は全部は、ハードウェアで構成されていてもよいし、CPU(Central Processing Unit)又はGPU(Graphics Processing Unit)等が実行するソフトウェア(プログラム)の情報処理で構成されてもよい。ソフトウェアの情報処理で構成される場合には、前述した実施形態における各装置の少なくとも一部の機能を実現するソフトウェアを、CD-ROM(Compact Disc-Read Only Memory)、USB(Universal Serial Bus)メモリ等の非一時的な記憶媒体(非一時的なコンピュータ可読媒体)に収納し、コンピュータに読み込ませることにより、ソフトウェアの情報処理を実行してもよい。また、通信ネットワークを介して当該ソフトウェアがダウンロードされてもよい。さらに、ソフトウェアの処理の全部又は一部がASIC(Application Specific Integrated Circuit)又はFPGA(Field Programmable Gate Array)等の回路に実装されることにより、当該ソフトウェアによる情報処理がハードウェアにより実行されてもよい。

【0174】

30

40

50

ソフトウェアを収納する記憶媒体は、光ディスク等の着脱可能なものでもよいし、ハードディスク又はメモリ等の固定型の記憶媒体であってもよい。また、記憶媒体は、コンピュータ内部に備えられてもよいし（主記憶装置または補助記憶装置等）、コンピュータ外部に備えられてもよい。

#### 【0175】

図6は、前述した実施形態における各装置（情報処理装置）のハードウェア構成の一例を示すブロック図である。各装置は、一例として、プロセッサ71と、主記憶装置72（メモリ）と、補助記憶装置73（メモリ）と、ネットワークインターフェース74と、デバイスインターフェース75と、を備え、これらがバス76を介して接続されたコンピュータ7として実現されてもよい。

10

#### 【0176】

図6のコンピュータ7は、各構成要素を一つ備えているが、同じ構成要素を複数備えてもよい。また、図6では、1台のコンピュータ7が示されているが、ソフトウェアが複数台のコンピュータにインストールされて、当該複数台のコンピュータそれがソフトウェアの同一の又は異なる一部の処理を実行してもよい。この場合、コンピュータそれがネットワークインターフェース74等を介して通信して処理を実行する分散コンピューティングの形態であってもよい。つまり、前述した実施形態における各装置（情報処理装置）は、1又は複数の記憶装置に記憶された命令を1台又は複数台のコンピュータが実行することで機能を実現するシステムとして構成されてもよい。また、端末から送信された情報をクラウド上に設けられた1台又は複数台のコンピュータで処理し、この処理結果を端末に送信するような構成であってもよい。

20

#### 【0177】

前述した実施形態における各装置（情報処理装置）の各種演算は、1又は複数のプロセッサを用いて、又は、ネットワークを介した複数台のコンピュータを用いて、並列処理で実行されてもよい。また、各種演算が、プロセッサ内に複数ある演算コアに振り分けられて、並列処理で実行されてもよい。また、本開示の処理、手段等の一部又は全部は、ネットワークを介してコンピュータ7と通信可能なクラウド上に設けられたプロセッサ及び記憶装置の少なくとも一方により実現されてもよい。このように、前述した実施形態における各装置は、1台又は複数台のコンピュータによる並列コンピューティングの形態であってもよい。

30

#### 【0178】

プロセッサ71は、少なくともコンピュータの制御又は演算のいずれかを行う電子回路（処理回路、 Processing circuit、 Processing circuitry、 CPU、 GPU、 FPGA、 ASIC 等）であってもよい。また、プロセッサ71は、汎用プロセッサ、特定の演算を実行するために設計された専用の処理回路又は汎用プロセッサと専用の処理回路との両方を含む半導体装置等のいずれであってもよい。また、プロセッサ71は、光回路を含むものであってもよいし、量子コンピューティングに基づく演算機能を含むものであってもよい。

#### 【0179】

プロセッサ71は、コンピュータ7の内部構成の各装置等から入力されたデータやソフトウェアに基づいて演算処理を行ってもよく、演算結果や制御信号を各装置等に出力してもよい。プロセッサ71は、コンピュータ7のOS（Operating System）や、アプリケーション等を実行することにより、コンピュータ7を構成する各構成要素を制御してもよい。

40

#### 【0180】

前述した実施形態における各装置（情報処理装置）は、1又は複数のプロセッサ71により実現されてもよい。ここで、プロセッサ71は、1チップ上に配置された1又は複数の電子回路を指してもよいし、2つ以上のチップあるいは2つ以上のデバイス上に配置された1又は複数の電子回路を指してもよい。複数の電子回路を用いる場合、各電子回路は有線又は無線により通信してもよい。

50

**【 0 1 8 1 】**

主記憶装置72は、プロセッサ71が実行する命令及び各種データ等を記憶してもよく、主記憶装置72に記憶された情報がプロセッサ71により読み出されてもよい。補助記憶装置73は、主記憶装置72以外の記憶装置である。なお、これらの記憶装置は、電子情報を格納可能な任意の電子部品を意味するものとし、半導体のメモリでもよい。半導体のメモリは、揮発性メモリ又は不揮発性メモリのいずれでもよい。前述した実施形態における各装置（情報処理装置）において各種データ等を保存するための記憶装置は、主記憶装置72又は補助記憶装置73により実現されてもよく、プロセッサ71に内蔵される内蔵メモリにより実現されてもよい。例えば、前述した実施形態における記憶部102は、主記憶装置72又は補助記憶装置73により実現されてもよい。

10

**【 0 1 8 2 】**

前述した実施形態における各装置（情報処理装置）が、少なくとも1つの記憶装置（メモリ）と、この少なくとも1つの記憶装置に接続（結合）される少なくとも1つのプロセッサで構成される場合、記憶装置1つに対して、少なくとも1つのプロセッサが接続されてもよい。また、プロセッサ1つに対して、少なくとも1つの記憶装置が接続されてもよい。また、複数のプロセッサのうち少なくとも1つのプロセッサが、複数の記憶装置のうち少なくとも1つの記憶装置に接続される構成を含んでもよい。また、複数台のコンピュータに含まれる記憶装置とプロセッサによって、この構成が実現されてもよい。さらに、記憶装置がプロセッサと一体になっている構成（例えば、L1 キャッシュ、L2 キャッシュを含むキャッシュメモリ）を含んでもよい。

20

**【 0 1 8 3 】**

ネットワークインターフェース74は、無線又は有線により、通信ネットワーク8に接続するためのインターフェースである。ネットワークインターフェース74は、既存の通信規格に適合したもの等、適切なインターフェースを用いればよい。ネットワークインターフェース74により、通信ネットワーク8を介して接続された外部装置9Aと情報のやり取りが行われてもよい。なお、通信ネットワーク8は、WAN (Wide Area Network)、LAN (Local Area Network)、PAN (Personal Area Network) 等のいずれか、又は、それらの組み合わせであってよく、コンピュータ7と外部装置9Aとの間で情報のやりとりが行われるものであればよい。WANの一例としてインターネット等があり、LANの一例として IEEE 802.11 やイーサネット（登録商標）等があり、PANの一例として Bluetooth（登録商標）や NFC（Near Field Communication）等がある。

30

**【 0 1 8 4 】**

デバイスインターフェース75は、外部装置9Bと直接接続するUSB等のインターフェースである。

**【 0 1 8 5 】**

外部装置9Aは、コンピュータ7とネットワークを介して接続されている装置である。  
外部装置9Bは、コンピュータ7と直接接続されている装置である。

**【 0 1 8 6 】**

外部装置9A又は外部装置9Bは、一例として、入力装置であってもよい。入力装置は、例えば、カメラ、マイクロフォン、モーションキャプチャ、各種センサ等、キーボード、マウス又はタッチパネル等のデバイスであり、取得した情報をコンピュータ7に与える。また、パーソナルコンピュータ、タブレット端末又はスマートフォン等の入力部とメモリとプロセッサを備えるデバイスであってもよい。

40

**【 0 1 8 7 】**

また、外部装置9A又は外部装置9Bは、一例として、出力装置でもよい。出力装置は、例えば、LCD (Liquid Crystal Display)、有機EL (Electro Luminescence) パネル等の表示装置であってもよいし、音声等を出力するスピーカ等であってもよい。また、パーソナルコンピュータ、タブレット端末又はスマートフォン等の出力部とメモリとプロセッサを備えるデバイスであってもよい。

**【 0 1 8 8 】**

50

また、外部装置9A又は外部装置9Bは、記憶装置（メモリ）であってもよい。例えば、外部装置9Aは、ネットワークストレージ等であってもよく、外部装置9Bは、HDD等のストレージであってもよい。

#### 【0189】

また、外部装置9A又は外部装置9Bは、前述した実施形態における各装置（情報処理装置）の構成要素の一部の機能を有する装置でもよい。つまり、コンピュータ7は、外部装置9A又は外部装置9Bに処理結果の一部又は全部を送信してもよいし、外部装置9A又は外部装置9Bから処理結果の一部又は全部を受信してもよい。

#### 【0190】

本明細書（請求項を含む）において、「a、b及びcの少なくとも1つ（一方）」又は「a、b又はcの少なくとも1つ（一方）」の表現（同様な表現を含む）が用いられる場合は、a、b、c、a-b、a-c、b-c又はa-b-cのいずれかを含む。また、a-a、a-b-b、a-a-b-b-c-c等のように、いずれかの要素について複数のインスタンスを含んでもよい。さらに、a-b-c-dのようにdを有する等、列挙された要素（a、b及びc）以外の他の要素を加えることも含む。10

#### 【0191】

本明細書（請求項を含む）において、「データを入力として／を用いて／データに基づいて／に従って／に応じて」等の表現（同様な表現を含む）が用いられる場合は、特に断りがない場合、データそのものを用いる場合や、データに何らかの処理を行ったもの（例えば、ノイズ加算したもの、正規化したもの、データから抽出した特徴量、データの中間表現等）を用いる場合を含む。また、「データを入力として／を用いて／データに基づいて／に従って／に応じて」何らかの結果が得られる旨が記載されている場合（同様な表現を含む）、特に断りが無い場合、当該データのみに基づいて当該結果が得られる場合や、当該データ以外の他のデータ、要因、条件及び／又は状態等にも影響を受けて当該結果が得られる場合を含む。また、「データを出力する」旨が記載されている場合（同様な表現を含む）、特に断りがない場合、データそのものを出力として用いる場合や、データに何らかの処理を行ったもの（例えば、ノイズ加算したもの、正規化したもの、データから抽出した特徴量、データの中間表現等）を出力として用いる場合を含む。20

#### 【0192】

本明細書（請求項を含む）において、「接続される（connected）」及び「結合される（coupled）」との用語が用いられる場合は、直接的な接続／結合、間接的な接続／結合、電気的（electrically）な接続／結合、通信的（communicatively）な接続／結合、機能的（operatively）な接続／結合、物理的（physically）な接続／結合等のいずれをも含む非限定的な用語として意図される。当該用語は、当該用語が用いられた文脈に応じて適宜解釈されるべきであるが、意図的に或いは当然に排除されるのではない接続／結合形態は、当該用語に含まれるものとして非限定的に解釈されるべきである。30

#### 【0193】

本明細書（請求項を含む）において、「AがBするよう構成される（A configured to B）」との表現が用いられる場合は、要素Aの物理的構造が、動作Bを実行可能な構成を有するとともに、要素Aの恒常的（permanent）又は一時的（temporary）な設定（setting / configuration）が、動作Bを実際に実行するように設定（configured / set）されていることを含んでよい。例えば、要素Aが汎用プロセッサである場合、当該プロセッサが動作Bを実行可能なハードウェア構成を有するとともに、恒常的（permanent）又は一時的（temporary）なプログラム（命令）の設定により、動作Bを実際に実行するように設定（configured）されていればよい。また、要素Aが専用プロセッサ又は専用演算回路等である場合、制御用命令及びデータが実際に付属しているか否かとは無関係に、当該プロセッサの回路的構造等が動作Bを実際に実行するように構築（implemented）されていればよい。40

#### 【0194】

本明細書（請求項を含む）において、含有又は所有を意味する用語（例えば、「含む（

50

comprising / including)」及び「有する (having)」等)が用いられる場合は、当該用語の目的語により示される対象物以外の物を含有又は所有する場合を含む、open-endedな用語として意図される。これらの含有又は所有を意味する用語の目的語が数量を指定しない又は単数を示唆する表現 (a 又は an を冠詞とする表現) である場合は、当該表現は特定の数に限定されないものとして解釈されるべきである。

#### 【0195】

本明細書 (請求項を含む)において、ある箇所において「1つ又は複数 (one or more)」又は「少なくとも1つ (at least one)」等の表現が用いられ、他の箇所において数量を指定しない又は単数を示唆する表現 (a 又は an を冠詞とする表現) が用いられているとしても、後者の表現が「1つ」を意味することを意図しない。一般に、数量を指定しない又は単数を示唆する表現 (a 又は an を冠詞とする表現) は、必ずしも特定の数に限定されないものとして解釈されるべきである。

10

#### 【0196】

本明細書において、ある実施形態の有する特定の構成について特定の効果 (advantage / result) が得られる旨が記載されている場合、別段の理由がない限り、当該構成を有する他の1つ又は複数の実施形態についても当該効果が得られると理解されるべきである。但し、当該効果の有無は、一般に種々の要因、条件及び / 又は状態等に依存し、当該構成により必ず当該効果が得られるものではないと理解されるべきである。当該効果は、種々の要因、条件及び / 又は状態等が満たされたときに実施形態に記載の当該構成により得られるものに過ぎず、当該構成又は類似の構成を規定したクレームに係る発明において、当該効果が必ずしも得られるものではない。

20

#### 【0197】

本明細書 (請求項を含む)において、「最大化する (maximize) / 最大化 (maximization)」等の用語が用いられる場合は、グローバルな最大値を求めること、グローバルな最大値の近似値を求めること、ローカルな最大値を求めること及びローカルな最大値の近似値を求めることを含み、当該用語が用いられた文脈に応じて適宜解釈されるべきである。また、これら最大値の近似値を確率的又はヒューリスティックに求めることを含む。同様に、「最小化する (minimize) / 最小化 (minimization)」等の用語が用いられる場合は、グローバルな最小値を求めること、グローバルな最小値の近似値を求めること、ローカルな最小値を求めること及びローカルな最小値の近似値を求めることを含み、当該用語が用いられた文脈に応じて適宜解釈されるべきである。また、これら最小値の近似値を確率的又はヒューリスティックに求めることを含む。同様に、「最適化する (optimize) / 最適化 (optimization)」等の用語が用いられる場合は、グローバルな最適値を求めること、グローバルな最適値の近似値を求めること、ローカルな最適値を求めること、及びローカルな最適値の近似値を求めることを含み、当該用語が用いられた文脈に応じて適宜解釈されるべきである。また、これら最適値の近似値を確率的又はヒューリスティックに求めることを含む。

30

#### 【0198】

本明細書 (請求項を含む)において、複数のハードウェアが所定の処理を行う場合、各ハードウェアが協働して所定の処理を行ってもよいし、一部のハードウェアが所定の処理の全てを行ってもよい。また、一部のハードウェアが所定の処理の一部を行い、別のハードウェアが所定の処理の残りを行ってもよい。本明細書 (請求項を含む)において、「1又は複数のハードウェアが第1処理を行い、前記1又は複数のハードウェアが第2処理を行う」等の表現 (同様な表現を含む) が用いられている場合、第1処理を行うハードウェアと第2処理を行うハードウェアは同じものであってもよいし、異なるものであってもよい。つまり、第1処理を行うハードウェア及び第2処理を行うハードウェアが、前記1又は複数のハードウェアに含まれていればよい。なお、ハードウェアは、電子回路、又は、電子回路を含む装置等を含んでもよい。

40

#### 【0199】

本明細書 (請求項を含む)において、複数の記憶装置 (メモリ) がデータの記憶を行う

50

場合、複数の記憶装置のうち個々の記憶装置は、データの一部のみを記憶してもよいし、データの全体を記憶してもよい。また、複数の記憶装置のうち一部の記憶装置がデータを記憶する構成を含んでもよい。

### 【0200】

以上、本開示の実施形態について詳述したが、本開示は上記した個々の実施形態に限定されるものではない。特許請求の範囲に規定された内容及びその均等物から導き出される本開示の概念的な思想と趣旨を逸脱しない範囲において、種々の追加、変更、置き換え及び部分的削除等が可能である。例えば、前述した実施形態において、数値又は数式を説明に用いている場合、これらは例示的な目的で示されたものであり、本開示の範囲を限定するものではない。また、実施形態で示した各動作の順序も、例示的なものであり、本開示の範囲を限定するものではない。

10

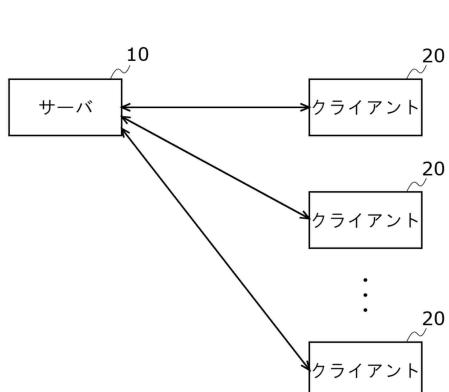
### 【符号の説明】

#### 【0201】

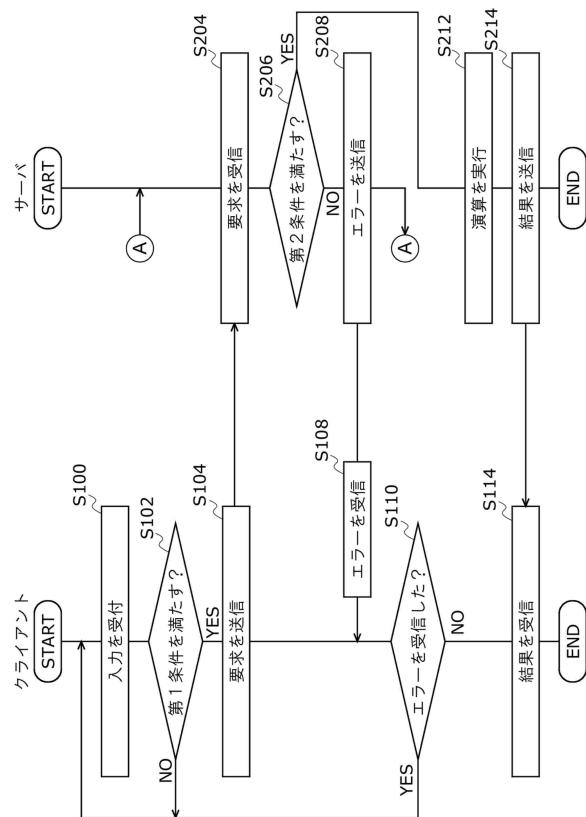
- 1: 情報処理システム、
- 10: サーバ、
- 20: クライアント、
- 30: ゲートウェイ

### 【図面】

#### 【図1】



#### 【図2】



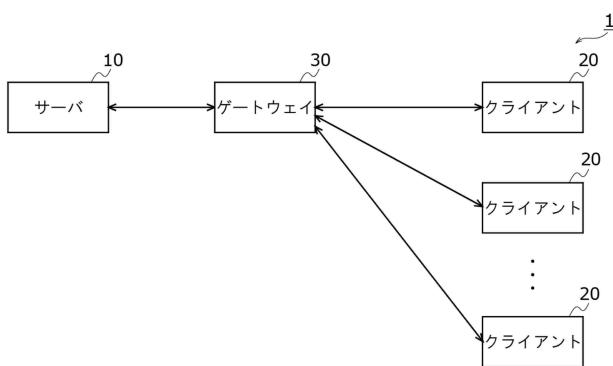
20

30

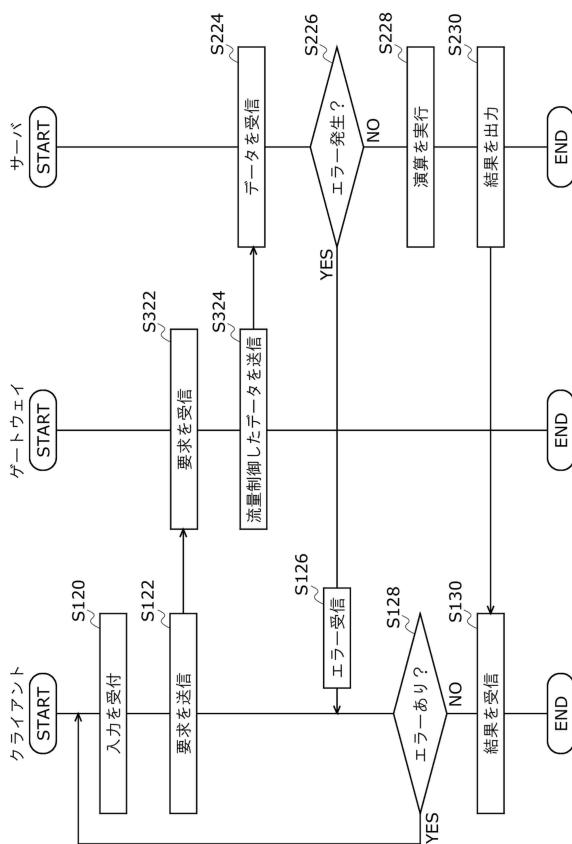
40

50

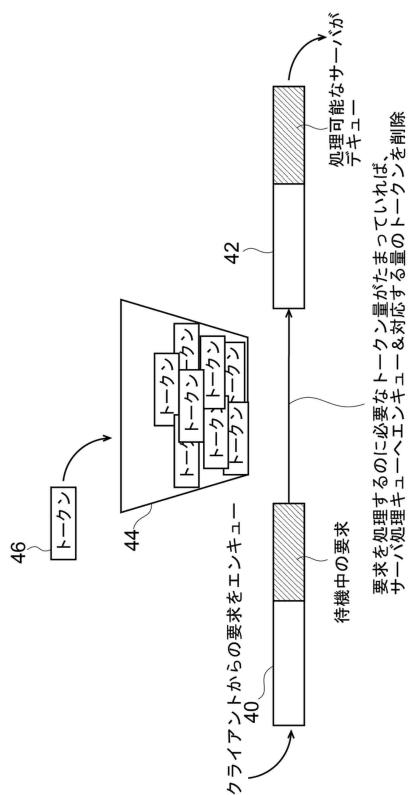
【図3】



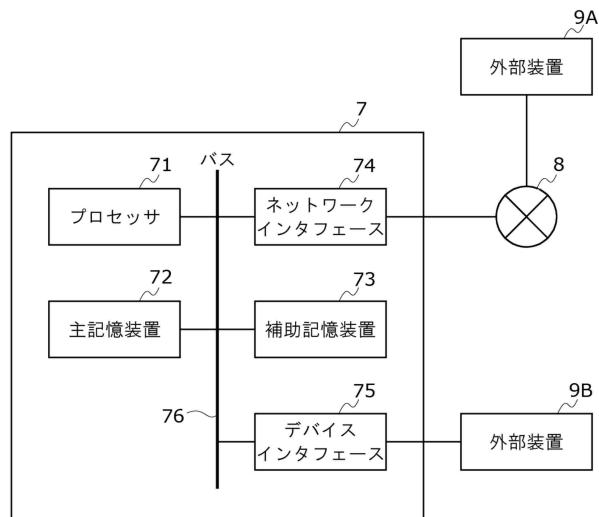
【図4】



【図5】



【図6】



10

20

30

40

50

---

フロントページの続き

(72)発明者 小方 拓也

東京都千代田区大手町1丁目6番1号 大手町ビル 株式会社 Preferred Network  
s 内

(72)発明者 坪井 祐太

東京都千代田区大手町1丁目6番1号 大手町ビル 株式会社 Preferred Network  
s 内