

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7622865号
(P7622865)

(45)発行日 令和7年1月28日(2025.1.28)

(24)登録日 令和7年1月20日(2025.1.20)

(51)国際特許分類 F I
G 0 1 N 35/00 (2006.01) G 0 1 N 35/00 E

請求項の数 8 (全30頁)

(21)出願番号	特願2023-552756(P2023-552756)	(73)特許権者	000001993 株式会社島津製作所 京都府京都市中京区西ノ京桑原町1番地
(86)(22)出願日	令和4年9月6日(2022.9.6)	(74)代理人	110001195 弁理士法人深見特許事務所
(86)国際出願番号	PCT/JP2022/033379	(72)発明者	倉田 星哉 京都府京都市中京区西ノ京桑原町1番地 株式会社島津製作所内
(87)国際公開番号	WO2023/058384	(72)発明者	池上 将弘 京都府京都市中京区西ノ京桑原町1番地 株式会社島津製作所内
(87)国際公開日	令和5年4月13日(2023.4.13)	(72)発明者	伴野 太一 京都府京都市中京区西ノ京桑原町1番地 株式会社島津製作所内
審査請求日	令和6年3月8日(2024.3.8)	審査官	野田 華代
(31)優先権主張番号	特願2021-164631(P2021-164631)		
(32)優先日	令和3年10月6日(2021.10.6)		
(33)優先権主張国・地域又は機関	日本国(JP)		

最終頁に続く

(54)【発明の名称】 制御装置および方法

(57)【特許請求の範囲】

【請求項1】

実験プロトコルに基づいて実験装置を制御する制御装置であって、
実験の処理に対応する複数のノードを含む有向グラフで設計された実験プロトコルの入力を受け付ける入力部と、

第1ノードおよび前記第1ノードの処理の完了によって実行可能となる第2ノードが前記複数のノードに含まれる場合、前記第2ノードの処理の開始が前記第1ノードの処理の完了に依存することを示す第1依存情報を含む依存関係リストを生成する生成部と、

前記複数のノードの中から処理を開始させるノードを前記依存関係リストに基づいて選択する選択部と、

前記選択部により選択されたノードの処理を実験装置に指令する指令部と、

前記複数のノードの処理のうちのいずれかが完了した場合に、前記依存関係リストまたは前記複数のノードの状態を更新する更新部とを備え、

前記選択部は、前記第1ノードの処理が完了した場合、前記第2ノードを選択する、制御装置。

【請求項2】

前記更新部は、前記第1ノードの処理が完了した場合、前記第1依存情報を第1情報から第2情報に更新し、

前記選択部は、前記第1依存情報が前記第1情報から第2情報に更新された場合、前記第2ノードを選択する、請求項1に記載の制御装置。

【請求項 3】

前記生成部は、前記複数のノードの各々のステータスを特定するためのステータスリストをさらに生成し、

前記更新部は、前記第 1 ノードの処理が完了した場合、前記ステータスリストにおける前記第 1 ノードのステータスを実行中から完了に更新し、前記第 2 ノードの処理を開始する場合、前記ステータスリストにおける前記第 2 ノードのステータスを実行待ちから実行中に更新する、請求項 2 に記載の制御装置。

【請求項 4】

前記複数のノードは、前記第 1 ノードの処理の完了によって実行可能となる第 3 ノードを含み、

前記依存関係リストは、前記第 3 ノードの処理の開始が前記第 1 ノードの処理の完了に依存することを示す第 2 依存情報を含み、

前記更新部は、前記第 1 ノードの処理が完了した場合、前記第 1 依存情報および前記第 2 依存情報を前記第 1 情報から前記第 2 情報に更新し、

前記選択部は、前記第 1 依存情報および前記第 2 依存情報が前記第 1 情報から第 2 情報に更新された場合、前記第 2 ノードおよび前記第 3 ノードを選択する、請求項 3 に記載の制御装置。

【請求項 5】

前記複数のノードは、前記第 2 ノードおよび前記第 3 ノードの処理が完了した後に、処理を開始することが可能となる第 4 ノードを含み、

前記依存関係リストは、前記第 4 ノードの処理の開始が前記第 2 ノードの処理の完了に依存することを示す第 3 依存情報と、前記第 4 ノードの処理の開始が前記第 3 ノードの処理の完了に依存することを示す第 4 依存情報とを含み、

前記更新部は、前記第 2 ノードの処理が完了した場合、前記第 3 依存情報を前記第 1 情報から前記第 2 情報に更新し、前記第 3 ノードの処理が完了した場合、前記第 4 依存情報を前記第 1 情報から前記第 2 情報に更新し、

前記選択部は、前記第 3 依存情報および前記第 4 依存情報が前記第 1 情報から第 2 情報に更新された場合、前記第 4 ノードを選択する、請求項 4 に記載の制御装置。

【請求項 6】

前記依存関係リストを表示するための信号を出力する出力部をさらに備える、請求項 1 に記載の制御装置。

【請求項 7】

前記入力部に入力された前記実験プロトコルが、サンプルを収容する容器に対応するノードを含む第 1 有向グラフで設計されている場合、前記入力部に入力された前記実験プロトコルを、前記容器を移動する装置の動作を含む第 2 有向グラフに変換する変換部をさらに備え、

前記生成部は、前記第 2 有向グラフに基づいて前記依存関係リストを生成する、請求項 1 に記載の制御装置。

【請求項 8】

実験プロトコルに基づいて実験装置を制御する方法であって、

実験の処理に対応する複数のノードを含む有向グラフで設計された実験プロトコルの入力を受け付けるステップと、

第 1 ノードおよび前記第 1 ノードの処理の完了によって実行可能となる第 2 ノードが前記複数のノードに含まれる場合、前記第 2 ノードの処理の開始が前記第 1 ノードの処理の完了に依存することを示す第 1 依存情報を含む依存関係リストを生成するステップと、

前記複数のノードの中から処理を開始させるノードを前記依存関係リストに基づいて選択するステップと、

選択されたノードの処理を実験装置に指令するステップと、

前記複数のノードの処理のうちのいずれかが完了した場合に、前記依存関係リストまたは前記複数のノードの状態を更新するステップとを含み、

10

20

30

40

50

前記処理を開始させるノードを選択するステップは、前記第1ノードの処理が完了した場合、前記第2ノードを選択する、方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、制御装置および方法に関し、特に、実験プロトコルに基づいて実験装置を制御する制御装置および実験プロトコルに基づいて実験装置を制御する方法に関する。

【背景技術】

【0002】

従来、ユーザがコンピュータ上で設計した実験プロトコルに従って実験を行う構成が知られている。たとえば、国際公開第2016/208623号(特許文献1)に開示されているシステムは、実験プロトコルに関する情報を含むデータベースから実験の連鎖が網目状に形成されたグラフを取得して表示する。

10

【先行技術文献】

【特許文献】

【0003】

【文献】国際公開第2016/208623号

【発明の概要】

【発明が解決しようとする課題】

【0004】

実験プロトコルに基づいた処理の自動化を目指す場合、実験プロトコルに基づいた一つひとつの処理を制御装置がシーケンシャルに実行する手法を採用することが考えられる。このような手法では、1つの処理の完了を待って、次の処理に進むため、制御装置の制御を単純化することができる。たとえば、制御装置は、実験プロトコルに含まれる複数の処理を1次元の配列構造を持った命令集として順次実行していただくだけでよい。

20

【0005】

ところで、複数の処理を同時に実行することで実験効率を向上させることができる実験プロトコルも存在する。たとえば、リキッドハンドラで試料を作成している間に、別の試料を遠心分離機で分離するなど、多数の試料を複数の装置で同時に分析することで、より多くの試料を作成および分析することができる場合がある。

30

【0006】

このような並列処理を含む実験プロトコルは、有向グラフを用いることによって比較的容易に作成することができる。有向グラフでは、実験装置の処理などをノードで表現し、複数の処理の前後関係をエッジで表現する。エッジは、先に処理するノードから後に処理するノードへと向かう。並列処理を有向グラフで表現するためには、たとえば、あるノードと複数のノードとをエッジで接続すればよい。

【0007】

しかし、並列処理が含まれる実験プロトコルを制御装置に実行させることは容易ではない。1つの処理の完了を待って、次の処理に進むような制御を繰り返す制御装置では、1つの処理から枝分かれして並行して進む複数の処理の順序などを管理することができない。

40

【0008】

本発明は、このような課題を解決するためになされたものであって、その目的は、有向グラフで作成された実験プロトコルに基づいた実験効率を高めることである。

【課題を解決するための手段】

【0009】

本発明の一局面に係る制御装置は、実験プロトコルに基づいて実験装置を制御する制御装置である。制御装置は、実験の処理に対応する複数のノードを含む有向グラフで設計された実験プロトコルの入力を受け付ける入力部と、第1ノードおよび第1ノードの処理の完了によって実行可能となる第2ノードが複数のノードに含まれる場合、第2ノードの処理の開始が第1ノードの処理の完了に依存することを示す第1依存情報を含む依存関係リ

50

ストを生成する生成部と、複数のノードの中から処理を開始させるノードを依存関係リストに基づいて選択する選択部と、選択部により選択されたノードの処理を実験装置に指令する指令部と、複数のノードの処理のうちいずれかが完了した場合に、依存関係リストまたは複数のノードの状態を更新する更新部とを備え、選択部は、第1ノードの処理が完了した場合、第2ノードを選択する。

【0010】

本発明の他の局面に係る方法は、実験プロトコルに基づいて実験装置を制御する方法である。方法は、実験の処理に対応する複数のノードを含む有向グラフで設計された実験プロトコルの入力を受け付けるステップと、第1ノードおよび第1ノードの処理の完了によって実行可能となる第2ノードが複数のノードに含まれる場合、第2ノードの処理の開始が第1ノードの処理の完了に依存することを示す第1依存情報を含む依存関係リストを生成するステップと、複数のノードの中から処理を開始させるノードを依存関係リストに基づいて選択するステップと、選択部により選択されたノードの処理を実験装置に指令するステップと、複数のノードの処理のうちいずれかが完了した場合に、依存関係リストまたは複数のノードの状態を更新するステップとを含み、処理を開始させるノードを選択するステップは、第1ノードの処理が完了した場合、第2ノードを選択する。

10

【発明の効果】

【0011】

本開示によれば、有向グラフで作成された実験プロトコルに基づいた実験効率を高めることができる。

20

【図面の簡単な説明】

【0012】

【図1】自動実験管理システム1000の構成を示すブロック図である。

【図2】端末装置400のハードウェア構成を示すブロック図である。

【図3】実験プロトコル設計アプリケーション500のGUI構成を示す図である。

【図4】自動実験システムウィンドウにおいてある処理が選択された様子を示す図である。

【図5】選択された処理に対応する処理ノードがプロトコル設計ウィンドウに追加された様子を示す図である。

【図6】実験プロトコルの設計例である有向グラフDG1を示す図である。

【図7】ロボットアーム121の動作ノードを含む有向グラフDG2を示す図である。

30

【図8】有向グラフDG3を示す図である。

【図9】有向グラフDG3に基づいて制御装置110が作成するイベントハンドラを示す図である。

【図10】有向グラフDG3に基づいて制御装置110が作成する隣接行列リストを示す図である。

【図11】有向グラフDG3に基づいた処理の進行に応じてイベントハンドラが更新される様子を示す図である。

【図12】有向グラフDG3に基づいた処理の進行に応じて隣接行列リストが更新される様子を示す図である。

【図13】実験プロトコルに基づく自動実験の流れを説明するフローチャートである。

40

【図14】実験プロトコルに応じて制御装置110がイベントハンドラおよび隣接行列リストを生成する処理を示すフローチャートである。

【図15】制御装置110がイベントハンドラおよび隣接行列リストに基づいて実験プロトコルを実行する処理を示すフローチャートである。

【図16】制御装置110の機能に関わる構成を説明するための自動実験管理システム1000の構成を示すブロック図である。

【図17】サーバ装置200のハードウェア構成を示すブロック図である。

【図18】変形例1に係る自動実験管理システム1100の構成を示すブロック図である。

【図19】変形例1に関わる端末装置400Aのハードウェア構成を示すブロック図である。

50

【図 2 0】変形例 2 に係る自動実験システム 1 B の構成を示すブロック図である。

【図 2 1】制御装置 1 1 0 B のハードウェア構成を示すブロック図である。

【図 2 2】有向グラフ D G 4 を示す図である。

【図 2 3】有向グラフ D G 4 および有向グラフ D G 4 に対してロボットアーム 1 2 1 のノードを追加した有向グラフ D G 5 を示す図である。

【発明を実施するための形態】

【0013】

以下に、実施の形態について図面を参照して詳細に説明する。なお、以下では図中の同一または相当部分には同一符号を付してその説明は原則的に繰り返さない。

【0014】

図 1 は、自動実験管理システム 1 0 0 0 の構成を示すブロック図である。図 1 に示されるように、自動実験管理システム 1 0 0 0 は、自動実験システム 1 と、サーバ装置 2 0 0 と、データベース 3 0 0 と、端末装置 4 0 0 とを備える。データベース 3 0 0 は、サーバ装置 2 0 0 に接続されている。データベース 3 0 0 には、たとえば自動実験システム 1 に関する情報、サンプルに関する情報、実験プロトコル、および実験プロトコルの実行による出力データ（実験結果）等が登録される。

【0015】

端末装置 4 0 0 は、入出力部 4 3 0 を含む。入出力部 4 3 0 は、ディスプレイ 4 3 1 と、キーボード 4 3 2 と、タッチパッド 4 3 3 とを含む。端末装置 4 0 0 は、たとえば、ノートパソコン、パーソナルコンピュータ、スマートフォン、およびタブレットである。自動実験システム 1、サーバ装置 2 0 0、および端末装置 4 0 0 は、ネットワーク NW を介して互いに接続されている。ネットワーク NW は、たとえばインターネット、WAN (Wide Area Network)、または LAN (Local Area Network) を含む。なお、ネットワーク NW に接続されている端末装置 4 0 0 は 2 つ以上であってもよいし、自動実験システムは 2 つ以上であってもよい。

【0016】

サーバ装置 2 0 0 は、実験プロトコル設計アプリケーション 5 0 0 (特定アプリケーション) を Web アプリケーションとして端末装置 4 0 0 に提供する。実験プロトコル設計アプリケーション 5 0 0 は、端末装置 4 0 0 において Web ブラウザ 6 0 0 を介してディスプレイ 4 3 1 に表示される。キーボード 4 3 2 およびタッチパッド 4 3 3 は、ユーザによる実験プロトコル設計アプリケーション 5 0 0 への GUI 操作を受け付ける。すなわち、端末装置 4 0 0 のユーザは、キーボード 4 3 2 およびタッチパッド 4 3 3 を介する GUI 操作によって、実験プロトコル設計アプリケーション 5 0 0 において自動実験システムを選択して、自動実験システムによって実行される実験プロトコルを設計する。

【0017】

実験プロトコルにおいては、ユーザによって選択された自動実験システム 1 に含まれる少なくとも 1 つの実験装置の処理順序が規定されている。端末装置 4 0 0 は、ユーザによって設計された実験プロトコルをサーバ装置 2 0 0 に送信する。サーバ装置 2 0 0 は、ユーザによって指定された自動実験システム 1 に実験プロトコルを送信する。実験プロトコルを設計する端末装置 4 0 0 と、実験プロトコルを実行する自動実験システム 1 との間にサーバ装置 2 0 0 を介在させることにより、複数の端末装置 4 0 0 および複数の自動実験システム 1 をサーバ装置 2 0 0 によって一括管理することができる。

【0018】

自動実験システム 1 は、制御装置 1 1 0 と、複数の実験装置 1 2 0 とを含む。制御装置 1 1 0 は、複数の実験装置 1 2 0 を制御して、サーバ装置 2 0 0 からの実験プロトコルを自動的に実行する。複数の実験装置 1 2 0 は、ロボットアーム 1 2 1 と、インキュベータ 1 2 2 と、リキッドハンドラ 1 2 3 と、マイクロプレートリーダー 1 2 4 と、遠心分離機 1 2 5 と、液体クロマトグラフ質量分析装置 (LCMS: Liquid Chromatograph Mass Spectrometer) 1 2 6 とを含む。自動実験システムに含まれる実験装置は 1 つであってもよい。

10

20

30

40

50

【 0 0 1 9 】

制御装置 1 1 0 は、制御部としてのプロセッサ 1 1 1 と、記憶部としてのメモリ 1 1 2 とを含む。

【 0 0 2 0 】

プロセッサ 1 1 1 は、典型的には、CPU (Central Processing Unit) または MPU (Multi-Processing Unit) などの演算処理部である。プロセッサ 1 1 1 は、メモリ 1 1 2 に記憶されたプログラムを読み出して実行することで、実験プロトコルを実行するための処理を実現する。

【 0 0 2 1 】

メモリ 1 1 2 は、RAM (Random Access Memory)、ROM (Read Only Memory) およびフラッシュメモリなどの不揮発性メモリによって実現される。メモリ 1 1 2 は、プロセッサ 1 1 1 が可読可能な形式で非一時的にプログラムを記録することができれば、CD-ROM (Compact Disc - Read Only Memory)、DVD-ROM (Digital Versatile Disk - Read Only Memory)、およびUSB (Universal Serial Bus) などにより構成してもよい。

10

【 0 0 2 2 】

ロボットアーム 1 2 1 は、実験プロトコルに規定された複数の処理の順序に従って、サンプルを収容する容器であるプレート P 1 t 1 または P 1 t 2 を実験装置へ移動する。プレート P 1 t 1、P 1 t 2 の各々は、たとえば培養された大腸菌を含む寒天を収容する。インキュベータ 1 2 2 は、温度管理を行いながら細胞を培養する。リキッドハンドラ 1 2 3 は、自動的に、複数のマイクロプレート (ウェル) の各々へ一定量ずつサンプルを分配 (分注) する。マイクロプレートリーダー 1 2 4 は、マイクロプレート内のサンプルの光学的性質を測定する。マイクロプレートリーダー 1 2 4 は、たとえば吸光度および蛍光強度を測定する。遠心分離機 1 2 5 は、遠心力によってサンプルの成分を分離する。LCMS 1 2 6 は、液体クロマトグラフによって分離されたサンプルの成分を質量電荷比 (m/z) 毎に分離する質量分析を行う。

20

【 0 0 2 3 】

図 2 は、端末装置 4 0 0 のハードウェア構成を示すブロック図である。図 2 に示されるように、端末装置 4 0 0 は、プロセッサ 4 2 1 と、記憶部としてのメモリ 4 2 2 およびハードディスク 4 2 3 と、通信インターフェース 4 2 4 と、入出力部 4 3 0 とを含む。これらは、バス 4 4 0 を介して相互に通信可能に接続されている。

30

【 0 0 2 4 】

ハードディスク 4 2 3 は、不揮発性の記憶装置である。ハードディスク 4 2 3 には、たとえばオペレーティングシステム (OS: Operating System) のプログラム 4 1、および Web ブラウザのプログラム 4 2 が保存されている。図 2 に示されるデータ以外にも、ハードディスク 4 2 3 には、たとえば各種アプリケーションの設定および出力が保存される。メモリ 4 2 2 は、揮発性の記憶装置であり、たとえばDRAM (Dynamic Random Access Memory) を含む。

【 0 0 2 5 】

プロセッサ 4 2 1 は、CPU (Central Processing Unit) を含む。プロセッサ 4 2 1 は、ハードディスク 4 2 3 に保存されているプログラムをメモリ 4 2 2 に読み込んで実行する。プロセッサ 4 2 1 は、通信インターフェース 4 2 4 を介してネットワークNWに接続する。

40

【 0 0 2 6 】

図 3 は、実験プロトコル設計アプリケーション 5 0 0 のGUI構成を示す図である。図 3 に示されるように、実験プロトコル設計アプリケーション 5 0 0 は、キュリストウィンドウ 5 1 0 と、プロトコルリストウィンドウ 5 2 0 と、プロトコル設計ウィンドウ 5 3 0 と、自動実験システムウィンドウ 5 4 0 と、サンプル容器ウィンドウ 5 5 0 と、ツールウィンドウ 5 6 0 と、選択カーソル Cr とを含む。

【 0 0 2 7 】

50

キューリストウィンドウ 5 1 0 には、複数のプロトコルが順序付けられたキューが表示される。図 3 においては、キュー q 1 , q 2 がキューリストウィンドウ 5 1 0 に表示されている。プロトコルリストウィンドウ 5 2 0 には、実験プロトコルが表示される。図 3 においては、実験プロトコル p 1 がプロトコルリストウィンドウ 5 2 0 に表示され、選択されている。

【 0 0 2 8 】

プロトコル設計ウィンドウ 5 3 0 においては、実験プロトコルが有向グラフで設計される。有向グラフにおいては、複数のノード間の接続関係がエッジとして規定される。有向グラフは、予め定められた構造化データフォーマットに従って、グラフ構造データとして保存される。構造化データフォーマットとしては、たとえば X M L (Extensible Markup Language) または J s o n (JavaScript (登録商標) Object Notation) を挙げることができる。有向グラフの頂点として選択可能な複数のノードは、 G U I として形成され、容器ノード、処理ノード、およびデータノードを含む。容器ノードは、サンプルを収容する容器に対応するノードである。処理ノードは、自動実験システムに含まれる装置の各々の処理に対応するノードである。データノードは、実験装置の処理の出力データに対応するノードである。

10

【 0 0 2 9 】

プロトコル設計ウィンドウ 5 3 0 は、容器領域 5 3 1、処理領域 5 3 2、およびデータ領域 5 3 3 に分けられている。ある実験プロトコルの設計を開始する初期状態において処理領域 5 3 2 には、実験プロトコルの開始を表すスタートノード M s、実験プロトコルの終了を表すエンドノード M e、およびスタートノード M s からエンドノード M e へ向かうエッジ E 1 0 が表示されている。

20

【 0 0 3 0 】

自動実験システムウィンドウ 5 4 0 には、ユーザによって選択された自動実験システムに含まれる少なくとも 1 つの実験装置の各々によって実行可能な処理が表示される。図 3 においては、自動実験システム 1 が選択されている。ロボットアーム 1 2 1 によって実行可能な処理として、「容器の搬送」が表示されている。インキュベータ 1 2 2 によって実行可能な処理として、「細胞の培養」が表示されている。リキッドハンドラ 1 2 3 によって実行可能な処理として、「液体の分注」が表示されている。マイクロプレートリーダー 1 2 4 によって実行可能な処理として、「吸光度測定」および「蛍光強度測定」が表示されている。遠心分離機 1 2 5 によって実行可能な処理として、「遠心分離」が表示されている。 L C M S 1 2 6 によって実行可能な処理として、「質量分析」が表示されている。

30

【 0 0 3 1 】

サンプル容器ウィンドウ 5 5 0 には、サンプルを収容する容器が表示される。図 3 においては、サンプルの一例である大腸菌を収容する容器として、プレート P 1 t 1 , P 1 t 2 が表示されている。

【 0 0 3 2 】

ツールウィンドウ 5 6 0 には、自動実験システム 1 の制御装置 1 1 0 によって行われる特定の処理が表示される。図 3 においては、「特徴量抽出」、「条件分岐」、「反復」、および「タイマ」が表示されている。「特徴量抽出」は、ユーザによって選択されたデータノードに対応するデータからユーザによって指定された特徴量を抽出する処理に対応する。「条件分岐」は、ユーザによって指定された条件の成否に基づいて分岐処理を行う処理に対応する。「反復」は、ユーザによって指定された回数だけ指定された処理を反復する処理に対応する。「タイマ」は、ユーザによって指定された時間の間、実験プロトコルの進行を待機する処理に対応する。

40

【 0 0 3 3 】

図 4 は、自動実験システムウィンドウ 5 4 0 においてある処理が選択された様子を示す図である。図 4 に示されるように、ユーザによって、自動実験システムウィンドウ 5 4 0 において「吸光度測定」が選択され、スタートノード M s とエンドノード M e との間にドラッグされる。

50

【 0 0 3 4 】

図 5 は、選択された処理に対応する処理ノードがプロトコル設計ウィンドウ 5 3 0 に追加された様子を示す図である。図 5 に示されるように、「吸光度測定」に対応する処理ノード M 1 がスタートノード M s とエンドノード M e との間に追加され、選択されている。処理ノード M 1 の追加に伴い、容器ノード C 1 およびデータノード D 1 が自動的に容器領域 5 3 1 およびデータ領域 5 3 3 にそれぞれ自動的に追加される。処理ノード M 1 の選択に伴い、選択されたノードに関する情報を含む情報ウィンドウ 5 7 0 が表示されている。図 5 においては、処理ノード M 1 に対応する吸光度測定のパラメータとして、測定波長および測定対象ウェルが表示されている。

【 0 0 3 5 】

スタートノード M s および処理ノード M 1 は、スタートノード M s から処理ノード M 1 に向かうエッジ E 1 によって接続されている。処理ノード M 1 およびエンドノード M e は、処理ノード M 1 からエンドノード M e に向かうエッジ E 2 によって接続されている。容器ノード C 1 および処理ノード M 1 は、容器ノード C 1 から処理ノード M 1 に向かうエッジ E 3 (第 1 エッジ) によって接続されている。処理ノード M 1 およびデータノード D 1 は、処理ノード M 1 からデータノード D 1 に向かうエッジ E 4 (第 2 エッジ) によって接続されている。エッジ E 3 は、容器ノード C 1 に対応する容器が処理ノード M 1 に対応する処理に入力されることを示す。エッジ E 4 は、処理ノード M 1 に対応する処理の出力データがデータノード D 1 に対応することを示す。処理ノードの追加に伴い、処理ノードに接続された容器ノードおよびデータノードが自動的に追加されることにより、実験プロトコルの設計を効率化することができる。なお、図 5 においては、容器ノード C 1 が対応するサンプル容器が指定されていないため、容器ノード C 1 およびエッジ E 3 が点線で示されている。

【 0 0 3 6 】

以上に説明した手順で、ユーザは、実験プロトコル設計アプリケーション 5 0 0 によって提供されるウィンドウから必要な処理を選択することによって、実験プロトコルを有向グラフで設計することができる。ここで、ユーザによって有向グラフの形式で作成された実験プロトコルの設計例を示す。

【 0 0 3 7 】

図 6 は、実験プロトコルの設計例である有向グラフ D G 1 を示す図である。プロトコル設計ウィンドウ 5 3 0 には、有向グラフ D G 1 が示されている。有向グラフ D G 1 は、スタートノード M s、エンドノード M e、処理ノード M 1 1, M 1 2, M 1 3, M 1 4, M 1 5, M 1 6、容器ノード C 1 1, C 1 2, C 1 3, C 1 4, C 1 5, C 1 6, C 1 7, C 1 8、およびデータノード D 1 1, D 1 2 を含む。

【 0 0 3 8 】

処理ノード M 1 1 および M 1 2 は、「液体の分注」に対応する。処理ノード M 1 3 は、「培養装置への取り込み」に対応する。処理ノード M 1 4 は、「吸光度測定」に対応する。処理ノード M 1 5 は、「遠心分離」に対応する。処理ノード M 1 6 は、「LCMS 分析」に対応する。

【 0 0 3 9 】

容器ノード C 1 1, C 1 6 は、「培養プレート」に対応する。容器ノード C 1 2, C 1 3, C 1 7, C 1 8 は、「測定プレート 1」に対応する。容器ノード C 1 4, C 1 5 は、「測定プレート 2」に対応する。

【 0 0 4 0 】

データノード D 1 1 は、「吸光度データ」に対応する。データノード D 1 2 は、「LCMS データ」に対応する。

【 0 0 4 1 】

各ノードは、エッジによって他のノードに接続される。たとえば、スタートノード M s は、エッジによってノード M 1 1 に接続される。ノード C 1 1 および C 1 2 は、エッジによってノード M 1 1 に接続される。ノード M 1 1 は、エッジによってノード M 1 2 に接続

10

20

30

40

50

される。さらに、ノードM 1 1はエッジによってノードC 1 6に接続される。

【0042】

容器ノードC 1 1, 1 2から処理ノードM 1 1に向かうエッジは、培養プレートと測定プレート1とが所定の位置に搬送されることによって、処理ノードM 1 1に対応する「液体の分注」が実行されることを表す。

【0043】

処理ノードM 1 4からデータノードD 1 1に向かうエッジは、吸光度測定によって吸光度データが得られることを表す。

【0044】

このように、ユーザは、実験プロトコルを設計する段階において、搬送対象のプレート名を容器ノードとして用いるとともに、実験プロトコルによって得られるデータをデータノードとして用いることができる。その結果、ユーザは、実際の実験プロトコルの流れをイメージして直感的に有向グラフを作成することができる。ユーザーは、このような有向グラフを端末装置400を用いて作成することができる。サーバ装置200は、端末装置400で作成された有向グラフを保存する。サーバ装置200は、保存した有向グラフを実験システム1の制御装置110に送信する。

10

【0045】

容器ノードおよびデータノードを含む有向グラフは、人間がそれを見たときに実際の実験プロトコルの流れを直感的に理解できるというメリットを有する。しかし、この有向グラフにはプレートを搬送するロボットアーム121の動作が描かれていない。実験システム1において有向グラフに基づいて自動的に実験プロトコルを実行するためには、ロボットアーム121の動作がノードとして特定される必要がある。そこで、有向グラフを受信した制御装置110は、容器ノードおよびデータノードを含む有向グラフを、ロボットアーム121の動作ノードを含む有向グラフに変換する。

20

【0046】

図7は、ロボットアーム121の動作ノードを含む有向グラフDG2を示す図である。制御装置110は、ユーザによって作成された有向グラフDG1(図6参照)を、図7に示す有向グラフDG2に変換する。有向グラフDG2には、ロボットアーム121の動作ノードに対応するノードR11~R14が含まれる。ノードR11~R14は、図6に示される容器ノードC11~C18をロボットアーム121の動作に変換したものである。

30

【0047】

たとえば、ノードR11は、培養プレートに関するノードC11の処理と、測定プレート1に関するノードC12の処理とを含む。ノードR12は、測定プレート1に関するノードC13の処理と、プレート2に関するノードC14の処理と、培養プレートに関するノードC16の処理とを含む。ノードR13は、測定プレート2に関するノードC15の処理と、測定プレート1に関するノードC17の処理とを含む。ノードR14は、測定プレート1に関するノードC18の処理を含む。

【0048】

このように、制御装置110は、容器ノードおよびデータノードを含む有向グラフを、ロボットアーム121の動作ノードを含む有向グラフに変換する。図7に示される有向グラフDG2の動作ノードR11~R14では、説明を簡単にするため、ロボットアーム121が搬送するプレートおよび搬送先の装置に関する情報の記述を省略している。実際に使用する有向グラフの動作ノードには、これらの情報が含まれる。

40

【0049】

以上、説明したとおり、本実施の形態によれば、ユーザは、実験プロトコルを有向グラフで容易に設計することができる。特に、ユーザは、実験プロトコルを有向グラフで作成することにより、並列処理を含む複雑な実験プロトコルを比較的簡単に設計することができる。

【0050】

本実施の形態に関わる制御装置110によれば、容器ノードおよびデータノードを含む

50

有向グラフが、ロボットアーム 1 2 1 の動作ノードを含む有向グラフに変換される。このため、ユーザは、ロボットアーム 1 2 1 の動作を意識することなく、実際の実験プロトコルの流れをイメージして図 6 に例示するような有向グラフを作成することができる。

【 0 0 5 1 】

また、有向グラフに応じて、複数のノードの処理を同時に進めることによって、実験効率を向上させることができる。たとえば、図 7 に示される有向グラフ DG 2 では、ノード R 1 2 の処理の完了後、ノード M 1 2 とノード M 1 3 とが並列処理され、両ノード M 1 2 および M 1 3 の処理の完了後にノード R 1 3 の処理が実行されることが規定されている。

【 0 0 5 2 】

しかし、並列処理が含まれる実験プロトコルを制御装置 1 1 0 に実行させることは容易ではない。たとえば、1 つのノードの処理が完了することを待って、次のノードの処理を指令するように構成された制御装置 1 1 0 は、並行して進む複数のノードの処理を管理することができない。

【 0 0 5 3 】

このため、従来の実験実行ソフトウェアでは、一つひとつの装置をシーケンシャルに動作させることによって、実験作業をシーケンシャルに実行していた。一般に、複数の装置で構成される自動実験システムを動作させる場合、複数の装置を並行して動作させるよりも、一つひとつの装置をシーケンシャルに動作させる方が容易である。1 次元の配列構造を持った命令集を順次実行していくことによって、実験自動システムに含まれる各装置を制御することができるからである。しかし、この方法では、実験実行ソフトウェアは、命令を与えた装置から命令に対応する動作の完了を受信してから、他の装置に対して命令を与えることになる。

【 0 0 5 4 】

実験プロトコルでは、複数の装置を並行して動作させることによって実験効率を向上させることができる。たとえば、リキッドハンドラ 1 2 3 である試料を調整しているときに、別の試料を遠心分離機 1 2 5 で分離したり、多数の試料を複数の液体クロマトグラフ質量分析装置 1 2 6 で並行して分析したりすることで、より多くの試料を短時間で作成および分析することが可能となる。

【 0 0 5 5 】

有向グラフで作成された並列処理を含む実験プロトコルを自動的に実行するために、一次元配列構造を持つ命令の配列に対して、各ノードの処理に要する時間などの各種の情報を含めることが考えられる。しかし、あるノードの処理を開始するために、処理の完了を待つ必要のあるノードの情報、各ノードの処理の所要時間に関する情報、およびノードの処理を開始する時刻などの情報をユーザが考えてシステムに設定する必要がある。このような設定作業は、ユーザの負担になる。

【 0 0 5 6 】

そこで、本実施の形態では、上記のような設計作業をユーザに強いることなく、有向グラフで作成された実験プロトコルを制御装置 1 1 0 が自動的に実行できる手法を提案する。具体的には、制御装置 1 1 0 は、有向グラフに基づいて、イベントハンドラおよび隣接行列リストを生成する。イベントハンドラおよび隣接行列リストは、有向グラフに基づいた実験を進めるために制御装置 1 1 0 が利用するデータである。

【 0 0 5 7 】

図 8 は、有向グラフ DG 3 を示す図である。ここでは、イベントハンドラおよび隣接行列リストの理解を容易にするため、ノード数の少ない有向グラフ DG 3 を用いてイベントハンドラおよび隣接行列リストを説明する。図 8 に示されるように、有向グラフ DG 3 は、スタートノードおよびエンドノードに加えて、A ~ F のノードにより構成されるものとする。A ~ F のノードは、たとえば、図 7 に示されるロボットアーム 1 2 1 の動作ノードを含む各種の処理ノードである。図 8 は、それらの処理ノードが A ~ F という概念で表されている。

【 0 0 5 8 】

10

20

30

40

50

図 8 に示されるように、有向グラフ D G 3 によれば、ノード B とノード C とが並列処理される必要がある。実験効率を高めるために、ノード B およびノード C の処理は、ノード A の処理が完了した後、速やかに実行されなければならない。しかし、ノード A の処理が完了するタイミングを正確に予測することは制御装置 1 1 0 にとって容易でない。制御装置 1 1 0 には、このような実験効率を考慮した適切な制御が求められる。

【 0 0 5 9 】

有向グラフ D G 3 によれば、ノード B およびノード C の処理の完了を待ってノード D の処理が実行される必要があり、ノード D およびノード E の処理の完了を待ってノード F の処理が実行される必要がある。実験効率を高めるために、ノード D の処理は、ノード B およびノード C の処理が完了した段階で速やかに実行される必要がある。しかし、ノード B およびノード C の処理が完了するタイミングを正確に予測することは制御装置 1 1 0 によって容易でない。同様の理由により、ノード F の処理は、ノード D およびノード E の処理が完了した段階で速やかに実行される必要がある。しかし、ノード D およびノード E の処理が完了するタイミングを正確に予測することは制御装置 1 1 0 にとって容易でない。制御装置 1 1 0 には、このような実験効率を考慮した適切な制御が求められる。

10

【 0 0 6 0 】

有向グラフ D G 3 によれば、ノード D とノード E とが並列処理される必要がある。ノード E の処理を進めるためには、ノード B の処理の完了を待つ必要がある。ノード D の処理を進めるためには、ノード B およびノード C の処理の完了を待つ必要がある。ノード D およびノード E の並列処理を効率的に進めるため、制御装置 1 1 0 には、ノード B およびノード C の並列処理を実現する場合よりも複雑な制御が求められる。

20

【 0 0 6 1 】

制御装置 1 1 0 は、各ノードの処理の順序および開始タイミングをイベントハンドラおよび隣接行列リストで制御する。次に、イベントハンドラおよび隣接行列リストの例を挙げて、それぞれについて説明する。

【 0 0 6 2 】

図 9 は、有向グラフ D G 3 に基づいて制御装置 1 1 0 が作成するイベントハンドラを示す図である。図 1 0 は、有向グラフ D G 3 に基づいて制御装置 1 1 0 が作成する隣接行列リストを示す図である。

【 0 0 6 3 】

ユーザによって設計された実験プロトコルは、サーバ装置 2 0 0 から制御装置 1 1 0 に送信される。制御装置 1 1 0 は、受信した実験プロトコルを実行するため、実験プロトコルに基づいて隣接行列およびイベントハンドラを生成する。制御装置 1 1 0 は、イベントハンドラおよび隣接行列リストを利用して実験プロトコルを実行する。制御装置 1 1 0 は、イベントハンドラおよび隣接行列リストを利用することにより、有向グラフの形式で設計された実験プロトコルを実行することができる。

30

【 0 0 6 4 】

図 9 に示されるように、イベントハンドラは、有向グラフ D 3 のノードとステータスとの組み合わせにより構成される。イベントハンドラは、ノードの処理のステータスを示すデータである。したがって、イベントハンドラは、ステータスリストとして機能する。ここで、「T o d o」は、対応するノードの処理が実行待ちであることを意味する。図 9 に示されるイベントハンドラでは、すべてのノードのステータスが「T o d o」である。したがって、図 9 のイベントハンドラは、すべてのノードの処理が実行待ちの状態であることを示す。すべてのノードの処理が実行待ちの状態であることを初期状態という。制御装置 1 1 0 は、有向グラフ D 3 に基づいて、初期状態に対応するイベントハンドラおよび隣接行列リストを生成する。

40

【 0 0 6 5 】

イベントハンドラの「T o d o」に対応するノードのうちのいずれかの処理が開始されると、対応するノードのステータスは「T o d o」から「R u n」に変化する。「R u n」に対応するノードの処理が完了すると、そのノードの処理のステータスは「R u n

50

」から「Done」に変化する。

【0066】

図10に示されるように、隣接行列リストは、有向グラフD3のすべてのノードをX軸方向とY軸方向との2方向に展開した行列形式のデータによって構成される。図10では、理解を容易にするため、X軸方向に並ぶノード名には"x"を付加し、Y軸方向に並ぶノード名には"y"を付加している。これらの付加した記号は、実際のデータに設ける必要はない。

【0067】

隣接行列リストには、スタートx列、Ax列、Bx列、Cx列...が構成されるとともに、スタートy行、Ay行、By行、Cy行...が構成される。スタートx列、Ax列、Bx列、Cx列...は、それぞれ、Column1、Column2、Column3、Column4、...とも表現される。

10

【0068】

隣接行列リストには、ノードの状態を表す成分値が設定される。たとえば、図10のBx列とAy行とによって特定される位置には、ノードの状態を表す成分値の一例として「wait」が設定されている。以下、Bx列とAy行とによって特定される位置を(Bx, Ay)のようにX軸およびY軸を用いた座標系で記載する。

【0069】

図10に示される隣接行列リストは、初期状態に対応するリストである。あるノードの処理が実行可能であるか否かは、そのノード名の列の成分値によって判定される。ここで、「ノードの処理が実行可能な状態」とは、そのノードの処理を実行するために必要となる他のノードの処理が完了していることを意味する。

20

【0070】

制御装置110は、あるノードの処理が完了すると、そのノード名に対応する行の成分値を「wait」から「ready」に更新する。たとえば、ノードAの処理が完了すると、Ay行に存在するすべての成分値が「wait」から「ready」に更新される。その結果、(Bx, By)および(Cx, Ay)の成分値が「wait」から「ready」に更新される。

【0071】

このように、成分値が「wait」から「ready」に更新される契機は、ノードの処理の完了である。また、「wait」から「ready」に更新される対象は、処理が完了したノード名に対応する行の成分値である。

30

【0072】

制御装置110は、あるノードの処理を実行可能であるか否かを判定するために、そのノードに対応する列をチェックする。そのノードに対応する列に「wait」の成分値が1つでも存在する場合、制御装置110は、その列に対応するノードの処理は実行可能でないと判定する。そのノードに対応する列に「wait」の成分値が1つも存在しない場合、制御装置110は、その列に対応するノードの処理は実行可能であると判定する。たとえば、図10に示される隣接行列リストの状態からすると、実行できる状態にあるノードの処理は、スタートノードの処理のみである。スタートx列には「wait」の成分値が1つも存在しない一方、他の列には「wait」の成分値が少なくとも1つ存在するからである。

40

【0073】

スタートノードに実質的な処理は存在しないので、開始したスタートノードの処理はただちに完了する。制御装置110は、自らが出したスタートノードの処理を開始する指令を自身で受けて、スタートノードの処理の完了信号を生成する。これによって、即座にスタートノードの処理を完了させる。このとき、制御装置110は、スタートノードに対応するスタートy行の成分値を「wait」から「ready」に更新する。その結果、(Ax, スタートy)の成分値が「ready」に変化する。すると、Ax列に「wait」の成分値が存在しなくなる。図9のイベントハンドラにおいて、ノードAは処理の実行

50

待ちの状態である。イベントハンドラおよび隣接行列リストの状態から、ノード A の処理を実行可能な状態であることが特定される。

【 0 0 7 4 】

制御装置 1 1 0 は、ノード A の処理を開始させるとともに、隣接行列リストにおいて、A x 列の成分値を「 r e a d y 」から「 p a s s e d 」に更新する。その結果、(A x , スタート y) の成分値が「 p a s s e d 」に変化する。

【 0 0 7 5 】

このように、成分値が「 r e a d y 」から「 p a s s e d 」に更新される契機は、ノードの処理の完了である。また、「 r e a d y 」から「 p a s s e d 」に更新される対象は、処理が開始したノード名に対応する列の成分値である。

10

【 0 0 7 6 】

次に、実験プロトコルに基づいた実験が進行しているときのイベントハンドラおよび隣接行列リストの一例を説明する。図 1 1 は、有向グラフ D G 3 に基づいた処理の進行に応じてイベントハンドラが更新される様子を示す図である。図 1 2 は、有向グラフ D G 3 に基づいた処理の進行に応じて隣接行列リストが更新される様子を示す図である。

【 0 0 7 7 】

図 1 1 に示されるように、ここでは、有向グラフ D 3 のうち、スタートノード、ノード A、およびノード C の処理が完了し、ノード B の処理が実行中である状態を例に挙げて、その状態に対応するイベントハンドラおよび隣接行列リストを説明する。

【 0 0 7 8 】

20

イベントハンドラには、有向グラフ D 3 におけるノードの処理の状態が反映される。図 1 1 に示されるように、スタートノード、ノード A、およびノード C に対応するステータスは「 D o n e 」となり、ノード B に対応するステータスは「 R u n 」となる。処理待ちとされているノード D、ノード E、ノード F、およびエンドノードに対応するステータスは「 T o d o 」である。

【 0 0 7 9 】

図 1 2 に示される隣接行列リストにおいて、A x 列、B x 列、および C x 列の成分値は「 p a s s e d 」である。これらの成分値は、ノード A、ノード B、およびノード C の処理が処理待ちでないことを意味する。ここで、図 1 1 に示されるイベントハンドラを参照することにより、ノード A およびノード C の処理が実行済みであり、ノード B の処理が実行中であることが特定される。制御装置 1 1 0 は、イベントハンドラおよび隣接行列リストを参照することにより、これらの状態を特定する。

30

【 0 0 8 0 】

隣接行列リストにおいて、(D x , C y) の値は「 r e a d y 」であるが、(D x , B y) の値は「 w a i t 」である。(D x , B y) の値は、B y 行に対応するノード B の処理が完了していないことを意味する。(D x , C y) の値は、C y 行に対応するノード C の処理が完了していることを意味する。

【 0 0 8 1 】

制御装置 1 1 0 は、ノード B の処理が完了すると、イベントハンドラのノード B に対応するステータスを「 R u n 」から「 D o n e 」に変化させる。また、制御装置 1 1 0 は、ノード B の処理が完了すると、隣接行列リストの B y 行の 2 箇所の「 w a i t 」を「 r e a d y 」に変化させる。これにより、D x 列および E x 列には「 w a i t 」が存在しなくなる。このとき、制御装置 1 1 0 は、ノード D およびノード E が実行可能な状態と判定できる。制御装置 1 1 0 は、ノード D およびノード E の処理を開始させるとともに、イベントハンドラのノード D およびノード E のステータスを「 T o d o 」から「 R u n 」に変化させる。さらに、制御装置 1 1 0 は、隣接行列リストの D x 列および E x 列の「 r e a d y 」を「 p a s s e d 」に変化させる。

40

【 0 0 8 2 】

以上、説明したように、制御装置 1 1 0 は、実験プロトコル内のノード間のつながりを行列の成分で表した隣接行列リストを生成し、隣接行列リストの成分の値を変化させるこ

50

とによって、ノードの処理を実行するタイミングを管理する。制御装置 110 は、隣接行列リストにおいて依存関係にあるノードの処理がすべて完了すると、次のノードの処理を指令する。1つのノードは、基本的に1つの装置の動作条件の情報を持つ命令で構成される。ノードに対応する処理を実行するときには、そのノードに対応する命令が制御装置 110 から対象の装置に送信される。制御装置 110 は、命令を送信した装置から動作の完了を示す完了信号を受信すると、そのノードに対応する処理が完了したものの判定する。

【0083】

制御装置 110 は、自動実験を実行しているときに、最新のイベントハンドラおよび隣接行列リストを端末装置 400 へ出力してもよい。端末装置 400 は、入力されたイベントハンドラおよび隣接行列リストを表示装置に表示してもよい。

10

【0084】

このように、制御装置 110 は、有向グラフに基づいてイベントハンドラおよび隣接行列リストを生成する。制御装置 110 は、イベントハンドラおよび隣接行列リストに基づいて、有向グラフによって設計された実験プロトコルを実行する。

【0085】

図 13 は、実験プロトコルに基づく自動実験の流れを説明するフローチャートである。図 13 に示されるように、端末装置 400 は、ユーザの操作に基づいて実験プロトコルを有向グラフで設計し、実験プロトコルをサーバ装置 200 に送信する（ステップ S11）。サーバ装置 200 は、ユーザの操作に従って、自動実験システム 1 に実験プロトコルを送信する（ステップ S12）。自動実験システム 1 の制御装置 110 は、サーバ装置 200 から受信した実験プロトコルを自動的に実行する（ステップ S13）。制御装置 110 は、実験プロトコルに含まれる処理の出力データをサーバ装置 200 に送信する（ステップ S14）。

20

【0086】

本実施の形態においては、端末装置 400 において設計された実験プロトコルがサーバ装置 200 を介して自動実験システム 1 に送信される場合について説明した。端末装置 400 は、実験プロトコルを自動実験システム 1 に直接送信してもよい。

【0087】

次に、制御装置 110 が実験プロトコルを自動的に実行する処理（図 13 のステップ S13）に関して、フローチャートを用いてより具体的に説明する。図 14 は、実験プロトコルに応じて制御装置 110 がイベントハンドラおよび隣接行列リストを生成する処理を示すフローチャートである。

30

【0088】

はじめに、制御装置 110 は、有向グラフで作成された実験プロトコルを受信する（ステップ S101）。有向グラフで作成された実験プロトコルは、サーバ装置 200 から制御装置 110 に送信される。

【0089】

次に、制御装置 110 は、受信した有向グラフの形式を変換する（ステップ S102）。具体的には、制御装置 110 は、受信した有向グラフがロボットアーム 121 の動作ノードを含むように、有向グラフ形式を変換する。これにより、たとえば、図 6 に示される有向グラフが図 7 に示される有向グラフに変換される。

40

【0090】

次に、制御装置 110 は、変換後の有向グラフに基づいて、イベントハンドラおよび隣接行列リストを生成し（ステップ S103、ステップ S104）、本フローチャートに基づく処理を終了する。たとえば、制御装置 110 は、図 8 に示される有向グラフに基づいて、図 9 に示されるイベントハンドラおよび図 10 に示される隣接行列リストを生成する。

【0091】

図 15 は、制御装置 110 がイベントハンドラおよび隣接行列リストに基づいて実験プロトコルを実行する処理を示すフローチャートである。

【0092】

50

はじめに、制御装置 110 は、変数 `Column` を 1 に設定する (ステップ S111)。ここで、`Column` は、図 10 に示される隣接行列リストの列の番号に対応する。たとえば、ステップ S111 において 1 に設定された `Column` は、図 10 に示されるスタート x 列に対応する。

【0093】

次に、制御装置 110 は、隣接行列リストの `Column` に対応する列のデータを検索する (ステップ S112)。次に、制御装置 110 は、`Column` に対応する列に「wait」が存在しないか否かを判定する (ステップ S113)。制御装置 110 は、`Column` に対応する列に「wait」が存在しない場合、ステップ S118 の処理に進み、`Column` の値を更新する。これにより、たとえば、`Column` の値は 1 から 2 に更新される。

10

【0094】

制御装置 110 は、ステップ S113 において、`Column` に対応する列に「wait」が存在しないと判定した場合、イベントハンドラにおいて、`Column` に対応するノードのステータスは「To do」か否かを判定する (ステップ S114)。これにより、制御装置 110 は、`Column` に対応するノードが処理対象であり、かつ、実行待ちであることを確認する。

【0095】

制御装置 110 は、ステップ S114 において、`Column` に対応するノードのステータスが「To do」でないと判定した場合、ステップ S118 の処理に進む。制御装置 110 は、ステップ S114 において、`Column` に対応するノードのステータスが「To do」であると判定した場合、`Column` に対応するイベントハンドラのノードのステータスを「To do」から「Run」に変更する (ステップ S115)。これにより、新たに処理を開始させるノードが制御装置 110 によって選択されたことになる。次に、制御装置 110 は、`Column` に対応する列に存在するすべての「ready」を「passed」に変更する (ステップ S116)。

20

【0096】

次に、制御装置 110 は、ノードの処理に対応する装置へ動作を指示する (ステップ S117)。ステップ S117 の処理により、ステップ S115 にてステータスが「Run」に変更されたノードに対応する処理が実行される。

30

【0097】

ノードの処理に対応する装置には、たとえば、ロボットアーム 121 と、インキュベータ 122 と、リキッドハンドラ 123 と、マイクロプレートリーダー 124 と、遠心分離機 125 と、液体クロマトグラフ質量分析装置 126 とが含まれる。これらの装置は、制御装置 110 からステップ S117 においてノードの処理に応じた動作の指示を受ける。これらの装置は、指示に応じた動作を実行することにより、実験プロトコルに含まれる処理を実現する。これらの装置は、指示に応じた動作を完了したときに、動作の完了を示す完了信号を制御装置 110 へ送信する。

【0098】

ノードの処理に対応する装置には、制御装置 110 がさらに含まれる。制御装置 110 は、スタートノードおよびエンドノードの処理に加えて、「特徴量抽出」などに対応するノードの処理を実行する。制御装置 110 は、自身の指示に応じた動作を自ら実行することにより、実験プロトコルに含まれる処理を実現する。制御装置 110 は、指示に応じた動作を完了したときに、動作の完了を示す完了信号を自らに返す。

40

【0099】

次に、制御装置 110 は、`Column` の値を更新する (ステップ S118)。次に、制御装置 110 は、いずれかの装置から完了信号を受信したか否かを判定する (ステップ S119)。制御装置 110 は、完了信号を受信しない場合、ステップ S122 に処理を進める。制御装置 110 は、完了信号を受信した場合、イベントハンドラにおいて、完了した処理に対応するノードのステータスを「Run」から「Done」に変更する (ステ

50

ップ S 1 2 0)。

【 0 1 0 0 】

さらに、制御装置 1 1 0 は、隣接行列リストにおいて、完了した処理のノードに対応する行に存在するすべての「wait」を「ready」に変更する（ステップ S 1 2 1）。たとえば、ノード B の処理が終了した場合、図 1 0 に示される隣接行列リストにおいて、B y 行に存在する 2 つの「wait」が「ready」に変更される。

【 0 1 0 1 】

次に、制御装置 1 1 0 は、イベントハンドラおよび隣接行列リストを表示するための表示信号を出力する（ステップ S 1 2 2）。ユーザは、予め表示信号の出力先を制御装置 1 1 0 に通知してもよい。この場合、制御装置 1 1 0 は、通知されている出力先へイベントハンドラおよび隣接行列リストの表示信号を出力する。たとえば、制御装置 1 1 0 に表示装置が接続されている場合、制御装置 1 1 0 は、表示装置へ表示信号を出力してもよい。これにより、表示装置にイベントハンドラおよび隣接行列リストが表示される。

【 0 1 0 2 】

端末装置 4 0 0 に表示装置が接続されている場合、制御装置 1 1 0 は、端末装置 4 0 0 へ表示信号を出力してもよい。この場合、端末装置 4 0 0 は、表示装置の画面にイベントハンドラおよび隣接行列リストを表示する。

【 0 1 0 3 】

ユーザは、画面で最新の状態のイベントハンドラおよび隣接行列リストを見ることができ。このため、ユーザは、自動実験の進行状況を簡単に把握できる。

【 0 1 0 4 】

次に、制御装置 1 1 0 は、イベントハンドラのステータスがすべて「Done」か否かを判定する（ステップ S 1 2 3）。イベントハンドラのステータスがすべて「Done」でない場合、実行していないノードの処理が存在することになる。そこで、この場合、制御装置 1 1 0 は、ステップ S 1 1 2 に戻り、処理を繰り返す。制御装置 1 1 0 は、イベントハンドラのステータスがすべて「Done」であると判定した場合、予定された実験プロトコルの処理はすべて終了と判断し、本フローチャートに基づく処理を終える。

【 0 1 0 5 】

制御装置 1 1 0 は、イベントハンドラおよび隣接行列リストを用いて、以上に説明した手順で有向グラフに基づいた実験プロトコルを実行する。このため、制御装置 1 1 0 によれば、有向グラフ D G 3（図 8 参照）に含まれるような複数のノードの並列処理を適切に制御することができる。また、制御装置 1 1 0 によれば、有向グラフ D G 3 に含まれるノード D、F のように、複数のノードの処理の完了を待って開始させる必要のあるノードの処理も適切に制御することができる。その結果、制御装置 1 1 0 によれば、有向グラフで作成された実験プロトコルに基づいた実験効率を高めることができる。

【 0 1 0 6 】

図 1 6 は、制御装置 1 1 0 の機能に関わる構成を説明するための自動実験管理システム 1 0 0 0 の構成を示すブロック図である。フローチャートに基づいて説明した制御装置 1 1 0 の処理に関して、ここでは、制御装置 1 1 0 の機能として説明する。制御装置 1 1 0 は、図 1 6 に示されるように、入力部 1 1 3、生成部 1 1 4、選択部 1 1 5、指令部 1 1 6、更新部 1 1 7、出力部 1 1 8、および変換部 1 1 9 を備える。これらの構成は、図 1 に示した制御装置 1 1 0 のプロセッサ 1 1 1 およびメモリ 1 1 2 によって実現される。

【 0 1 0 7 】

入力部 1 1 3 は、実験の処理に対応する複数のノードを含む有向グラフで設計された実験プロトコルの入力を受け付ける。生成部 1 1 4 は、隣接行列リストなどを生成する。選択部 1 1 5 は、複数のノードの中から処理を開始させるノードを隣接行列リストなどに基づいて選択する。指令部 1 1 6 は、選択部 1 1 5 により選択されたノードの処理を種々の実験装置に指令する。更新部 1 1 7 は、複数のノードの処理のうちいずれかが完了した場合に、隣接行列リストなどを更新する。出力部 1 1 8 は、隣接行列リストなどを表示するための信号を出力する。変換部 1 1 9 は、入力部 1 1 3 に入力された実験プロトコルが

10

20

30

40

50

、サンプルを収容する容器に対応するノードを含む第1有向グラフで設計されている場合、入力部113に入力された実験プロトコルを、容器を移動する装置の動作を含む第2有向グラフに変換する。

【0108】

図17は、サーバ装置200のハードウェア構成を示すブロック図である。図17に示されるように、サーバ装置200は、プロセッサ201と、記憶部としてのメモリ202およびハードディスク203と、通信部としての通信インターフェース204と、入出力部205とを含む。これらは、バス210を介して相互に通信可能に接続されている。

【0109】

ハードディスク203は、不揮発性の記憶装置である。ハードディスク203には、たとえばオペレーティングシステム(OS: Operating System)のプログラム51、および自動実験管理プログラム52が保存されている。ハードディスク203には、たとえば各種アプリケーションの設定および出力が保存される。メモリ202は、揮発性の記憶装置であり、たとえばDRAM(Dynamic Random Access Memory)を含む。

【0110】

プロセッサ201は、CPU(Central Processing Unit)を含む。プロセッサ201は、ハードディスク203に保存されているプログラムをメモリ202に読み込んで実行し、サーバ装置200の各種機能を実現する。たとえば、自動実験管理プログラム52を実行するプロセッサ201は、端末装置400に実験プロトコル設計アプリケーション500を提供する。プロセッサ201は、通信インターフェース204を介してネットワークNWに接続する。

【0111】

図18は実施の形態の変形例1に係る自動実験管理システム1100の構成を示すブロック図である。自動実験管理システム1100の構成は、図1の自動実験管理システム1000からサーバ装置200およびデータベース300が除かれ、端末装置400が400Aに置き換えられた構成である。これら以外は同様であるため、説明を繰り返さない。端末装置400Aのディスプレイ431には、実験プロトコル設計アプリケーション500Aが表示されている。

【0112】

図19は、変形例1に関わる端末装置400Aのハードウェア構成を示すブロック図である。端末装置400Aの構成は、図2のハードディスク423に自動実験管理プログラム52Aが追加された構成である。これ以外は同様であるため説明を繰り返さない。自動実験管理プログラム52Aがプロセッサ421によって実行されることにより、実験プロトコル設計アプリケーション500Aおよび自動実験システムによる実験プロトコルの自動実行が実現される。

【0113】

実験プロトコルの設計は、自動実験システムの制御装置において行われてもよい。図20は、変形例2に係る自動実験システム1Bの構成を示すブロック図である。自動実験システム1Bの構成は、図1の自動実験システム1において、制御装置110が110Bに置き換えられた構成である。これ以外は同様であるため、説明を繰り返さない。

【0114】

図20に示されるように、制御装置110Bは、入出力部130と、計算機140(処理部)とを含む。入出力部130は、ディスプレイ131(表示部)と、キーボード132(入力部)と、マウス133(入力部)とを含む。ディスプレイ131、キーボード132、およびマウス133は、計算機140に接続されている。ディスプレイ131には、実験プロトコル設計アプリケーション500BのGUIが表示されている。キーボード132およびマウス133は、ユーザによる実験プロトコル設計アプリケーション500BへのGUI操作を受け付ける。すなわち、ユーザは、ディスプレイ131の表示を参照しながら、キーボード132の操作、またはマウス133の操作によって、実験プロトコル設計アプリケーション500Bへ所望のGUI操作を行う。

10

20

30

40

50

【0115】

図21は、制御装置110Bのハードウェア構成を示すブロック図である。図21に示されるように、計算機140は、プロセッサ141と、記憶部としてのメモリ142およびハードディスク143と、通信インターフェース144とを含む。これらは、バス145を介して相互に通信可能に接続されている。

【0116】

ハードディスク143は、不揮発性の記憶装置である。ハードディスク143には、たとえばオペレーティングシステム(OS: Operating System)のプログラム61、および自動実験管理プログラム52Bが保存されている。図21に示されるデータ以外にも、ハードディスク143には、たとえば各種アプリケーションの設定および出力が保存される。メモリ142は、揮発性の記憶装置であり、たとえばDRAM(Dynamic Random Access Memory)を含む。

10

【0117】

プロセッサ141は、CPU(Central Processing Unit)を含む。プロセッサ141は、ハードディスク143に保存されているプログラムをメモリ142に読み込んで実行する。自動実験管理プログラム52Bがプロセッサ141によって実行されることにより、実験プロトコル設計アプリケーション500Bおよび複数の実験装置120による実験プロトコルの自動実行が実現される。プロセッサ141は、通信インターフェース144を介してネットワークに接続する。

【0118】

図22は、有向グラフDG4を示す図である。本実施の形態では、端末装置400を用いて様々な形式で有向グラフを作成することができる。有向グラフDG4では、ノードが装置の名称を用いて特定されている。

20

【0119】

ノードLH1は、リキッドハンドラ123による第1の処理を示す。リキッドハンドラ123は、たとえば、サンプルを分配する。ノードLH2は、リキッドハンドラ123による第2の処理を示す。INCは、インキュベータ122による処理を示す。インキュベータ122は、たとえば、細胞を培養する。PRは、マイクロプレートリーダー124による処理を示す。マイクロプレートリーダー124は、たとえば、マイクロプレート内のサンプルの光学的性質を測定する。CENは、遠心分離機125による処理を示す。LCMSは、液体クロマトグラフ質量分析装置126による処理を示す。

30

【0120】

有向グラフDG4では、ロボットアーム121のノードが省略されている。

図23は、有向グラフDG4および有向グラフDG4に対してロボットアーム121のノードを追加した有向グラフDG5を示す図である。有向グラフDG4を、ロボットアーム121のノードを追加した有向グラフDG5に変換できるように制御装置110を構成してもよい。

【0121】

以上説明した本実施の形態および変形例1に係る方法およびシステム、ならびに実施の形態の変形例2に係る装置によれば、より高度な処理に対応可能な実験プロトコルの設計方法を提供することができる。

40

【0122】

本実施の形態によれば、ユーザが作成した有向グラフの形式の実験プロトコルに基づいて、制御装置110により実験が自動的に進められる。このため、本実施の形態によれば、軽量な実装でグラフ構造の実験プロトコルを自動実験システム1においてそのまま実行できるようになる。

【0123】

本実施の形態によれば、可視化されたグラフ構造の実験プロトコルのとおりに、実験に関わる装置が制御される。このため、ユーザは並行して動作する装置の動作の所要時間や動作完了のタイミング、次の動作開始のタイミングなどを考慮せずに、並列処理を含む複

50

雑な実験プロトコルを容易に作成することができる。また、並列処理を含む実験プロトコルに基づいて実験が実行されるため、実験の効率を高めることができる。

【0124】

本実施の形態によれば、装置の動作を1つのノードとしたグラフ構造のプロトコルを用いることによって、ユーザは、自身の把握している装置の動作に対応したわかりやすいプロトコルを作成することが可能となる。たとえば、ユーザは、LCMSのオートローダー、バッチファイルロード、およびバッチ開始の動作命令を1つのLCMSのノードで表現することも可能となる。

【0125】

本実施の形態によれば、ユーザは、試料を搬送するロボットアーム121の動作を省略して実験プロトコルを作成することができる。この場合、ユーザは、ロボットアーム121の動作のノードに代えて、試料が入ったプレートに対応するノード、プレートの移動先に対応するノード、およびプレートの移動元に対応するノードを用いることができる。ユーザは、それらのノードを実験プロトコルの処理の流れに対応するようにエッジで接続することにより、より直感的に実験プロトコルを作成することができる(図6参照)。

10

【0126】

本実施の形態によれば、ノードの依存関係を表す隣接行列リストを使用することにより、並列処理を含む有向グラフに基づいて実験を進めることができる。

【0127】

以下、本実施の形態に関わるさらなる変形例や特徴点に言及する。

20

イベントハンドラに設定できるステータスの種類を少なくしてもよい。たとえば、「ToDo」、「Run」、および「Done」のうち、「Run」が用いられることなく、イベントハンドラが更新されてもよい。「Run」が用いられない場合であっても、制御装置110は、「ToDo」および「Done」によって、対象のノードの処理を実行させる必要があるか、対象のノードの処理が実行済みであるかを判定できる。

【0128】

隣接行列リストに設定できる成分値の種類を少なくしてもよい。たとえば、「wait」、「ready」、および「passed」のうち、「ready」が用いられることなく、隣接行列リストが更新されてもよい。「ready」が用いられない場合であっても、制御装置110は、「wait」および「passed」によって、対象のノードの処理を実行させることが可能であるか否かを判定できる。

30

【0129】

イベントハンドラに設定できるステータスの種類を変更または追加してもよい。たとえば、実行中の処理にエラーが発生した場合、その処理に対応するノードのステータスとして、「Error」がイベントハンドラに設定されるように構成してもよい。イベントハンドラに設定された場合、制御装置110は、対応するノードの処理に関わる装置に対して、再起動の命令を送信してもよい。

【0130】

隣接行列リストにイベントハンドラの機能を設けてもよい。たとえば、隣接行列リストに、ノードの実行待ち、実行中、実行完了を示す成分値を設けてもよい。

40

【0131】

制御装置110は、インターネットなどの通信網を介して、イベントハンドラおよび隣接行列リストのうち、少なくとも一方を外部に出力してもよい。ユーザは、制御装置110から出力されたデータを表示装置に表示することにより、遠隔地で自動実験の進行状況を把握できる。

【0132】

有向グラフに含まれるノードには、動作対象の装置、動作対象の装置の処理、および動作条件などの情報を保持させておく。第1条件が成立したときに装置を動作させる第1命令と、第2条件が成立したときに装置を動作させる第2命令とを実験プロトコルに含ませる場合、第1命令と第2命令とに基づいて動作する装置が同じであっても、第1命令と第

50

2 命令とは別のノードとしてユーザにより作成されることが望ましい。

【0133】

ある実験プロトコルでは、試料がロボットアーム121によって様々な装置間を移動することにより、試料の作製、調整、および分析が行われる。本実施の形態によれば、このような実験プロトコルを有向グラフで作成する場合、試料をノードとして表現し、移動する装置間の関係をエッジで表現することができる。この場合、ユーザは、ロボットアーム121の動作のノードを省略して有向グラフを作成することができる。しかし、ロボットアーム121の動作をノードで表現した有向グラフを作成することに興味のあるユーザがいるかもしれない。したがって、端末装置400は、そのような有向グラフをユーザが作成できるシステム環境を提供してもよい。

10

【0134】

有向グラフにおいて、第1ノードと第2ノードとをエッジで接続し、第1ノードと第3ノードとをエッジで接続することにより、第2ノードと第3ノードとを並列処理する命令を表現することができる。この場合、第2ノードに対応して動作する装置と第3ノードに対応して動作する装置とは、異なる装置が設定される必要がある。したがって、ユーザが第2ノードに対して指定した装置と第3ノードに対して指定した装置とが同一の装置である場合、ユーザの指定を受け付けずに端末装置400を構成することが望ましい。

【0135】

ただし、同種の装置が複数台存在する場合、端末装置400は、そのようなユーザの指定を受け付けてもよい。また、ユーザが第2ノードに対して指定した試料と第3ノードに対して指定した試料とが同一である場合、ユーザの指定を受け付けずに端末装置400を構成することが望ましい。もちろん、第2ノードの処理と第3ノードの処理とに処理の順序が明確に定まっている場合、端末装置400はその指定を受け付けてもよい。

20

【0136】

複数の実験装置が実験プロトコルのメソッドを受け付ける場合、実験装置毎に独自に Ready / Busy、Available / Lockedなどのステータスや、メソッドキューを管理してもよい。

【0137】

ユーザに理解し易い実験プロトコルのグラフ構造を提供するために、1つの装置の複数の動作条件の情報、および複数の装置の動作条件の情報を含む複数の命令のセットを1つのノードとして構成してもよい。

30

【0138】

試料がロボットアーム121によって様々な装置間を移動することにより、試料の作製、調整、および分析が行われる実験プロトコルも存在する。このような実験プロトコルの場合、ロボットアーム121の単独の動作のノードを省略し、装置の動作とロボットアーム121による試料の搬入および搬出の動作との命令をセットにした1つのノードで表現すればよい。この場合、1つのノード内の複数の命令はシーケンシャルに実行される。制御装置110は、最後の命令に対応した完了信号を装置から受信すると、ノードの処理が完了したものと判定する。

【0139】

並列処理されるノード数に対応する数のロボットアーム121が存在する場合、各々のロボットアーム121が並行して処理を進めるような実験プロトコルを設計してもよい。これにより、たとえば、各々のロボットアーム121が同時に試料を運搬するような実験プロトコルを作成することができる。

40

【0140】

並列するノードの数よりもロボットアーム121の数が少ない場合、ロボットアーム121の動作をノードで表現してもよい。並列するノードの数よりもロボットアーム121の数が少ない場合、ロボットアーム121の動作に対応するノードに代えて、装置のノードとしてロボットアーム121のノードを採用してもよい。

【0141】

50

図 2 3 に示される有向グラフ D G 4 では、ロボットアーム 1 2 1 のノードが省略されている。システムにロボットアーム 1 2 1 が 1 台しか存在しない場合、ロボットアーム 1 2 1 以外の各装置のノードの直前に、ロボットアーム 1 2 1 の動作ノードを挿入してもよい。この場合、ロボットアーム 1 2 1 の動作ノード内に、各装置へ試料を搬入する動作または各装置から試料を搬出する動作をシーケンシャルに含めてもよい。

【 0 1 4 2 】

複数のノードの処理を並行して実行させる場合、並行に動作させる複数のノードの直前または直後にロボットアーム 1 2 1 の動作のノードを挿入する。たとえば、図 2 3 に示される有向グラフ D G 5 では、並行に動作させる L H 2 ノードと I N C ノードとの直前に R B ノードが挿入されている。また、ロボットアーム 1 2 1 の数に対応する数しか並列した

10

【 0 1 4 3 】

[態 様]

上述した例示的な実施の形態は、以下の態様の具体例であることが当業者により理解される。

【 0 1 4 4 】

(第 1 項) 一態様に係る制御装置は、実験プロトコルに基づいて実験装置を制御する制御装置である。制御装置は、実験の処理に対応する複数のノードを含む有向グラフで設計された実験プロトコルの入力を受け付ける入力部と、第 1 ノードおよび第 1 ノードの処理の完了によって実行可能となる第 2 ノードが複数のノードに含まれる場合、第 2 ノードの

20

【 0 1 4 5 】

第 1 項に記載の制御装置によれば、有向グラフで作成された実験プロトコルに基づいた実験効率を高めることができる。

【 0 1 4 6 】

(第 2 項) 第 1 項に記載の制御装置において、更新部は、第 1 ノードの処理が完了した場合、第 1 依存情報を第 1 情報から第 2 情報に更新し、選択部は、第 1 依存情報が第 1 情報から第 2 情報に更新された場合、第 2 ノードを選択する。

30

【 0 1 4 7 】

第 2 項に記載の制御装置によれば、有向グラフで作成された実験プロトコルに基づいた実験効率を高めることができる。

【 0 1 4 8 】

(第 3 項) 第 2 項に記載の制御装置において、生成部は、複数のノードの各々のステータスを特定するためのステータスリストをさらに生成し、更新部は、第 1 ノードの処理が完了した場合、ステータスリストにおける第 1 ノードのステータスを実行中から完了に更新し、第 2 ノードの処理を開始する場合、ステータスリストにおける第 2 ノードのステータスを実行待ちから実行中に更新する。

40

【 0 1 4 9 】

第 3 項に記載の制御装置によれば、ステータスリストに基づいて複数のノードの各々のステータスを特定しつつ、処理を進めることができる。

【 0 1 5 0 】

(第 4 項) 第 3 項に記載の制御装置において、複数のノードは、第 1 ノードの処理の完了によって実行可能となる第 3 ノードを含み、依存関係リストは、第 3 ノードの処理の開始が第 1 ノードの処理の完了に依存することを示す第 2 依存情報を含み、更新部は、第 1 ノードの処理が完了した場合、第 1 依存情報および第 2 依存情報を第 1 情報から第 2 情報

50

に更新し、選択部は、第1依存情報および第2依存情報が第1情報から第2情報に更新された場合、第2ノードおよび第3ノードを選択する。

【0151】

第4項に記載の制御装置によれば、第2ノードの処理および第3ノードの処理を並列して実行することができる。

【0152】

(第5項)第4項に記載の制御装置において、複数のノードは、第2ノードおよび第3ノードの処理が完了した後に、処理を開始することが可能となる第4ノードを含み、依存関係リストは、第4ノードの処理の開始が第2ノードの処理の完了に依存することを示す第3依存情報と、第4ノードの処理の開始が第3ノードの処理の完了に依存することを示す第4依存情報とを含み、更新部は、第2ノードの処理が完了した場合、第3依存情報を第1情報から第2情報に更新し、第3ノードの処理が完了した場合、第4依存情報を第1情報から第2情報に更新し、選択部は、第3依存情報および第4依存情報が第1情報から第2情報に更新された場合、第4ノードを選択する。

10

【0153】

第5項に記載の制御装置によれば、並列して実行される第2ノードおよび第3ノードの処理が完了したタイミングで、第4ノードの処理を実行することができる。

【0154】

(第6項)第1項～第5項のいずれか1項に記載の制御装置において、依存関係リストを表示するための信号を出力する出力部をさらに備える。

20

【0155】

第6項に記載の制御装置によれば、ユーザは、表示された依存関係リストを確認することによって、実験プロトコルの進行状況を把握できる。

【0156】

(第7項)第1項～第6項のいずれか1項に記載の制御装置において、入力部に入力された実験プロトコルが、サンプルを収容する容器に対応するノードを含む第1有向グラフで設計されている場合、入力部に入力された実験プロトコルを、容器を移動する装置の動作を含む第2有向グラフに変換する変換部をさらに備え、生成部は、第2有向グラフに基づいて依存関係リストを生成する。

【0157】

第7項に記載の制御装置によれば、サンプルを収容する容器に対応するノードを用いて直感的にユーザが作成した第1有向グラフに基づいて、実験が進められる。このため、ユーザは、より一層、容易に実験プロトコルを作成することができる。

30

【0158】

(第8項)一態様に係る方法は、実験プロトコルに基づいて実験装置を制御する方法である。方法は、実験の処理に対応する複数のノードを含む有向グラフで設計された実験プロトコルの入力を受け付けるステップと、第1ノードおよび第1ノードの処理の完了によって実行可能となる第2ノードが複数のノードに含まれる場合、第2ノードの処理の開始が第1ノードの処理の完了に依存することを示す第1依存情報を含む依存関係リストを生成するステップと、複数のノードの中から処理を開始させるノードを依存関係リストに基づいて選択するステップと、選択部により選択されたノードの処理を実験装置に指令するステップと、複数のノードの処理のうちのいずれかが完了した場合に、依存関係リストまたは複数のノードの状態を更新するステップとを含み、処理を開始させるノードを選択するステップは、第1ノードの処理が完了した場合、第2ノードを選択する。

40

【0159】

第8項に記載の方法によれば、有向グラフで作成された実験プロトコルに基づいた実験効率を高めることができる。

【0160】

なお、上述した実施の形態および変更例について、明細書内で言及されていない組み合わせを含めて、不都合または矛盾が生じない範囲内で、実施の形態で説明された構成を適

50

宜組み合わせることは出願当初から予定されている。

【0161】

今回開示された実施の形態はすべての点で例示であって制限的なものではないと考えられるべきである。本発明の範囲は上記した説明ではなくて請求の範囲によって示され、請求の範囲と均等の意味および範囲内でのすべての変更が含まれることが意図される。

【符号の説明】

【0162】

1, 1B 自動実験システム、110, 110B 制御装置、111 プロセッサ、112 メモリ、113 入力部、114 生成部、115 選択部、116 司令部、117 更新部、118 出力部、119 変換部、120 実験装置、121 ロボットアーム、122 インキュベータ、123 リキッドハンドラ、124 マイクロプレートリーダー、125 遠心分離機、130, 205, 430 入出力部、131, 431 ディスプレイ、132, 432 キーボード、133 マウス、140 計算機、141, 201, 421 プロセッサ、142, 202, 422 メモリ、143, 203, 423 ハードディスク、144, 204, 424 通信インターフェース、145, 210, 440 バス、200 サーバ装置、300 データベース、400, 400A 端末装置、433 タッチパッド、500, 500A, 500B 実験プロトコル設計アプリケーション、510 キューリストウィンドウ、520 プロトコルリストウィンドウ、530 プロトコル設計ウィンドウ、531 容器領域、532 処理領域、533 データ領域、540 自動実験システムウィンドウ、550 サンプル容器ウィンドウ、560 ツールウィンドウ、570 情報ウィンドウ、600 Webブラウザ、1000, 1100 自動実験管理システム、Cr 選択カーソル、NW ネットワーク、Plt1, Plt2 プレート、p1 実験プロトコル。

10

20

30

40

50

【図面】
【図 1】

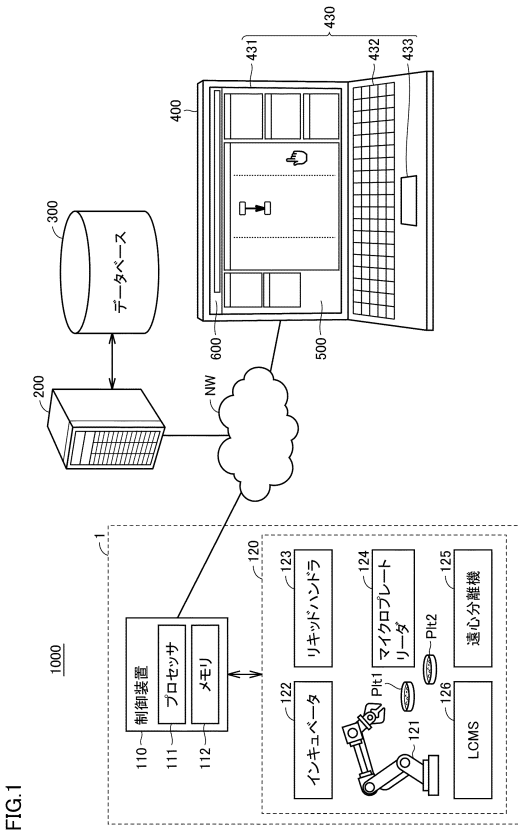
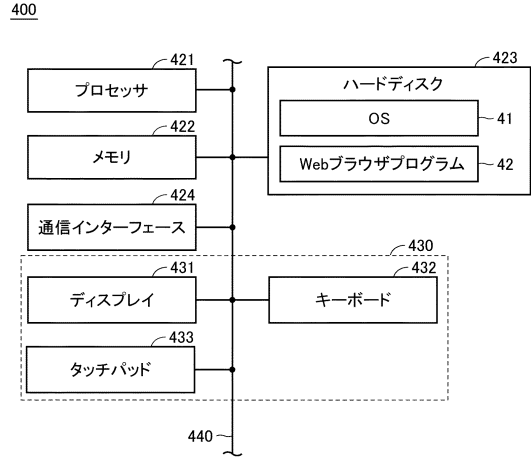


FIG.1

【図 2】

FIG.2



10

20

【図 3】

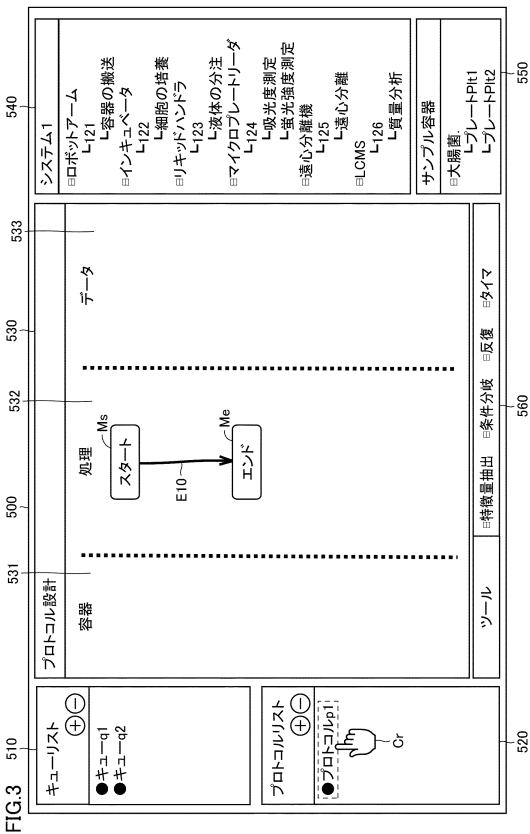


FIG.3

【図 4】

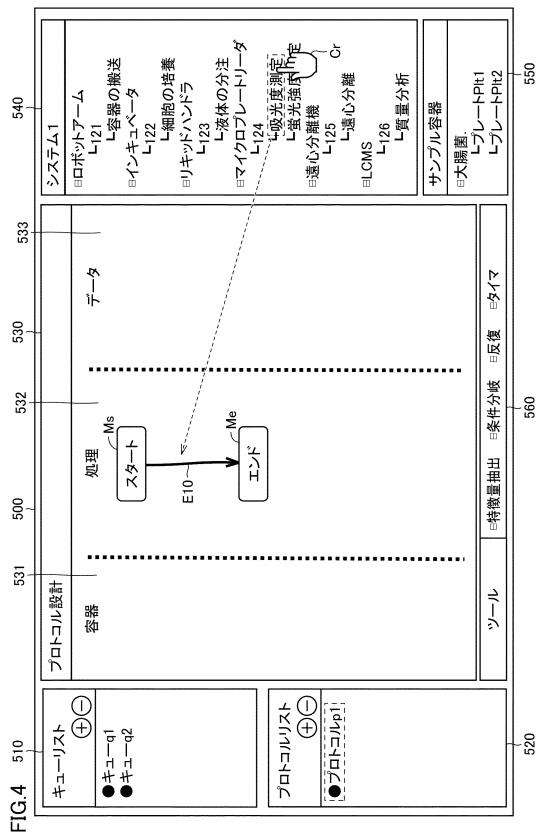


FIG.4

30

40

50

【図5】

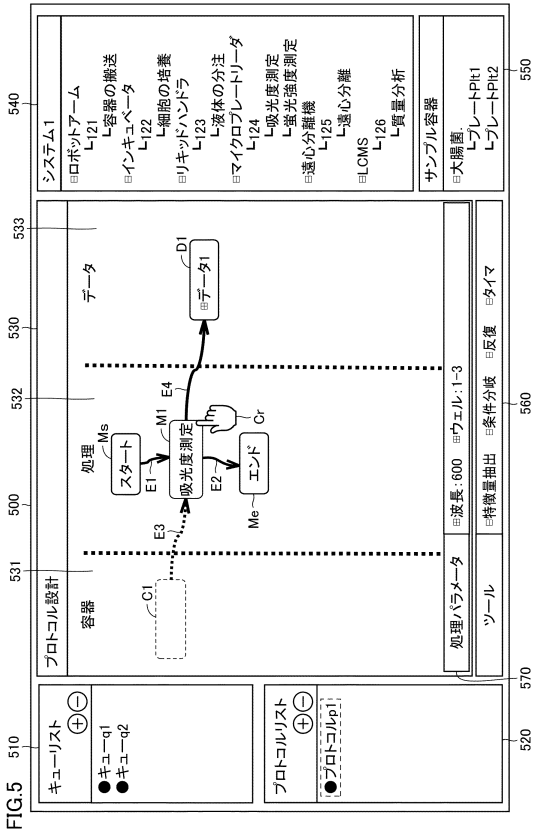
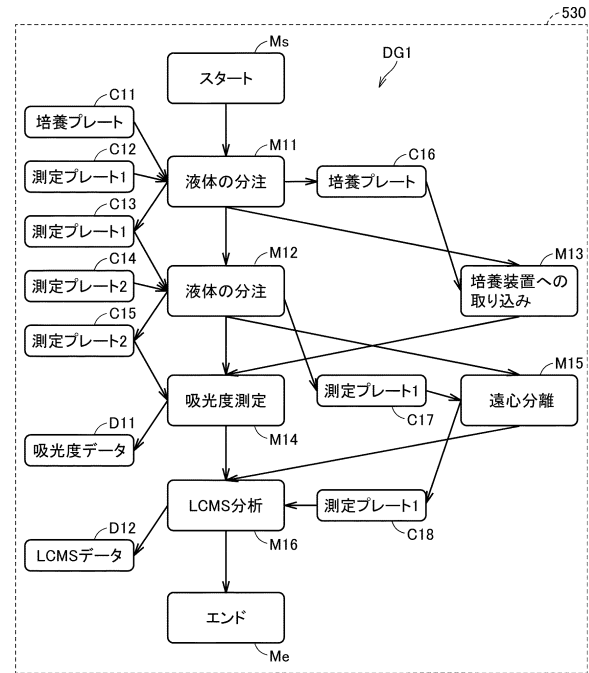


FIG.5

【図6】

FIG.6

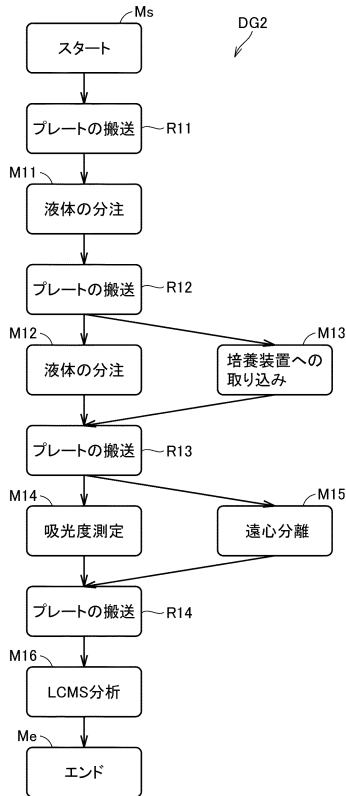


10

20

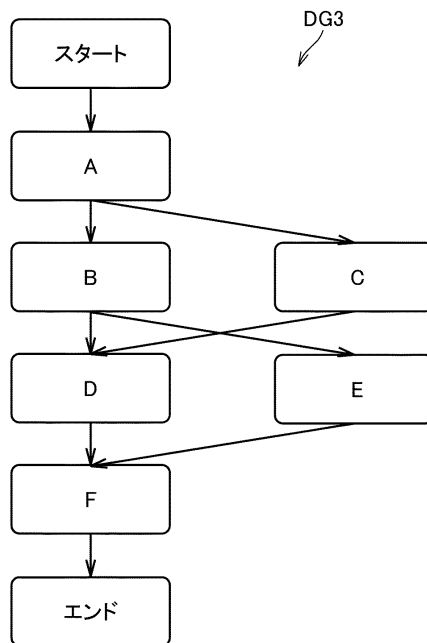
【図7】

FIG.7



【図8】

FIG.8



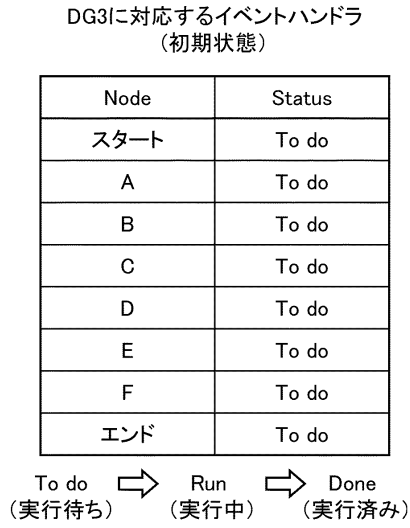
30

40

50

【 図 9 】

FIG.9



【 図 1 0 】

FIG.10

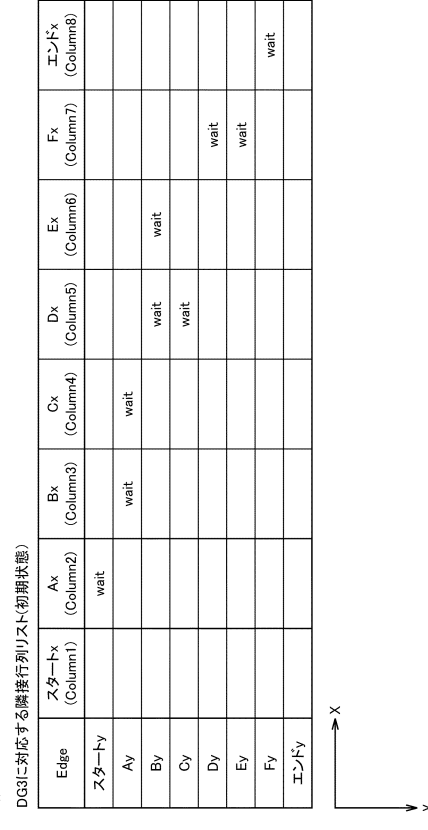
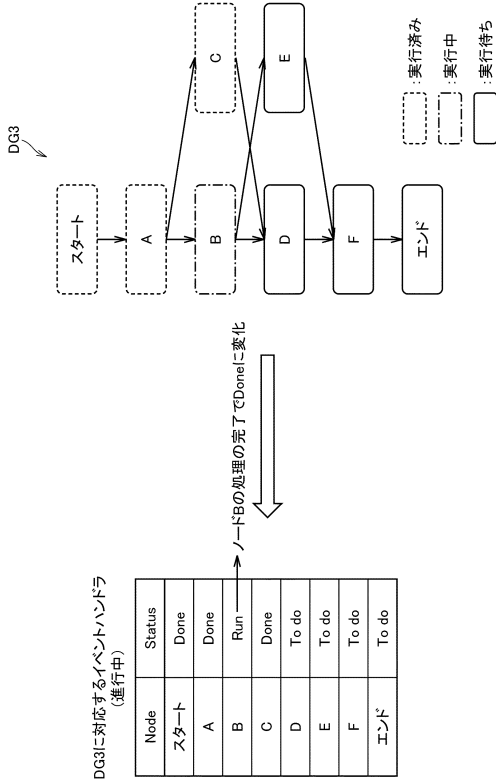


FIG.10

【 図 1 1 】

FIG.11



【 図 1 2 】

FIG.12

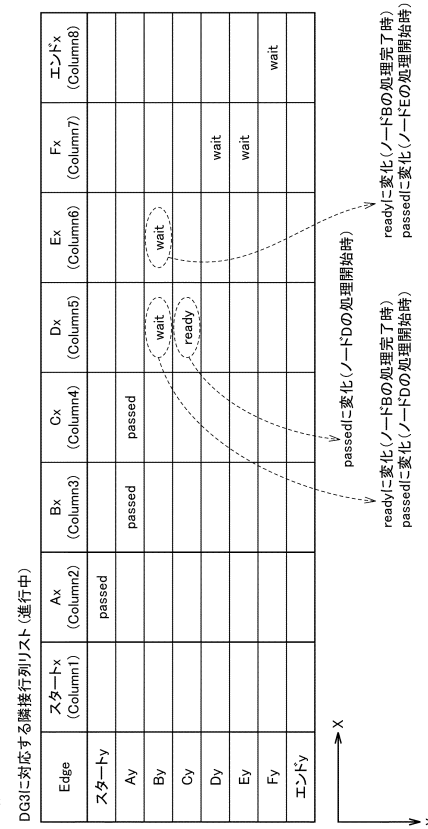
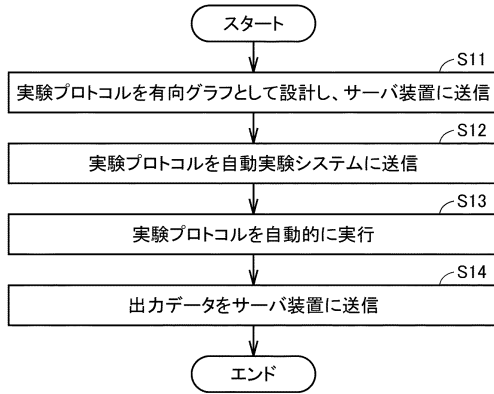


FIG.12

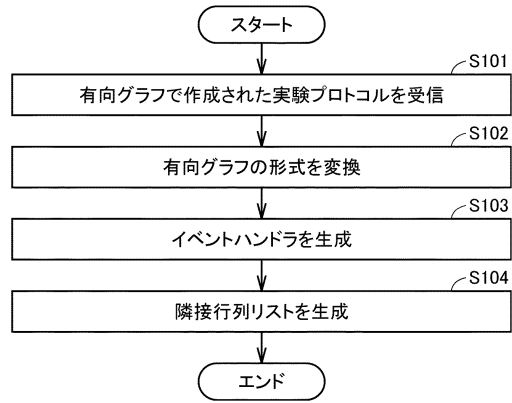
【図13】

FIG.13



【図14】

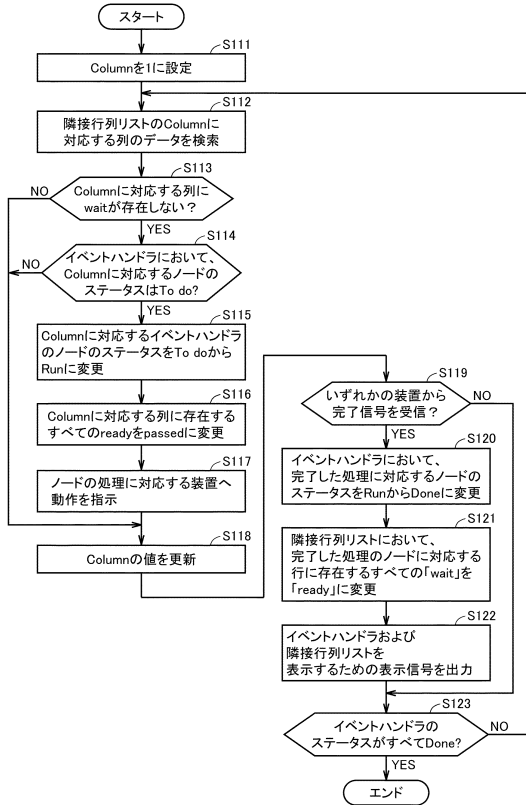
FIG.14



10

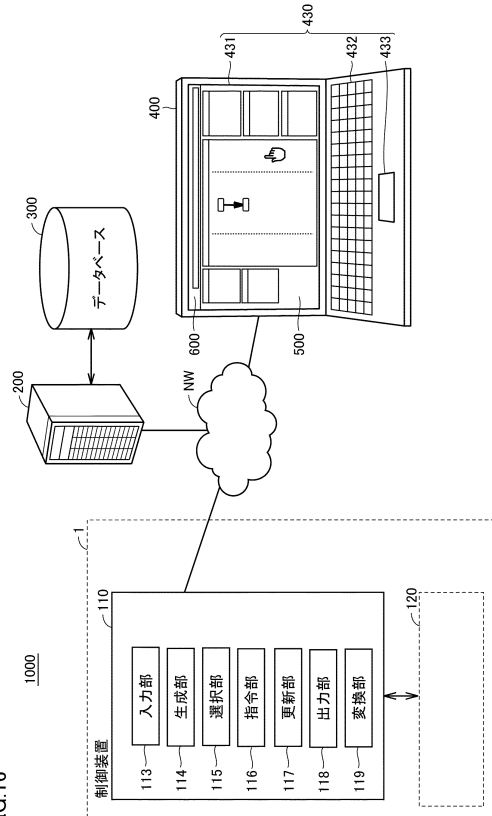
【図15】

FIG.15



【図16】

FIG.16



20

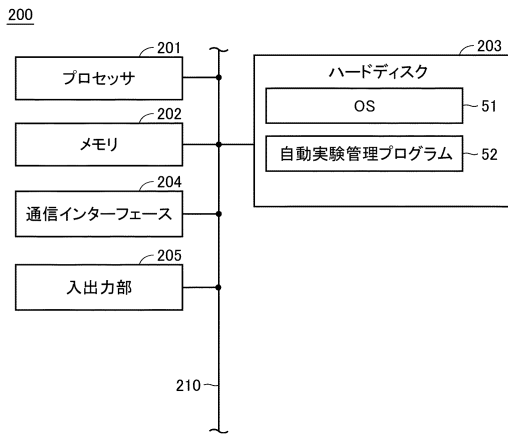
30

40

50

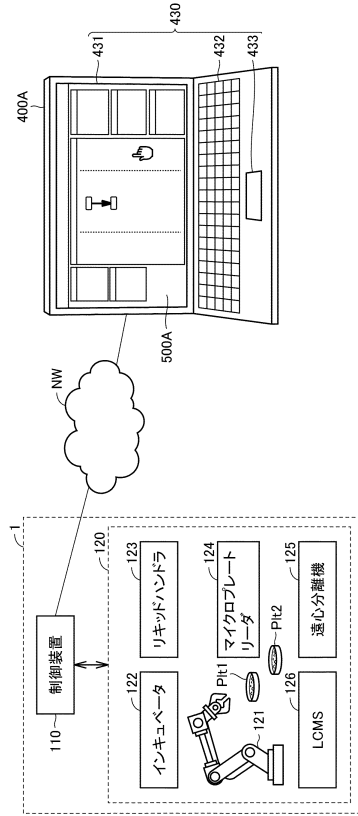
【図 17】

FIG.17



【図 18】

FIG.18

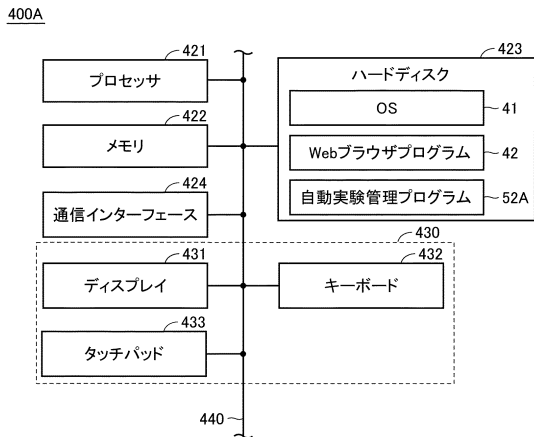


10

20

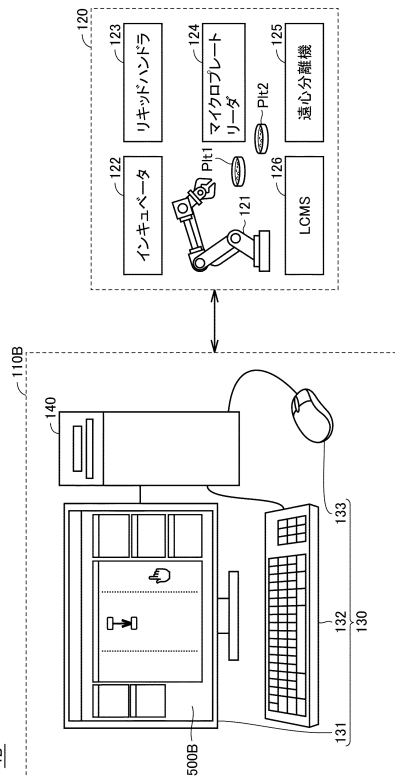
【図 19】

FIG.19



【図 20】

FIG.20



30

40

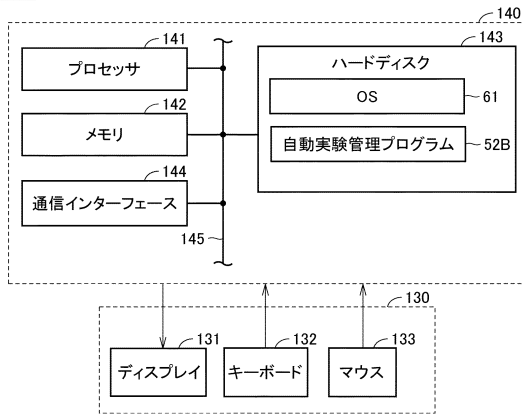
FIG.20

50

【 図 2 1 】

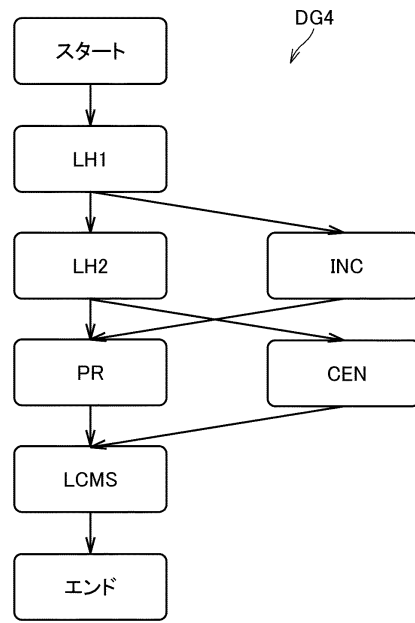
FIG.21

110B



【 図 2 2 】

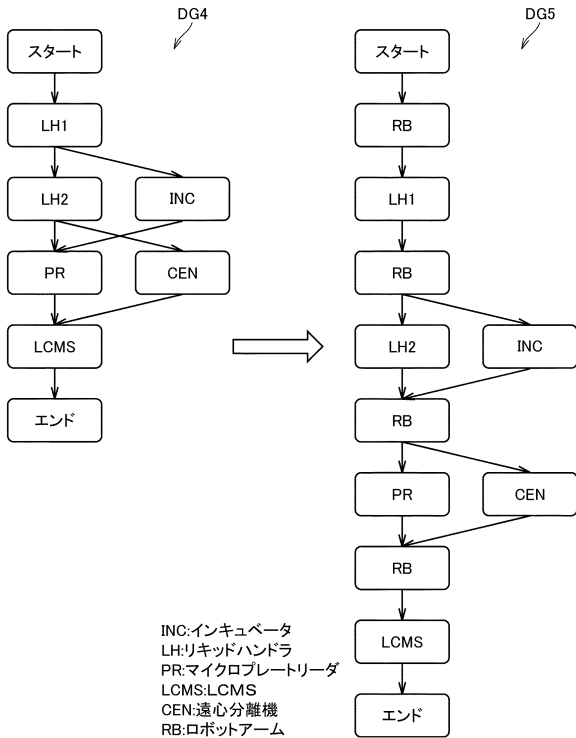
FIG.22



INC:インキュベータ
 LH:リキッドハンドラ
 PR:マイクロプレートリーダ
 LCMS:LCMS
 CEN:遠心分離機

【 図 2 3 】

FIG.23



INC:インキュベータ
 LH:リキッドハンドラ
 PR:マイクロプレートリーダ
 LCMS:LCMS
 CEN:遠心分離機
 RB:ロボットアーム

10

20

30

40

50

フロントページの続き

- (56)参考文献 特開2013-77335(JP,A)
国際公開第2021/5835(WO,A1)
特開2009-174997(JP,A)
国際公開第2021/38778(WO,A1)
特開2008-298798(JP,A)
国際公開第2020/183428(WO,A2)
国際公開第2022/4061(WO,A1)
- (58)調査した分野 (Int.Cl., DB名)
G01N 35/00-37/00