(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7562426号
(P7562426)

(45)発行日　令和6年10月7日(2024.10.7)

(24)登録日　令和6年9月27日(2024.9.27)

(51)国際特許分類

Ｇ１６Ｂ　30/10　(2019.01)	Ｇ１６Ｂ　30/10
Ｇ０６Ｆ　16/901(2019.01)	Ｇ０６Ｆ　16/901

ＦＩ

請求項の数　17　（全37頁）

(21)出願番号	特願2020-572925(P2020-572925)
(86)(22)出願日	令和1年10月29日(2019.10.29)
(65)公表番号	特表2022-503435(P2022-503435A)
(43)公表日	令和4年1月12日(2022.1.12)
(86)国際出願番号	PCT/US2019/058476
(87)国際公開番号	WO2020/092309
(87)国際公開日	令和2年5月7日(2020.5.7)
審査請求日	令和4年10月28日(2022.10.28)
(31)優先権主張番号	62/753,786
(32)優先日	平成30年10月31日(2018.10.31)
(33)優先権主張国・地域又は機関	米国(US)

(73)特許権者　500358711
イルミナ インコーポレイテッド
アメリカ合衆国 カリフォルニア州 ９２１２２ サンディエゴ イルミナ ウェイ ５２００

(74)代理人　100108453
弁理士　村山 靖彦

(74)代理人　100110364
弁理士　実広 信哉

(74)代理人　100133400
弁理士　阿部 達彦

(72)発明者　ジャオ， チェン
アメリカ合衆国 カリフォルニア ９２１２２， サンディエゴ， イルミナ ウェイ ５２００， イルミナ， インコーポレイ

最終頁に続く

(54)【発明の名称】　配列決定リードのグルーピングおよびコラプシングのためのシステムおよび方法

(57)【特許請求の範囲】

【請求項１】

ヌクレオチド配列決定リードを識別するためのシステムであって、前記システムは、

以下を格納するように構成された一時的でないメモリ：

実行可能な命令、

配列決定リードの複数のペアを格納するための第１のハッシュデータ構造および第２のハッシュデータ構造；ならびに

以下を行うために、前記実行可能な命令によってプログラムされたハードウェアプロセッサ：

第１の問い合わせヌクレオチド配列決定リードおよび第２の問い合わせヌクレオチド配列決定リードのペアを受容する工程；

複数の第１の問い合わせ識別子サブシーケンスおよび複数の第２の問い合わせ識別子サブシーケンスを、それぞれ、前記第１の問い合わせヌクレオチド配列決定リードおよび前記第２の問い合わせヌクレオチド配列決定リードから生成する工程；

前記第１の問い合わせヌクレオチド配列決定リードおよび前記第２の問い合わせヌクレオチド配列決定リードに関する第１の問い合わせシグナチャーおよび第２の問い合わせシグナチャーを、それぞれ、前記複数の第１の問い合わせ識別子サブシーケンスおよび前記複数の第２の問い合わせ識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；ならびに

１またはこれより多くの第１の格納したペアおよび１またはこれより多くの第２

の格納したペアを、それぞれ、前記第１のハッシュデータ構造および前記第２のハッシュデータ構造から、前記第１の問い合わせシグナチャーおよび前記第２の問い合わせシグナチャーを使用して回収する工程であって、ここで各回収され、格納したペアは、第１の格納したヌクレオチド配列決定リードおよび第２の格納したヌクレオチド配列決定リードを含む工程；ならびに

前記第１の格納したペアおよび第２の格納したペアの両方に存在する第１の格納したヌクレオチド配列決定リードおよび第２の格納したヌクレオチド配列決定リードの各ペアを、配列決定リード１および配列決定リード２として決定する工程、

を含む、システム。

【請求項２】

配列決定リードの各ペアは、第１のヌクレオチド配列決定リードおよび第２のヌクレオチド配列決定リードを含み、配列決定リードの各ペアは、前記第１のヌクレオチド配列決定リードの第１の識別子配列の第１の識別子サブシーケンスのハッシングによって生成した前記ペアの第１のヌクレオチド配列決定リードの第１のシグナチャーに基づいて、前記第１のハッシュデータ構造の複数の第１のビンのうちの１つに割り当てられ、配列決定リードの各ペアは、前記第２のヌクレオチド配列決定リードの第２の識別子配列のハッシングによって生成した前記ペアの第２のヌクレオチド配列決定リードの第２のシグナチャーに基づいて、前記第２のハッシュデータ構造の複数の第２のビンのうちの１つに割り当てられる、請求項１に記載のシステム。

【請求項３】

前記ハードウェアプロセッサは、

配列決定リードの各ペアに関して：

複数の第１の識別子サブシーケンスを、配列決定リードの前記ペアのうちの前記第１のヌクレオチド配列決定リードの第１の識別子配列から生成する工程；

前記第１のヌクレオチド配列決定リードに関する第１のシグナチャーを、前記複数の第１の識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；および

配列決定リードの前記ペアを、前記第１のシグナチャーに基づいて、前記第１のハッシュデータ構造のうちの少なくとも１つの第１の特定のビンに割り当てる工程；および

前記第１のハッシュデータ構造の各第１の特定のビンに関するヌクレオチド配列を、前記１またはこれより多くのペアのうちの前記第１のヌクレオチド配列決定リードおよび前記第２のヌクレオチド配列決定リードから割り当てられた第１のヌクレオチド配列決定リードおよび第２のヌクレオチド配列決定リードの１またはこれより多くのペアとともに決定する工程、

を行うために前記実行可能な命令によってプログラムされている、請求項２に記載のシステム。

【請求項４】

配列決定リードの前記ペアを割り当てる工程が、

前記第１のシグナチャーの複数のサブシーケンスを、配列決定リードの前記ペアのうちの前記第１のヌクレオチド配列決定リードの前記第１のシグナチャーから決定すること、および

配列決定リードの前記ペアを、前記第１のシグナチャーのサブシーケンスに基づいて、複数の第１のハッシュデータ構造のうちの各第１のハッシュデータ構造の第１の特定のビンに割り当てること

を含む、請求項３に記載のシステム。

【請求項５】

配列決定リードの前記ペアを割り当てる工程が、

前記第１のシグナチャーの複数のサブシーケンスを、配列決定リードの前記ペアのうちの前記第１のヌクレオチド配列決定リードの前記第１のシグナチャーから決定すること、および

配列決定リードの前記ペアを、前記第１のシグナチャーの前記複数のサブシーケンスに基づいて、前記第１のハッシュデータ構造の複数の第１の特定のビンに割り当てることを含む、請求項３に記載のシステム。

【請求項６】

前記第１の特定のビンは、前記第１のハッシュデータ構造に存在するビンであり、配列決定リードの前記ペアまたは配列決定リードの前記ペアの１つ、および前記第１のハッシュデータ構造の前記第１の特定のビンに割り当てられた配列決定リードの別ペアまたはその１つの配列決定リードのアラインメントスコアは、アラインメントスコア閾値を上回る、前記第１の特定のビンは、前記第１のハッシュデータ構造に存在するビンであり、配列決定リードの前記ペアまたはその１つの配列決定リード、および前記第１のハッシュデータ構造の前記第１の特定のビンに割り当てられたヌクレオチド配列決定リードの任意のペアまたはその１つの配列決定リードの最高のアラインメントスコアは、アラインメントスコア閾値を上回る、ならびに／または
前記第１の特定のビンは、前記第１のハッシュデータ構造の新しいビンであり、配列決定リードの前記ペアまたはその１つの配列決定リード、および前記第１のハッシュデータ構造の任意に存在するビンに割り当てられた配列決定リードの任意のペアまたはその１つの配列決定リードのアラインメントスコアは、アラインメントスコア閾値を下回る、
請求項３に記載のシステム。

【請求項７】

前記第１のシグナチャーは、前記第１のハッシュデータ構造の前記第１の特定のビンのキーにマッチする、および／または前記第１のシグナチャーおよび前記第１のハッシュデータ構造の前記第１の特定のビンの前記キーは、同一である、請求項３に記載のシステム。

【請求項８】

前記ヌクレオチド配列を決定する工程は、前記第１の特定のビンに割り当てられた配列決定リードの前記１またはこれより多くのペアのコンセンサス配列を決定することを包含し、必要に応じて、前記コンセンサス配列を決定することは、前記第１の特定のビンに割り当てられた最高の品質スコアを有する配列決定リードのペアまたは配列決定リードの前記ペアの１つを、前記第１の特定のビンの前記コンセンサス配列として決定することを包含する、請求項３に記載のシステム。

【請求項９】

前記ヌクレオチド配列を決定する工程は、前記第１の特定のビンに割り当てられた配列決定リードの前記１またはこれより多くのペアの配列を、前記第１の特定のビンの代表的配列として選択することを包含する、請求項３に記載のシステム。

【請求項１０】

前記ヌクレオチド配列を決定する工程は、前記第１の特定のビンに割り当てられた配列決定リードの前記１またはこれより多くのペアのうちの２つのアラインメントスコアがアラインメントスコア閾値を上回ることを決定することを包含する、請求項３に記載のシステム。

【請求項１１】

複数の前記ヌクレオチド配列決定リードは、同一の物理的識別子配列と関連するか、または複数の前記ヌクレオチド配列決定リードは、いかなる物理的識別子配列とも関連しない、請求項３に記載のシステム。

【請求項１２】

配列決定リードの各ペアは、第１の識別子配列および第２の識別子配列と関連し、前記ハードウェアプロセッサは、

配列決定リードの第１のペアの前記第１の識別子配列および前記第２の識別子配列、ならびに配列決定リードの第２のペアの前記第２の識別子配列および第１の識別子配列が同一であることを決定する工程；ならびに

配列決定リードの前記第１のペアおよび配列決定リードの前記第２のペアのヌクレオチド配列を決定する工程、

を行うために前記実行可能な命令によってプログラムされている、請求項２に記載のシステム。
【請求項１３】
前記第１の問い合わせヌクレオチド配列決定リードおよび前記第２の問い合わせヌクレオチド配列決定リードはペアードエンドヌクレオチド配列決定リードである、請求項１に記載のシステム。
【請求項１４】
ヌクレオチド配列決定リードを識別するためのコンピューターで実行される方法であって、前記方法は、
第１の問い合わせヌクレオチド配列決定リードおよび第２の問い合わせヌクレオチド配列決定リードのペアを受容する工程；
複数の第１の問い合わせ識別子サブシーケンスおよび複数の第２の問い合わせ識別子サブシーケンスを、それぞれ前記第１の問い合わせヌクレオチド配列決定リードおよび前記第２のヌクレオチド配列決定リードから生成する工程；
それぞれ前記第１の問い合わせヌクレオチド配列決定リードおよび前記第２の問い合わせヌクレオチド配列決定リードに関する第１の問い合わせシグナチャーおよび第２の問い合わせシグナチャーを、それぞれ前記複数の第１の問い合わせ識別子サブシーケンスおよび前記複数の第２の問い合わせ識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；および
１またはこれより多くの第１の格納したヌクレオチド配列決定リードおよび１またはこれより多くの第２の格納した配列決定リードを、第１のハッシュデータ構造から、前記第１の問い合わせシグナチャーおよび前記第２の問い合わせシグナチャーを使用して回収し、それにより前記第１の格納したヌクレオチド配列決定リードおよび前記第２の格納した配列決定リードを、前記第１の問い合わせヌクレオチド配列決定リードおよび前記第２の問い合わせヌクレオチド配列決定リードから識別する工程、
を包含する方法。
【請求項１５】
１またはこれより多くの第１の格納したヌクレオチド配列決定リードおよび１またはこれより多くの第２の格納した配列決定リードを回収する工程が、前記第１のハッシュデータ構造から１またはこれより多くの第１の格納したペアを回収すること、配列決定リードの複数のペアを格納すること、前記第１の問い合わせシグナチャーおよび前記第２の問い合わせシグナチャーを使用することを含む、請求項１４に記載の方法。
【請求項１６】
請求項１から１３のいずれか一項に記載のシステムまたは請求項１４または１５に記載の方法により、ヌクレオチド配列決定リードからヌクレオチド配列を決定するためのシステムであって、前記システムは、
実行可能な命令および複数のビンの中にヌクレオチド配列決定リードを格納するための第１のハッシュデータ構造を格納するように構成された一時的でないメモリ；ならびに
以下を行うために、前記実行可能な命令によってプログラムされたハードウェアプロセッサ：
複数の第１のヌクレオチド配列決定リードを受容する工程；
各第１の問い合わせヌクレオチド配列決定リードに関して：
複数の第１の識別子サブシーケンスを、前記第１の問い合わせヌクレオチド配列決定リードの第１の識別子配列から生成する工程；
前記第１の問い合わせヌクレオチド配列決定リードに関する第１のシグナチャーを、前記複数の第１の識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；および
前記第１のヌクレオチド配列決定リードを、前記第１のシグナチャーに基づいて、前記第１のハッシュデータ構造のうちの少なくとも１つの第１の特定のビンに割り当てる工程；ならびに

前記第１のハッシュデータ構造の各第１の特定のビンに関するヌクレオチド配列を、割り当てられた１またはこれより多くの第１のヌクレオチド配列決定リードとともに決定する工程、
を含むシステム。
【請求項１７】
請求項１から１５のいずれか一項に記載の方法により、ヌクレオチド配列決定リードからヌクレオチド配列を決定するためのコンピューターで実行される方法であって、前記方法は、
複数の第１のヌクレオチド配列決定リードを受容する工程；
各第１のヌクレオチド配列決定リードに関して：
複数の第１の識別子サブシーケンスを、前記第１のヌクレオチド配列決定リードの第１の識別子配列から生成する工程；
前記第１のヌクレオチド配列決定リードに関する第１のシグナチャーを、前記複数の第１の識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；および
前記第１のヌクレオチド配列決定リードを、前記第１のシグナチャーに基づいて、第１のデータ構造の第１の特定のビンに割り当てる工程；および
前記第１のデータ構造の各第１の特定のビンに関するヌクレオチド配列を、割り当てられた１またはこれより多くの第１のヌクレオチド配列決定リードとともに決定する工程、
を包含する方法。
【発明の詳細な説明】
【技術分野】
【０００１】
関連出願への相互参照
本出願は、２０１８年１０月３１日出願の米国仮出願第６２／７５３，７８６号（その内容は、その全体において参考として援用される）に基づく優先権を主張する。
【０００２】
著作権の表示
この特許文書の開示の一部は、著作権保護に従う資料を含む。著作権者は、特許文書または特許の開示が特許商標庁の特許ファイルおよび記録にあることから、特許文書または特許の開示を何人かが複製することに異議はないが、そうでなければ、どのようなものでも、全ての著作権の権利を留保する。
【０００３】
背景
分野
本開示は、一般に、ヌクレオチド配列決定データを処理する、およびより詳細には、局所性鋭敏型ハッシュを使用するヌクレオチド配列決定データのコラプシング(collapsing)という分野に関する。
【背景技術】
【０００４】
関連技術の説明
リードコラプシングは、同じ供給源のデオキシリボ核酸（ＤＮＡ）分子から発生するとして、配列決定システムからの出力であるヌクレオチド配列決定リードを識別する計算(computational)方法である。上記配列決定システムは、Ｉｌｌｕｍｉｎａ，　Ｉｎｃ．（Ｓａｎ　Ｄｉｅｇｏ，　ＣＡ）のＮｅｘｔＳｓｅｑ機器のような次世代シーケンシング（ＮＧＳ）システムであり得る。リードコラプシングは、これらのリードセットにおいて見出される疑似エラーを低減する統計的方法を使用することを含み得る。リードコラプシングの結果のインシリコエラー低減は、超低アレル画分を有するバリアントの検出のような次世代シーケンシング（ＮＧＳ）内での適用のために、および臨床適用のための強調されたバ

リアント呼び出し特異性を可能にするにあたって有用であり得る。
【発明の概要】
【課題を解決するための手段】
【０００５】
要旨
配列決定リードをコラプシングし、かつ複数の異なる配列決定リードにおいて類似のヌクレオチド配列を識別するためのシステムおよび方法が、本明細書で開示される。一実施形態において、システムは、ラン可能な命令および複数のビンの中にヌクレオチド配列決定リードを格納するための第１のハッシュデータ構造を格納するように構成された一時的でないメモリを含む。上記システムはまた、以下を包含する方法を行うために、上記実行可能な命令によってプログラムされたハードウェアプロセッサ：複数のヌクレオチド配列決定リード（例えば、ペアードエンド配列決定リードのヌクレオチド配列決定リード１）を受容する工程；各ヌクレオチド配列決定リードに関して： 複数の第１の識別子サブシーケンスを、上記ヌクレオチド配列決定リードの第１の識別子配列から生成する工程；上記ヌクレオチド配列決定リードに関する第１のシグナチャーを、上記複数の第１の識別子サブシーケンスに対してハッシュ化（ハッシング）を適用することによって生成する工程；ならびに上記ヌクレオチド配列決定リードを、上記第１のシグナチャーに基づいて、上記第１のハッシュデータ構造のうちの少なくとも１つの第１の特定のビンに割り当てる工程；ならびに上記第１のハッシュデータ構造の各第１の特定のビンに関するヌクレオチド配列を、割り当てられた１またはこれより多くのヌクレオチド配列決定リードとともに決定する工程、を含み得る。
【０００６】
本発明の別の実施形態は、複数のヌクレオチド配列決定リード（例えば、ヌクレオチド配列決定リード）を受容する工程；各ヌクレオチド配列決定リードに関して： 複数の第１の識別子サブシーケンスを、上記ヌクレオチド配列決定リードの第１の識別子配列から生成する工程；上記ヌクレオチド配列決定リードに関する第１のシグナチャーを、上記複数の第１の識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；ならびに上記ヌクレオチド配列決定リードを、上記第１のシグナチャーに基づいて、第１のデータ構造の第１の特定のビンに割り当てる工程；ならびに上記第１のデータ構造の各第１の特定のビンに関するヌクレオチド配列を、割り当てられた１またはこれより多くのヌクレオチド配列決定リードとともに決定する工程を包含するコンピューターで実行される方法である。
【０００７】
さらに別の実施形態は、類似のヌクレオチド配列決定リードを識別するためのシステムおよび方法を包含する。一例において、システムは、以下を格納するように構成された一時的でないメモリ：実行可能な命令、配列決定リードの複数のペアを格納するための第１のハッシュデータ構造および第２のハッシュデータ構造；ならびに以下を包含する方法を行うために、上記実行可能な命令によってプログラムされたハードウェアプロセッサ：第１の問い合わせ(query)ヌクレオチド配列決定リードおよび第２の問い合わせヌクレオチド配列決定リードのペアを受容する工程；複数の第１の問い合わせ識別子サブシーケンスおよび複数の第２の問い合わせ識別子サブシーケンスを、それぞれ、上記第１の問い合わせヌクレオチド配列決定リードおよび上記第２の問い合わせヌクレオチド配列決定リードから生成する工程、を含む。上記第１および第２の問い合わせヌクレオチド配列決定リードは、ペアードエンド配列決定リードのペアのリードであり得る。上記方法は、上記第１のヌクレオチド配列決定リードおよび第２のヌクレオチド配列決定リードに関する第１の問い合わせシグナチャーおよび第２の問い合わせシグナチャーを、それぞれ、上記複数の第１の問い合わせ識別子サブシーケンスおよび上記複数の第２の問い合わせ識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；１またはこれより多くの第１の格納したペアおよび１またはこれより多くの第２の格納したペアを、それぞれ、上記第１のハッシュデータ構造および上記第２のハッシュデータ構造から、上記第１の

問い合わせシグナチャーおよび上記第２の問い合わせシグナチャーを使用して回収する工程であって、ここで上記第１のペアおよび上記第２のペアの各々は、第１の格納したヌクレオチド配列決定リードおよび第２の格納したヌクレオチド配列決定リードを含む工程；ならびに上記第１の格納したペアおよび第２の格納したペアの両方に存在する第１の格納したヌクレオチド配列決定リードおよび第２の格納したヌクレオチド配列決定リードの各ペアを、それぞれ、上記第１の問い合わせ配列決定リードおよび上記第２の問い合わせ配列決定リードに類似の第１の配列決定リードおよび第２の配列決定リードとして決定する工程を包含し得る。

【０００８】

別の実施形態は、第１の問い合わせヌクレオチド配列決定リードおよび第２の問い合わせヌクレオチド配列決定リードのペアを受容する工程；複数の第１の問い合わせ識別子サブシーケンスおよび複数の第２の問い合わせ識別子サブシーケンスを、それぞれ、上記第１の問い合わせヌクレオチド配列決定リードおよび上記第２の問い合わせヌクレオチド配列決定リードから生成する工程；上記第１のヌクレオチド配列決定リードおよび上記第２のヌクレオチド配列決定リードに関する第１の問い合わせシグナチャーおよび第２の問い合わせシグナチャーを、それぞれ、上記複数の第１の問い合わせ識別子サブシーケンスおよび上記複数の第２の問い合わせ識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；ならびに１またはこれより多くの第１の格納したペアを第１のハッシュデータ構造から回収する工程、配列決定リードの複数のペアを、上記第１の問い合わせシグナチャーおよび上記第２の問い合わせシグナチャーを使用して格納する工程であって、ここで前記第１のペアの各々は、上記第１の問い合わせヌクレオチド配列決定リードおよび上記第２の問い合わせヌクレオチド配列決定リードに類似の第１の格納したヌクレオチド配列決定リードおよび第２の格納したヌクレオチド配列決定リードを含む工程、を包含するコンピューターで実行される方法である。

【０００９】

本明細書に記載される主題の１またはこれより多くの実行の詳細は、添付の図面および以下の説明に示される。他の特長、局面、および利点は、説明、図面、および請求項から明らかになる。この要旨も以下の詳細な説明も、本発明の主題の範囲を規定することも限定することも主張しない。

【図面の簡単な説明】

【００１０】

【図１】図１は、配列決定リードのコラプシングの模式図である。

【００１１】

【図２－１】図２Ａ～２Ｄは、局所性鋭敏型ハッシュに基づくリードグルーピングおよびコラプシングの模式図を示す。この模式図は、屋根を葺くように並べる（ｓｈｉｎｇｌｉｎｇ）（図２Ａ）、最小ハッシュ化（図２Ｂ）、局所性鋭敏型ハッシュ（ＬＳＨ）挿入（図２Ｃ）、およびＬＳＨ問い合わせ（図２Ｄ）を含む。問い合わせ配列決定リード（示さず）が与えられると、上記問い合わせ配列決定リードの最小ハッシュまたはシグナチャー２３２が生成され得、上記最小ハッシュ２３２は、ハッシュ表１および２（２２４ａ，２２４ｂ）に対して問い合わせするために使用される２つのチャック２３２ａ、２３２ｂへと分割され得る。上記問い合わせ配列決定リードは、ハッシュ表１および２（２２４ａ，２２４ｂ）に格納される配列ＡＣＴＧＧＡＣ ２０４に類似であるが、同一ではない。上記問い合わせ配列決定リードの最小ハッシュ２３２は、最小ハッシュ２１２が、どのようにして図２Ａおよび２Ｂに図示される配列決定リードＡＣＴＧＧＡＣ（２０４）に関して生成されるかに類似して生成される。ハッシュ表１（２２４ａ）は、既存のビン２２８ａのキー２１２ａとしてチャックのうちの一方２３２ａを含まないので、上記問い合わせ配列決定リードに類似の配列決定リードは、ハッシュ表１（２２４ａ）において見出されない。ハッシュ表２（２２４ｂ）は、既存のビン２２８ｂのキー２１２ｂとしてチャックのうちの一方２３２ｂを含むので、上記問い合わせ配列決定リードは、既存のビン２２８ｂと関連する配列決定リード２０４に類似である。次いで、既存のビン２２８ｂに格納さ

れる配列ＡＣＴＧＧＡＣ（２０４）は、上記問い合わせ配列決定リードに類似の配列決定リードとして戻される。
【図２－２】同上。
【図２－３】同上。
【図２－４】同上。
【００１２】
【図３】図３は、ペアードエンド配列決定リードのリード１およびリード２に関する仮想ユニバーサル分子インデックス（ｖＵＭＩ）を生成する模式図を示す。
【００１３】
【図４】図４Ａおよび４Ｂは、仮想ＵＭＩからｋマー（図４Ａ）およびタイル状のｋマー（図４Ｂ）を生成する模式図を示す。
【００１４】
【図５】図５は、配列決定リードを、それらの配列およびそれらのハッシュを介してビニングするフロー図を示す。各物理的ＵＭＩバーコードは、１２０の考えられる物理的ＵＭＩバーコードのうちの１つから選択され得、その結果、配列決定リード１およびリード２の各ペアに関する物理的ＵＭＩバーコードの１２０×１２０＝１４，４００の組み合わせが存在する。フロー図において、配列決定リード１およびリード２の各ペアは、上記ペアの物理的ＵＭＩバーコードの組み合わせに基づいて、１４，４００　ＵＭＩビンのうちの各々に割り当てられ得る。本開示のリードコラプシング方法は、各ＵＭＩビンの配列決定リードに適用され得る。
【００１５】
【図６－１】図６Ａ～６Ｆは、局所性鋭敏型ハッシュを伴うリードコラプシングをよびアラインメントベースのリードコラプシングが類似の成績を有することを示す例示的プロットである。
【図６－２】同上。
【図６－３】同上。
【図６－４】同上。
【図６－５】同上。
【図６－６】同上。
【００１６】
【図７】図７は、局所性鋭敏型ハッシュを使用するリードコラプシングの例示的方法を示すフロー図である。
【００１７】
【図８】図８は、局所性鋭敏型ハッシュを使用する類似のリードを識別する例示的方法を示すフロー図である。
【００１８】
【図９】図９は、局所性鋭敏型ハッシュでリードコラプシングおよび問い合わせを実行するように構成された例証的コンピューティングシステムのブロック図である。
【発明を実施するための形態】
【００１９】
詳細な説明
　以下の詳細な説明において、その一部を形成する添付の図面に対して言及がなされる。図面において、類似の記号は、代表的には、状況が別段規定しなければ、類似の構成要素を識別する。詳細な説明、図面、および請求項に記載される例証的実施形態は、限定であることを意味されない。他の実施形態が利用されてもよく、本明細書で示される主題の趣旨または範囲から逸脱することなく、他の変更が行われてもよい。本開示の局面は、本明細書で概して記載され、図面の中で図示されるように、広く種々の異なる構成において配置され得、置換され得、組み合わされ得、分離され得、デザインされ得ることは、容易に理解される。これら構成は全て、本明細書で明示的に企図され、本明細書中の開示の一部を構成する。

【００２０】
概説
リードコラプシングは、同じ供給源のデオキシリボ核酸（ＤＮＡ）分子に由来するヌクレオチド配列決定リードを識別し、その後、統計的方法を使用して、これらのリードセットにおいて見出される疑似エラーを低減する計算方法である。図１を参照すると、プラス鎖１０８ａおよびマイナス鎖１０８ｂを有する同じＤＮＡ分子１０８の重複のリード１０４＋ｒ１、１０４＋ｒ２、１０４－ｒ１、１０４－ｒ２を全て与えると、リードコラプシングは、それらのリード１０４＋ｒ１、１０４＋ｒ２、１０４－ｒ１、１０４－ｒ２を一緒にグルーピングする工程を包含し得る。リードコラプシングは、統計的投票（ｓｔａｔｉｓｔｉｃａｌ　ｖｏｔｉｎｇ）を使用して、疑似エラーを低減する、例えば、単純コラプシングを用いて、ヌクレオチド鎖のヌクレオチド配列（例えば、ＤＮＡ分子１０８のプラス鎖１０８ａの配列）を決定する工程を包含し得る。リードコラプシングは、高忠実度を有する元のＤＮＡ分子１０８の配列を推測する工程、例えば、二重コラプシングを用いて、プラス鎖１０８ａの配列およびマイナス鎖１０８ｂの配列の両方から、ＤＮＡ分子１０８のヌクレオチド配列を決定する工程を包含し得る。本明細書で開示されるシステムおよび方法は、リードコラプシングのために局所性鋭敏型ハッシュ（ＬＳＨ）および仮想識別子配列（ｖＩＤ配列）を利用し得る。
【００２１】
リードコラプシングは、高品質リードを生成し得る。リードコラプシングは、サンプルが、識別子配列（ＩＤ配列）１１２ａ、１１２ｂ’、１１２ａ’、１１２ｂとともに配列決定されることを要求し得る。このような識別子配列はまた、本明細書で「物理的識別子配列（ｐｈｙｓｉｃａｌ　ｉｄｅｎｔｉｆｉｅｒ　ｓｅｑｕｅｎｃｅ）」（ｐＩＤ　ｓｅｑｕｅｎｃｅｓ）といわれる。これらの識別子配列は、ユニバーサル分子インデックス（ＵＭＩ）バーコードであり得る。このような識別子配列１１２ａ、１１２ｂ’、１１２ａ’、１１２ｂは、別の点では非常に類似であるように思われ得るリードおよび分子を区別する場合に増大した分離能を可能にするが、リードコラプシングは、特定の環境下でこのような識別子配列なしに行われ得る。リードコラプシングは、インシリコ（ｉｎ－ｓｉｌｉｃａ）エラー低減を生じ得る。このようなエラー低減は、次世代シーケンシング（ＮＧＳ）内での多くの適用のために有用であり得る。
【００２２】
このプロセスの１つの適用は、超低アレル画分に（例えば、循環する腫瘍ＤＮＡ（ｃｔＤＮＡ）に）存在するに過ぎないバリアントの検出である。別の適用は、臨床適用に関する強調したバリアント呼び出し特異性である。リードコラプシングは、ＤＮＡフラグメントの重複の観察（例えば、ＤＮＡフラグメントのＰＣＲ複製）全てを、単一の代表へと効果的に組み合わせるので、リードコラプシングは、下流で処理される必要のあるデータ量を顕著に低減するという利点を有する。重複の観察、またはリードを除去することは、データサイズにおいて１０倍の、またはこれより大きな減少を生じ得る。
【００２３】
ナイーブリードコラプシング法は、徹底的なペアワイズ配列比較を含み得、これは、ランタイムＯ（n^2）を要する。Ｏ（n^2）は、ＮＧＳデータにとって打破しがたい。例えば、およそ６億のリードペア（6×10^8）が、サンプルから生成され得る。徹底的なペアワイズ配列比較は、3.6×10^{17}の比較を要し得る。１つの比較／ナノ秒という速度においてすら、3.6×10^8秒（これは、およそ４，１６７日に等しい）が、上記ペアワイズ比較を計算するために要求される。５６プロセシングコアを用いてすら、１つのサンプルを分析するために２ヶ月超という計算時間がなおかかる。
【００２４】
従来のリードコラプシング方法は、アラインメント位置およびＵＭＩバーコード情報の組み合わせを使用して、重複のリードのグループを識別し得る。リードコラプシングのこれらの方法のうちの１つの否定的側面は、このような方法が、既に整列されかつソートされた入力リードを要求することである。リードコラプシングの前に整列およびソードされ

ているリードの前処理と関連する難題が存在する。第１に、従来のリード処理は、Ｏ（ｎ×ｌｏｇ（ｎ））の前処理を要する。「ｎ」個のリードを整列させることは、Ｏ（ｎ）というランタイムを要求し得る。「ｎ」個のリードをソートすることは、Ｏ（ｎ×ｌｏｇ（ｎ））というランタイムを要求する。「ｎ」個のリードを整列およびソートするという前処理は、Ｏ（ｎ×ｌｏｇ（ｎ））というランタイムを要求する。

【００２５】

従来の方法を使用してリードをコラプシングするプロセス全体は、Ｏ（ｎ×ｌｏｇ（ｎ））という下界のランタイムを有する。この超線形関数は、ランタイムが入力サイズより早くなることを意味する。生の配列決定データがより大きく生成されるにつれて、従来のリードコラプシングと関連する計算コストは、処理されている配列決定データの量より早く大きくなる。さらに、従来のリードコラプシングは、十分に特徴づけられた参照に拘束される。リードコラプシングをヌクレオチドアラインメントに基づかせると、その使用法は、高品質参照配列が存在する適用に拘束される。なぜならこれらの参照配列は、良好なアラインメントおよびその後の正確なコラプシングを可能にするために必要であるからである。例えば、合理的に良好なアラインメントは、構造的バリアント、反復拡大、および反復ゲノム領域にとって利用可能でないこともある。アラインメントに対する依存性は、新規または未知の種に対してリードコラプシング技術（およびＵＭＩ技術）を使用することを困難にし、ＵＭＩが可能にしたエラー低減技術の一般性を制限する。リードコラプシングは、配列決定エラーを識別することに役立つ高品質「参照」が存在しないので、これらの場において大きな影響を有し得る。

【００２６】

局所性鋭敏型ハッシュ（ＬＳＨ）を使用するリードコラプシングのためのシステムおよび方法が、本明細書で開示される。従来のリードコラプシングアプローチは、アラインメント情報を要求した。なぜならＵＭＩバーコードを既に共有したリードのサブグルーピングのための良好な代替法が存在しなかったからである。本明細書で開示されるとおりのＬＳＨベースのリードコラプシング方法は、この欠点に対処し、アラインメントを有することに対するコラプシングの依存性を除去し、アルゴリズムの複雑性、ならびにリードコラプシングおよび関連技術の一般的適用における大きな利益を可能にする。

【００２７】

一実施形態において、上記ＬＳＨベースのリードコラプシング方法は、Ｏ（ｎ×ｌｏｇ（ｎ））というランタイムを有さず、十分に特徴づけられた参照を要求しない。例えば、上記リードコラプシング方法は、アラインメント情報を全く、ましてやソートされたアラインメントを要求しなくてもよい。１つの実行において、上記方法は、物理的識別子配列（ｐＩＤ配列）（例えば、物理的ＵＭＩバーコード（ｐＵＭＩバーコード））、および仮想識別子配列（ｖＩＤ配列）（例えば、仮想ＵＭＩバーコード（ｖＵＭＩバーコード））に依拠して、種々のヌクレオチドフラグメントに存在する重複したリードのグループを識別する。物理的識別子配列はまた、本明細書で「識別子配列（ｉｄｅｎｔｉｆｉｅｒ　ｓｅｑｕｅｎｃｅ）」（ＩＤ配列）といわれる。物理的ＵＭＩバーコードはまた、本明細書で「ＵＭＩバーコード（ＵＭＩ　ｂａｒｃｏｄｅ）」といわれる。仮想識別子配列は、重複したリードのグループを識別するために、「仮想（ｖｉｒｔｕａｌ）識別子配列として作用するリードのサブシーケンスであり得る。ｖＵＭＩバーコードは、重複したリードのグループを識別するために、「仮想」バーコードとして作用するリードのサブシーケンスであり得る。物理的識別子配列または物理的ＵＭＩバーコードは、配列決定ライブラリー調製の間に、ヌクレオチドフラグメントに付加される識別子配列またはバーコードであり得る。

【００２８】

一実施形態において、上記方法は、類似のリードを一緒にグルーピングし、いかなる参照配列も徹底的な配列比較も要求しない。上記方法は、ＵＭＩバーコード（例えば、物理的ＵＭＩ、仮想ＵＭＩ、またはこれらの組み合わせ）によって規定されるビンへとリードの一次通過のナイーブグルーピングを決定する工程を包含し得る。上記方法は、仮想ＵＭ

Ｉと局所性鋭敏型ハッシュとを組み合わせる。上記方法は、類似の配列を有するリードが、それらのアラインメント情報なしに一緒にグルーピングされることを可能にし、上記方法は、リードコラプシングのプロセスをアラインメントの拘束から切り離す。上記方法は、そのビンにおいて他の配列決定リードをチェックすることを含む、類似の配列を決定する工程を包含し得る。

【００２９】

上記方法は、サンプルが由来する生物に拘わらず、任意のサンプル（例えば、ＤＮＡまたはＲＮＡ）からのリードをコラプシングするために使用され得る。さらに、ハッシュ化は、Ｏ（１）一定時間オペレーションであり、かつハッシュ化は、ｎ個のリードの各々に対して固定された時間数で行われる必要があるので、上記方法は、Ｏ（ｎ）ランタイムにおいて実行するリードコラプシングを可能にする。例えば、ハッシュ化は、１回、２回、３回、またはより多く、ｎ個のリードの各々に対して行われなければならない。このようなリードコラプシングランタイムは、ＮＧＳで生成されるデータのますます大きなセットに対して要求される処理時間を低減する。Ｏ（ｎ）ランタイムは、二次分析のランタイム複雑性における顕著な低減を可能にし、任意のサンプルへのリードコラプシングの融通の利く適用を可能にする。

【００３０】

一実施形態において、ＤＮＡ配列決定機器は、本明細書で開示されるＬＳＨベースのリードコラプシング方法を実行し得る。例えば、上記方法は、インシリコエラー低減のための機器上の方法（ｏｎ－ｉｎｓｔｒｕｍｅｎｔ　ｍｅｔｈｏｄ）として実行され得る。なぜなら上記方法は、リードコラプシングのために参照配列を要求しないからである。上記方法は、ＮＧＳの重複率をてこ入れしてエラー低減を実施することによって、全ての配列決定リードに関して大きく低減したエラー率を達成し得る。上記方法はまた、ユーザーが処理しなければならない配列決定データの量を顕著に低減し得、従って、ゲノム分析のアクセス性を増大させる。上記方法は、顧客にとってより少ない高品質リードを出力し、下流分析の複雑性を低減するシーケンサー上での技術として利用され得る。

【００３１】

一実施形態において、上記配列決定リードは、ＵＭＩバーコードと関連しないか、またはＵＭＩバーコードを使用して生成される。例えば、局所性鋭敏型ハッシュは、ヌクレオチドリードのグルーピングするために、仮想ＵＭＩに対して行われ得る。別の例として、物理的ＵＭＩバーコードによって提供されるビニング機能性を模倣するために、「層になった（ｔｉｅｒｅｄ）」仮想ＵＭＩストラテジーが使用され得る。上記方法は、２タイプの仮想ＵＭＩ（一方は、物理的ＵＭＩの模倣として使用され、一方は、仮想ＵＭＩとして使用される）を生成し得る。

【００３２】

重複マーキングは、ＰＣＲによって導入されるバイアスを低減するためのバイオインフォマティクス方法である。本明細書で開示されるものは、類似のリード配列を一緒にグルーピングし、局所性鋭敏型ハッシュで重複を作成するためのシステムおよび方法を含む。

【００３３】

リードコラプシング

本明細書で開示されるものは、リードコラプシングのためのシステムおよび方法を含む。一実施形態において、上記方法は、仮想識別子（ｖＩＤ）配列（例えば、仮想ユニバーサル分子インデックス（ｖＵＭＩ））を、局所性鋭敏型ハッシュとともに使用して、徹底的なペアワイズ比較を行うことなく、類似のリードの参照なのグルーピングを可能にする。配列決定リードの仮想識別子配列（例えば、仮想ＵＭＩ）は、潜在的に不連続の部分ストリングを含む、配列決定リード自体の内部にある任意の部分ストリングまたはサブシーケンスに言及する。仮想識別子配列は、物理的識別子（ｐＩＤ）配列とは異なる。物理的識別子配列（例えば、物理的ＵＭＩ（ｐＵＭＩ））は、配列決定ライブラリー調製の間に付加される識別子配列またはＵＭＩバーコードに言及する。

【００３４】

局所性鋭敏型ハッシュ（ＬＳＨ）は、徹底的なペアワイズ比較を行うことなく、「類似の（ｓｉｍｉｌａｒ）」データを、同じ計算「ビン（ｂｉｎ）」の中に置く計算法である。データ類似性は、レーベンシュタイン距離、ハミング距離、またはジャッカード距離のような測定規準とともに計算され得る。リードの配列類似性に言及する。上記ＬＳＨ関数は、各リードと関連する仮想ＵＭＩを「ハッシュ（ｈａｓｈ）」し得、その結果を使用して、類似の仮想ＵＭＩを有するリードの側に、各リードをビンの中に置く。本明細書で開示されるように配列決定リードに適用されるとおりのＬＳＨは、エラーを含む仮想ＵＭＩを有する配列決定リードが、仮想ＵＭＩに基づいて一緒にグルーピングされることを可能にする。配列決定技術はしばしば、エラーのない配列決定リードを生成しない。従って、小さな変異を有し得る類似の配列を一緒にビニングしかつ迅速に見出すことができることは、リードコラプシングを行うために必要なリードグルーピングを実施するために重要である。配列決定リード中の小さな変異はしばしば予測し難いことから、類似の配列決定リードのグルーピングのための一般的方法は、特定の変異パターンを仮定する類似の配列決定リードのグルーピングのための特定の方法よりも有用であり得る。いくつかの実行において、本明細書で開示される方法の、エラーが全体に影響を与えない特性（ｅｒｒｏｒ　ｔｏｌｅｒａｎｔ　ｐｒｏｐｅｒｔｉｅｓ）は、配列決定リードのうちのおよそ２０％において作用し始める。訂正されないままである場合、これらの配列決定リードは、コラプシング精度に大きく影響を与え得、その後、それ自体がバリアント呼び出しにおいて過度の偽陽性として現れ得る。

【００３５】

一実施形態において、類似の配列決定リードは、仮想ＵＭＩを、例えば、辞書式順序で（ｌｅｘｉｃｏｇｒａｐｈｉｃａｌｌｙ）ソートすることによって、配列決定リードから生成される仮想ＵＭＩに基づいて識別され得る。仮想ＵＭＩをソートすることに基づくリードコラプシング方法は、仮想ＵＭＩにおける変異を説明しなくてもよく、Ｏ（ｎ×ｌｏｇ（ｎ））というランタイム複雑性を有してもよい。別の実施形態において、類似の配列決定リードは、ナイーブ規準的ハッシュ化を仮想ＵＭＩとともに使用して識別され得る。ナイーブ規準的ハッシュ化に基づくリードコラプシング方法は、ＬＳＨベースのリードコラプシング方法と比較して類似の成績を有し得る。このようなリードコラプシング方法は、エラーが全体に影響を与えない特性を有しない。一実施形態において、類似の配列決定リードは、配列決定リードと関連するＵＭＩをクラスター化することによって識別され得る。クラスター化ベースのリードコラプシング方法は、わずかにミスマッチの仮想ＵＭＩバーコードを取り扱い得るが、Ｏ（n^2）ペアワイズ比較を必要とし、これは、従来のリードコラプシング方法のランタイムに関するＯ（ｎ×ｌｏｇ（ｎ））およびＬＳＨベースのリードコラプシング方法のＯ（ｎ）ランタイムより顕著に悪い。

【００３６】

ＬＳＨは、確率的データプロセスである。ＬＳＨで類似のリードデータを同じビンの中に置く確率は、高い。類似のデータが同じビンの中に入らない確率は、小さく、ゼロではない。類似のデータが同じビンの中に入らないか、または異なるデータが同じビンの中に入る見込みは、小さいかもしれない。一実施形態において、ＬＳＨは、類似のリードデータを同じビンの中に置く確率をＬＳＨで最小にするためにデザインされる。一実施形態において、上記ＬＳＨベースのリードコラプシング方法は、最大呼び戻し（ｒｅｃａｌｌ）および各ビンにおける各項目（通常は、５項目より少ない）に関するアラインメントベースのチェックを行うように構成され得る。

【００３７】

<u>配列に適用される局所性鋭敏型ハッシュ</u>

局所性鋭敏型ハッシュ（ＬＳＨ）は、各データ片（例えば、データの仮想ＵＭＩ）を「ハッシュ」関数に通過させ、その結果は、そのデータをビンの中に入れるために使用される。ＬＳＨを用いると、類似のデータは、同じ（または近い）ビンの中に入り、類似のデータに関する非常に迅速な問い合わせを可能にする。ＬＳＨベースのリードコラプシングは、屋根を葺くように並べること、最小ハッシュ化、および局所性鋭敏型ハッシュを含み

得る。屋根を葺くように並べることは、入力データを重なり合うセットまたは長さｋの文字のシングル（ｓｈｉｎｇｌｅ）へと整理することを含む。最小ハッシュ化は、各「シングル」を、ハッシュ関数のセットに通過させて、データのフィンガープリントまたはシグナチャーを生成することを含む。局所性鋭敏型ハッシュは、上記フィンガープリントを使用して、データを、類似のデータが類似のビニングスキームを共有するようである「ビン」に置くことを含む。例えば、配列決定リードは、上記配列決定リードの長さｋのサブシーケンスへと整理され得る。各サブシーケンスは、ハッシュ関数を通過して、上記配列決定リードのシグナチャーを生成し得る。上記シグナチャーは、上記配列決定リードを、類似の配列決定リードが類似のまたは同一のシグナチャーを共有するようである１またはこれより多くのビンに置くために使用され得る。

【００３８】

図２Ａ～２Ｄは、屋根を葺くように並べること（図２Ａ）、最小ハッシュ化（図２Ｂ）、局所性鋭敏型ハッシュ（ＬＳＨ）挿入（図２Ｃ）、およびＬＳＨ問い合わせ（図２Ｄ）を含む、局所性鋭敏型ハッシュベースのリードグルーピングおよびコラプシングの模式図を示す。屋根を葺くように並べることは、ｍ塩基対増分ずつｋ塩基のスライドするウインドウを動かし、従って、仮想ＵＭＩ２０４をｋマーの「シングル」２０８ａ～２０８ｄ（図２Ａ）に整理することを含む。例えば、４塩基のスライドするウインドウは、１塩基対増分ずつ動かされて、仮想ＵＭＩ２０４を４マーのシングルへと整理し得る。図２Ａは、配列の４つのシングルを示す。第１のシングルはＧＧＡＣ（２０８ａ）であり、第２のシングルはＴＧＧＡ（２０８ｂ）であり、第３のシングルはＣＴＧＧ（２０８ｃ）であり、第４のシングルはＡＣＴＧ（２０８ｄ）である。

【００３９】

最小ハッシュ化は、シングルのｋマーセットに関するハッシュ「シグナチャー（ｓｉｇｎａｔｕｒｅ）」２１２を、上記セットをいくつかのハッシュ関数２１６に通過させ、最小ハッシュ（ＭｉｎＨａｓｈ）を採用することによって、生成することを包含する。図２Ｂは、４つのシングル、ＧＧＡＣ、ＴＧＧＡ、ＣＴＧＧ、およびＡＣＴＧのセットが、８個のハッシュ関数２１６へと通過して、各シングルに関して、それぞれ、上記８個のハッシュ関数の８要素の出力２２０ａ～２２０ｄを生成し得ることを図示する。ハッシュ関数の数は、８、１６、３２、６４、１２８、２５６、５１２、１０２４、またはこれより多い可能性がある。そのハッシュ出力２２０ａ～２２０ｄの相当する要素の最小数は、上記最小数の最小ハッシュ２１２を計算するために採用され得る。上記最小ハッシュのジャッカード距離は、真のジャッカード距離の近似値である。ハッシュ関数がより多くなるほど、その近似値もよりよくなる。

【００４０】

ここで図２Ｃを参照すると、上記配列決定リード２０４は、上記最小ハッシュ２１２、または上記配列決定リード２０４から計算された上記最小ハッシュ２１２のサブシーケンス２１２ａ、２１２ｂに基づいて、ハッシュ表２２４ａ、２２４ｂに挿入され得る。図２Ｃは、配列決定リード２０４が、上記最小ハッシュ２１２のサブシーケンス２１２ａ、２１２ｂに基づいて、ハッシュ表２２４ａ、２２４ｂに挿入され得ることを図示する。配列決定リード２０４のＬＳＨ挿入は、ハッシュ「シグナチャー」２１２を費やし、次いで、上記シグナチャー２１２を、塊またはサブシーケンス２１２ａ、２１２ｂへと分割する。それらの塊２１２ａ、２１２ｂは、次いで、ハッシュ表２２４ａ、２２４ｂにおいて、特に、上記ハッシュ表２２４ａ、２２４ｂのビン２２８ａ、２２８ｂにおいて、キーとして使用される。この分割およびハッシュ化スキームは、ビンの「幅広さ（ｗｉｄｅｎｅｓｓ）」に関して、およびより高次の呼び戻しまたはより高次の特異性に関して調節可能である。図２Ｃに示されるように、同じ配列決定リード２０４は、２つの異なるハッシュ表２２４ａ、２２４ｂにおいて２つの異なるビンの中に入れられる。２つの配列決定リードが１またはこれより多くのビンの中で共有する、または格納される限りにおいて、上記配列決定リードは、類似であると考えられ得る。

【００４１】

図２Ｄは、問い合わせ配列決定リードが、ＬＳＨを使用して、ハッシュ表の中に格納された配列決定リードに類似であるか、または同一であるかを決定することを図示する。問い合わせ配列決定リードが与えられる場合、上記システムは、問い合わせ配列決定リードを最小ハッシュ化に通過させ、全てのハッシュ表に対して最小ハッシュを問い合わせる。上記問い合わせ配列のシグナチャー２３２は、２つの塊２３２ａ、２３２ｂに分割され得、これらは、全てのハッシュ表２２４ａ、２２４ｂに対して問い合わせられる。上記ハッシュ表２（２２４ｂ）が、チャックのうちの一方２３２ｂを、既存のビン２２８ｂのキー２１２ｂとして含むので、上記問い合わせ配列は、既存のビン２２８ｂと関連する（例えば、中に格納される）上記配列決定リード２０４に類似または同一である。

【００４２】

配列に適用される局所性鋭敏型ハッシュ

仮想ＵＭＩは、配列決定リード自体のサブシーケンスである。仮想ＵＭＩは、ヌクレオチドリード全体にまで及び得、連続するサブシーケンスであっても、連続していないサブシーケンスであってもよい。例えば、配列決定リードの仮想ＵＭＩは、上記配列決定リードの５’末端から２５塩基対（ｂｐｓ）であり得る。図３は、ペアードエンド配列決定リードのリード１（Ｒ１）およびリード２（Ｒ２）に関する仮想ユニバーサル分子インデックス（ｖＵＭＩ）を生成する模式図を示す。配列決定されているＤＮＡフラグメント３０８のプラス鎖またはマイナス鎖に相当するペアードエンド配列決定リードのリード１　３０４ｒ１およびリード２　３０４ｒ２は、仮想ＵＭＩ　３１２ｒ１、３１２ｒ２を生成するために処理され得る。

【００４３】

図４Ａおよび４Ｂは、仮想ＵＭＩからｋマー（図４Ａ）およびタイル状のｋマー（図４Ｂ）を生成する模式図を示す。２セットのｋマーのジャッカード類似性は、両方のセットが含むｋマーの数である。２つの同一の仮想ＵＭＩのジャッカード類似性は、７個のｋマー４１２が上記仮想ＵＭＩに関して生成される場合、７／７であり得る。図４Ａに図示されるように、仮想ＵＭＩ４０４および４０８の挿入を除いて同じ仮想ＵＭＩ　４０４ｉのジャッカード類似性は、２／１２であり得る。なぜなら挿入４０８を含み得る全てのｋマー４１２ａまたは上記挿入の３’は、影響を及ぼされるからである。仮想ＵＭＩにわたって重なり合うｋマーの使用は、挿入および欠失のより良好な寛容を可能にする。図４Ｂに図示されるように、図４Ａに図示される同じ挿入を用いると、仮想ＵＭＩ　４０４および挿入４０８を除いて同じ仮想ＵＭＩ　４０４ｉのジャッカード類似性は、１６個のｋマー４１２が、仮想ＵＭＩ　４０４、４０４ｉに関して生成される場合、１４／１８で、依然として高い。上記ジャッカード類似性は、挿入４０８を含むｋマー４１２ａのみが挿入４０８によって影響を受けることから、依然として極めて高い。屋根を葺くように並べた後の類似性が大きくなるほど、より類似のＭｉｎＨａｓｈシグナチャーがもたらされ、これは、つぎに、同じＬＳＨビンにおいて変化した配列が加わるのを助ける。

【００４４】

一実施形態において、ＬＳＨ自体は、コラプシングされるべきリードのグループを直接格納もインデックスもしない。むしろ、ＬＳＨは、所定の問い合わせリードに対して類似のリードを見出すことを補助する。類似のリードがいったん見出されると、それは、キーが最初にそのグループを開始した「重心（ｃｅｎｔｒｏｉｄ）」リードである従来のハッシュ表の中に格納される。言い換えると、上記ＬＳＨデータ構造は、上述の従来のハッシュ表と同じ「キー」を含む（そしてその２つは、足並みを揃えて更新される）。ここでこれらの「キー」は、他のリードが類似である場合に割り当てられる「グループアンカー（ｇｒｏｕｐ　ａｎｃｈｏｒ）」として働くリードである。ＬＳＨは、入ってくるリードが、マッチする可能性がある上記キー／アンカーを迅速に見出すことを可能にする。その結果、徹底的なチェックまたは比較は要求されない。実際のグループまたは配列決定リードは、従来のハッシュ表の中に格納され得る。

【００４５】

アラインメントスコアチェック

アラインメントスコアチェックは、類似の配列に関してＬＳＨビンをチェックした後に行われ得る。上記アラインメントスコアチェックは、似ていない仮想ＵＭＩが同等でないまたは類似とみなされないことを確実にする。仮想ＵＭＩに関する最小アラインメントスコアは、仮想ＵＭＩが等しいと考えられるために必要とされ得る。ＬＳＨマッチに対するアラインメントチェックを行う場合、アラインメントスコアが閾値を上回る（すなわち、最高のアラインメントスコアで）決定するために、最良のマッチが使用され得る。より多くのミスマッチ（例えば、単一ヌクレオチドバリアント（ＳＮＶ）ならびに挿入および欠失（インデル）は、アラインメントスコアを低減する。アラインメントを「スライドさせる」ことは、マッチの数を低減し、これは、次に、許容されるミスマッチの数を低減する。一実施形態において、仮想ＵＭＩの全体のアラインメントが行われ得、これは、計算するコストが高価であり得る。

【００４６】

ｖＵＭＩマッチングのための二重ビンＬＳＨ構造

各リードペアからの２つの仮想ＵＭＩは、「独立した（ｉｎｄｅｐｅｎｄｅｎｔ）」バーコードであるか、または同じＤＮＡフラグメントの同一性の独立した測定値である。各仮想ＵＭＩに関して２つの別個のＬＳＨデータ構造を維持することによって、偽陽性が低減され得る。より少ないＭｉｎＨａｓｈシグナチャーが、各ビンへと置かれるので、意図しない不一致の機会はより低い。２つの独立した問い合わせの結果を交差させることによって、大部分の残りの偽陽性ヒットは、正確なヒットを失うことなく除去され得る。

【００４７】

独立したＬＳＨデータ構造は、リード１および／またはリード２に関する仮想ＵＭＩのために使用され得る。ＬＳＨは、最適未満の特異性に伴う非常に高い呼び戻のために構成され得る。特異性を改善するために、２つの直交する問い合わせの間の交差は、感度に対して大きな影響なしに、特異性を改善するために採用され得る。仮想ＵＭＩをリード１およびリード２から分離することは、より反復性の領域における特異性も改善する。

【００４８】

一実施形態において、各ｋマーに対して６４のハッシュが存在し得る。リード１は、６４要素のｍｉｎＨａｓｈシグナチャーを有し、リード２は、６４要素のｍｉｎＨａｓｈシグナチャーを有する。リード１およびリード２のペアは、それぞれ、上記リード１のｍｉｎＨａｓｈシグナチャーおよび上記リード２のｍｉｎＨａｓｈシグナチャーに基づいて、二重ビンＬＳＨ構造の２つのハッシュ表の中に格納され得る。上記リード１のｍｉｎＨａｓｈシグナチャーは、サブシーケンスへと分割され得、その結果、リード１およびリード２のペアは、一方のハッシュ表の多数のビンの中に、および他方のハッシュ表の多数のビンの中に、格納され得る。ハッシュの数字は、実行が異なれば異なり得る（例えば、８、６４、２５６、１０２４、およびより多く）。より多くのハッシュは、わずかかにより遅いランタイムを犠牲にして、より正確なデータ構造成績のために使用され得、より少ないハッシュは、より速いランタイムを追加して、わずかに正確性の低いデータ構造成績のために使用され得る。あらゆるｋマーは、ｍｉｎｈａｓｈシグナチャーが一貫したサイズであることを確実にするために、６４回だろうが、８回だろうが、またはｎ回だろうが、同じ回数ハッシュされる。

【００４９】

一実施形態において、リード１およびリード２は、原子的で分離不能なユニットと考えられ、この原子的ユニットは、リード１からの仮想ＵＭＩのＭｉｎＨａｓｈシグナチャー、およびリード２からの仮想ＵＭＩのＭｉｎＨａｓｈシグナチャーの両方（ｖＵＭＩ　１およびｖＵＭＩ　２）によって参照される。各ハッシュ表内で、上記ＭｉｎＨａｓｈシグナチャーは、上記原子的リード１／リード２ペアが、上記シグナチャーの各チャックにつき１回、多数のビンの中に格納されるように、部分へと分割され得る。

【００５０】

二重ビンＬＳＨ構造の例示的な擬似コードは、以下に示される。

【数１】

```
Class DualBinLSH():
  self.matcher1 = LSH()
  self.matcher2 = LSH()

  func get_match(vUMI1, vUMI2):
    x = self.matcher1.query(vUMI1)
    y = self.matcher2.query(vUMI2)
    # Filter out spurious matches via vUMI alignment
    z = [items in intersect(x, y) with aln score ≥ cutoff]
    return argmax(z)  # return match with best aln score

  func has_match(vUMI1, vUMI2):
    if self.get_match(vUMI1, vUMI2) is not null/empty:
      return True
    return False

  func insert(vUMI1, vUMI2):
  self.matcher1.insert(vUMI1)
  self.matcher2.insert(vUMI2)
```

【００５１】

ＬＳＨにおいて二重ビンＬＳＨ構造を使用するという例示的な擬似コードは、以下に示される。類似の仮想ＵＭＩが二重ビンＬＳＨ構造の中に格納されていない仮想ＵＭＩのペアを挿入することによって、将来的な問い合わせが、挿入されたペアを見出すことが可能になる。本質的には、次回に対してマッチングされ得るファミリーのために、新たな「シード（ｓｅｅｄ）」が作られる。

【数２】

```
func bin_reads_by_virtual_UMI(reads w/ same UMIs):
  Matcher = DualBinLSH()
  Families = HashTable()
  for read_pair in reads:
    vUMI_1, vUMI_2 = read_pair.get_virtual_umis()
    if Matcher.has_match(vUMI_1, vUMI_2):
      match = Matcher.get_match(vUMI_1, vUMI_2)
      Families[match].extend(read_pair)
    else:
      Families[(vUMI_1, vUMI_2)] = List(read_pair)
      Matcher.insert(vUMI_1, vUMI_2)
  return Families
```

【００５２】

ｖＵＭＩマッチングのための二重ビンＬＳＨ構造

図５は、配列決定リードの、それらの配列およびそれらのハッシュを介するビニングのフロー図を示す。図５に図示されるビニングプロセスは、並列化可能であり、これは、数千スレッドで線形的高速化を生じ得る。図示されるビニングプロセスは、最小のメモリ使用率で中間キャッシングを必要とする。各物理的ＵＭＩバーコードは、１２０の考えられる物理的ＵＭＩバーコードのうちの１つから選択され得、その結果、物理的ＵＭＩバーコードの１２０×１２０＝１４，４００の組み合わせがあり、配列決定リード１およびリード２の各ペアは、物理的ＵＭＩバーコードの１４，４００の組み合わせのうちの１つを有し得る。ＬＳＨは、物理的ＵＭＩの各ペアと関連する上記配列決定リードに並行して適用され得る。

【００５３】

単純コラプシングは、同じ物理的＋仮想ＵＭＩペア形成を共有する全ての配列を同じ順序でコラプシングすることに言及する。これらの配列は同じ順序でそれらのバーコードを有することから、これは、これらの配列が、同じＤＮＡ分子に由来しただけでなく、そのＤＮＡ分子の同じ鎖にも由来したことを示す。単純コラプシングにおいて、グループ／ファミリーの中のあらゆるリードは、同じ第１のＵＭＩ、同じ第２のＵＭＩ、同じ第１の仮想ＵＭＩ、および同じ第２の仮想ＵＭＩを有する。多数のリードが、この条件を満たし得、この場合、それらは全て、同じ分子の同じ鎖に由来するリードであると考えられる。

【００５４】

単純コラプシングが行われた後に、二重コラプシングが行われ得る。二重コラプシングにおいて、コラプシングされたリードペアが与えられると、同じ物理的および仮想ＵＭＩペア形成を有する別の単純分子を逆の順序で見出そうとする試みが行われる － これは、その同じＤＮＡ分子の反対側の鎖を見出すことに類似である。このような二重マッチが見出される場合、二重コラプシングが行われる。

【００５５】

二重コラプシングは、単純コラプシングに引き続いて行われることから、全ての鎖特異的重複は、単純コラプシングで除去されている。二重コラプシングにおいて、同じ分子に由来する反対側の鎖からの既にコラプシングされたリードが見出される。例えば、第１のＵＭＩ　ｘ、第２のＵＭＩ　ｙ、第１の仮想ＵＭＩ　ａ、および第２の仮想ＵＭＩ　ｂとともにコラプシングされたリードペアが与えられると、二重コラプシングは、その第１のＵＭＩがｙであり、その第２のＵＭＩがｘであり、その第１の仮想ＵＭＩがｂであり、かつその第２の仮想ＵＭＩがａである上記反対側の鎖のリードペアを探す。上記反対側の鎖のリードペアの逆相補は、「ｆａｓｔｑ」ファイルのような出力ファイルにおいてどのようにリードが報告されるかというセマンティクスに起因して、必要とされないこともある。この二層構造の１本鎖、次いで、交差鎖（ｃｒｏｓｓ－ｓｔｒａｎｄ）コラプシングは、下流の分析においていくつかの高度バリアント呼び出し技術を可能にする。

【００５６】

結果

従来のアラインメントベースの方法と上記ＬＳＨ／仮想ＵＭＩベースの方法との間のリードコラプシング結果は、アラインメントサマリーメトリクスのレベルでも、およびバリアント呼び出し（例えば、構造的バリアント呼び出しおよび小さなバリアント呼び出し）に関していっても、両方で匹敵することが見出された。類似のヌクレオチド配列を見出すために、ビンの中の他の項目がチェックされ得る。本明細書で開示される仮想ＵＭＩベースの方法は、従って、バリアント呼び出しのための配列決定リードをコラプシングするために使用され得る。

【００５７】

図６Ａ～６Ｆは、局所性鋭敏型ハッシュを伴うリードコラプシングおよびアラインメントベースのリードコラプシングが、ホルマリン固定パラフィン包埋（ＦＦＰＥ）サンプルで実行したＮｅｘｔＳｅｑ（登録商標）（Ｉｌｌｕｍｉｎａ，　Ｉｎｃ．（Ｓａｎ　Ｄｉｅｇｏ，　ＣＡ））の類似の成績を有することを示す例示的プロットである。図６Ａは、ア

ラインメントベースのリードコラプシングおよびＬＳＨベースのリードコラプシング（これはまた、本明細書で「ｆａｓｔｑベースのコラプシング」または「ｆａｓｔｑリードコラプシング」ともいわれる）によって生成される極めて類似の標的領域適用率を示す。図６Ｂは、非参照証拠を有し、かつ小さなバリアント呼び出し成績のインジケーターであるゲノム遺伝子座の割合を判断するノイズＡＦを示す。ＬＳＨベースのリードコラプシングは、アラインメントコラプシングと類似のエラー検出および訂正能力を有した。図６Ｃは、ＳＶ呼び出し成績のインジケーターである、非線形的アラインメントを有するリードのパーセンテージを示す。ＬＳＨベースのリードコラプシングは、より少ない（～７％）キメラリードを生成した。生成したキメラリードがより少ないことは、ＬＳＨベースのリードコラプシングが、ノイズのあるリードへとより一般化され得、従って、よりきれいなアラインメントを生じるという証拠である。図６Ｄは、メガ塩基あたりの変異を判断する腫瘍遺伝子変異量（ｔｕｍｏｒ　ｍｕｔａｔｉｏｎ　ｂｕｒｄｅｎ）（ＴＭＢ）を示す。この図は、小さなバリアント呼び出しが、非常に一致したことを示す。ＬＳＨベースのリードコラプシングは、ゲノムのガイダンスがなくても、変異を運ぶリードを「野生型」リードから分離するのに問題がない。図６Ｅは、ゲノムの高度に反復性の領域における変異を判断するマイクロサテライト不安定性を示す。高度に反復性の領域は、配列複雑性／特有性の低さに起因して、取り扱うのが困難な領域である。ＬＳＨベースのリードコラプシングは、このような複雑性が低い領域にあっても機能した。図６Ｆは、ＬＳＨベースのリードコラプシングが、変動性のコピー数を有する領域にあっても十分に機能したことを示す。表１は、フュージョン呼び出しがこれらのサンプルにおいて予測されたように、フュージョン呼び出しがＬＳＨベースのリードコラプシングに伴って劇的に改善した特異性を示したことを示す。上記フュージョン呼び出し結果は、非線形リードの改善された取り扱いを示唆する。

【００５８】

表２は、フュージョン呼び出し特異性に対する改善が、ＮｏｖａＳｅｑ™（Ｉｌｌｕｍｉｎａ，　Ｉｎｃ．）ランの呼び戻しに否定的に影響しないことを示す。

表１．　フュージョン呼び出し偽陽性

【表１】

8 サンプルを横断	アラインメントベースの コラプサー	LSH ベースのコラプサー
フィルタ後の偽陽性	22	1

表２．　フュージョン呼び出し呼び戻し

【表２】

フュージョン 遺伝子ペア	ブレークポイント1	ブレークポイント2	サポートフュージョン（従来のReCo）	サポートWT（従来のReCo）	VF（従来のReCo）	サポートフュージョン（LSH ReCo）	サポートWT（LSH ReCo）	VF（LSH ReCo）
TMP3 | NTRK1	chr1:154137492	chr1:156843543	80	3280	2.44%	80	3264	2.45%
TMP3 | NTRK1	chr1:154137489	chr1:156843542	80	3287	2.43%	78	3279	2.34%
RET | CCDC6	chr10:43609948	chr10:61638611	68	2966	2.30%	70	2943	2.38%
ROS1 | SLC34A2	chr4:25666629	chr6:117658325	45	1387	3.24%	45	1373	3.28%
ROS1 | SLC34A2	chr4:25666625	chr6:117658307	3	1160	0.26%	3	1153	0.26%
ALK | NPM1	chr2:29447103	chr5:170819618	10	2428	0.41%	10	2400	0.42%
ALK | NPM1	chr2:29447024	chr5:170819667	6	2650	0.23%	6	2614	0.23%
ALK | EML4	chr2:29448092	chr2:42493956	10	2591	0.39%	11	2570	0.43%

【００５９】

全体として、これらのデータは、ＬＳＨベースのリードコラプシングが、アラインメントベースのコラプシングに都合良く匹敵し、サマリーレベルメトリクスおよびバリアント呼び出しで既存の成績に匹敵するかまたはこれを超えたことを示す。

【００６０】

リードコラプシング方法

図７は、局所性鋭敏型ハッシュを使用するリードコラプシングの例示的方法７００を示すフロー図である。方法７００は、コンピューティングシステムのコンピューター可読媒体（例えば、１またはこれより多くのディスクドライブ）に格納された実行可能なプログラム命令のセットの中に埋め込まれ得る。例えば、図９に示されかつ以下でより詳細に記載されるコンピューティングシステム９００は、方法７００を実行するために、実行可能なプログラム命令のセットを実行し得る。方法７００が開始される場合、上記実行可能なプログラム命令は、メモリ（例えば、ＲＡＭ）に搭載され得、コンピューティングシステム９００の１またはこれより多くのプロセッサによって実行され得る。方法７００が、図９に示されるコンピューティングシステム９００に関して記載されるが、その説明は例証に過ぎず、限定であることを意図しない。いくつかの実施形態において、方法７００またはその一部は、多数のコンピューティングシステムによって直列でまたは並列で行われ得る。コンピューティングシステム９００は、複数のビンにおいてヌクレオチド配列決定リードをソートするために、第１のハッシュデータ構造（例えば、ハッシュ表）を含み得る。

【００６１】

方法７００が、ブロック７０４で開始した後、方法７００は、ブロック７０８へと進み、ここでコンピューティングシステムは、複数の第１のヌクレオチド配列決定リードを受容する。上記複数の第１のヌクレオチド配列決定リードは、同一の物理的識別子配列と関連し得る。上記複数の第１のヌクレオチド配列決定リードは、任意の物理的識別子配列と関連し得る。

【００６２】

方法７００は、ブロック７０８からブロック７１２へと進み、ここで上記コンピューティングシステムは、複数の第１の識別子サブシーケンスを、各第１のヌクレオチド配列決定リードの第１の識別子配列から生成する。上記複数の第１の識別子サブシーケンスの生成は、上記配列決定リードの第１の識別子配列から複数のｋマーを生成することを包含し

得る。上記サブシーケンスは、ヌクレオチド挿入、ヌクレオチド欠失、ヌクレオチド置換、またはこれらの組み合わせを含み得る。２つの連続する第１の識別子サブシーケンスは、重なり合い得る。例えば、上記２つの連続する第１の識別子サブシーケンスは、ｋ－１ヌクレオチドが重なり合う。例えば、上記複数の第１の識別子サブシーケンスは、複数の４マーを含み、ここで上記第１の識別子配列は、約２５ヌクレオチドを含む。上記第１の識別子配列は、上記配列決定リード１のサブシーケンスであり得る。上記サブシーケンスは、上記配列決定リード１の連続するサブシーケンスであり得る。上記サブシーケンスは、上記配列決定リード１の非連続サブシーケンスであり得る。

【００６３】

ブロック７１２において第１の識別子サブシーケンスを生成した後、方法７００は、ブロック７１６へと進み、ここで上記コンピューティングシステムは、第１のヌクレオチド配列決定リードに関する第１のシグナチャーを、上記複数の第１の識別子サブシーケンスに対してハッシュ化を適用することによって生成する。上記第１のシグナチャーは、上記第１のハッシュデータ構造の第１の特定のビンのキーにマッチし得る。上記第１のシグナチャーおよび上記第１のハッシュデータ構造の第１の特定のビンのキーは、同一であり得る。

【００６４】

上記第１のシグナチャーを生成する工程は、各第１の識別子サブシーケンスに関する複数のハッシュを決定する工程を包含し得る。上記第１のシグナチャーを生成する工程は、上記第１のシグナチャーの各第１の要素を、上記複数の第１の識別子サブシーケンスの相当するハッシュから決定する工程を包含し得る。上記第１のシグナチャーの各第１の要素は、上記複数の第１の識別子サブシーケンスの相当するハッシュの最小であり得る。上記第１のシグナチャーの各第１の要素は、上記複数の第１の識別子サブシーケンスの相当するハッシュのうちの最小、平均、中間、または最大である。

【００６５】

上記方法は、ブロック７１６からブロック７２０へと進み、ここで上記コンピューティングシステムは、上記第１のヌクレオチド配列決定リードを、上記第１のシグナチャーに基づいて、上記第１のハッシュデータ構造のうちの少なくとも１つの第１の特定のビンに割り当てる。一実施形態において、上記第１のヌクレオチド配列決定リードを割り当てる工程は、上記第１のシグナチャーの複数のサブシーケンスを、上記第１のヌクレオチド配列決定リードの第１のシグナチャーから決定する工程；および上記第１のヌクレオチド配列決定リードを、上記第１のシグナチャーのサブシーケンスに基づいて、複数の第１のハッシュデータ構造の各第１のハッシュデータ構造の第１の特定のビンに割り当てる工程を包含する。別の実施形態において、上記第１のヌクレオチド配列決定リードを割り当てる工程は、上記第１のシグナチャーの複数のサブシーケンスを、上記第１のヌクレオチド配列決定リードの第１のシグナチャーから決定する工程；および上記第１のヌクレオチド配列決定リードを、上記第１のシグナチャーの複数のサブシーケンスに基づいて、上記第１のハッシュデータ構造の複数の第１の特定のビンに割り当てる工程を包含する。方法７００は、ブロック７２８で終了する。

【００６６】

一例において、上記第１の特定のビンは、上記第１のハッシュデータ構造に存在するビンであり、ここで上記第１のヌクレオチド配列決定リードおよび上記第１のハッシュデータ構造の第１の特定のビンに割り当てられた別の第１のヌクレオチド配列決定リードのシグナチャーのアラインメントスコアは、アラインメントスコア閾値を上回る。別の例では、上記第１の特定のビンは、上記第１のハッシュデータ構造に存在するビンであり、ここで上記第１のヌクレオチド配列決定リードおよび上記第１のハッシュデータ構造の第１の特定のビンに割り当てられた任意の第１のヌクレオチド配列決定リードのシグナチャーの最高のアラインメントスコアは、アラインメントスコア閾値を上回る。別の例では、上記第１の特定のビンは、上記第１のハッシュデータ構造の新しいビンであり、ここで上記第１のヌクレオチド配列決定リードおよび上記第１のハッシュデータ構造の任意に存在する

ビンに割り当てられた任意の第１のヌクレオチド配列決定リードのシグナチャーのアラインメントスコアは、アラインメントスコア閾値を下回る。

【００６７】

上記第１のヌクレオチド配列決定リードが、ブロック７２０において上記第１の特定のビンに割り当てられた後、方法７００は、ブロック７２４に進み、ここで上記コンピューティングシステムは、上記第１のハッシュデータ構造の各第１の特定のビンに関するヌクレオチド配列を、割り当てられた１またはこれより多くの第１のヌクレオチド配列決定リードとともに決定する。上記ヌクレオチド配列を決定する工程は、上記第１の特定のビンに割り当てられた上記１またはこれより多くの第１のヌクレオチド配列決定リードのコンセンサス配列を決定する工程を包含し得る。上記コンセンサス配列を決定する工程は、上記第１の特定のビンに割り当てられた最も頻度の高い第１のヌクレオチド配列決定リードを、上記第１の特定のビンのコンセンサス配列として決定する工程を包含し得る。上記コンセンサス配列は、上記第１の特定のビンに割り当てられた上記第１のヌクレオチド配列決定リードの各相当する位置に関して最も頻度の高いヌクレオチド塩基を含み得る。上記コンセンサス配列を決定する工程は、上記第１の特定のビンに割り当てられた最高品質スコアを有する第１のヌクレオチド配列決定リードを、上記第１の特定のビンのコンセンサス配列として決定する工程を包含し得る。最高品質スコアは、上記最高品質スコアを有する第１のヌクレオチド配列決定リード上の各塩基の品質スコアに基づいて決定され得る。上記ヌクレオチド配列を決定する工程は、上記第１の特定のビンに割り当てられた１またはこれより多くの第１のヌクレオチド配列決定リードの配列を、上記第１の特定のビンの代表的配列として選択する工程を包含し得る。上記ヌクレオチド配列を決定する工程は、上記第１の特定のビンに割り当てられた１またはこれより多くの第１のヌクレオチド配列決定リードのうちの２つのアラインメントスコアが、アラインメントスコア閾値を上回ることを決定する工程を包含し得る。

【００６８】

ペアードエンド配列決定リード

各第１のヌクレオチド配列決定リードは、第２のヌクレオチド配列決定リードと関連し得る。上記第１のヌクレオチド配列決定リードおよび上記第２のヌクレオチド配列決定リードは、ペアードエンドヌクレオチド配列決定リードを形成し得る。上記コンピューティングシステムは、複数の第２の識別子サブシーケンスを、上記第２のヌクレオチド配列決定リードの第２の識別子配列から決定し；上記複数の第２の識別子サブシーケンスに対してハッシュ化を適用することによって、上記第２のヌクレオチド配列決定リードの第２のシグナチャーを決定し得る。

【００６９】

上記第１のヌクレオチド配列決定リードを割り当てる工程は、実行が異なれば異なり得る。例えば、上記第１のヌクレオチド配列決定リードを割り当てる工程は、上記第１のヌクレオチド配列決定リードおよび上記第２のヌクレオチド配列決定リードを含む配列決定リードのペアを、上記第１のシグナチャーに基づいて、上記第１のハッシュデータ構造の第１の特定のビンに割り当てる工程を包含し得る。別の例として、上記第１のヌクレオチド配列決定リードを割り当てる工程は、上記第２のヌクレオチド配列決定リードを、上記第２のシグナチャーに基づいて、上記第１のハッシュデータ構造の第２の特定のビンに割り当てる工程を包含する。さらに別の例として、上記第１のヌクレオチド配列決定リードを割り当てる工程は、上記第１のヌクレオチド配列決定リードおよび上記第２のヌクレオチド配列決定リードを含む配列決定リードのペアを、上記第２のシグナチャーに基づいて、第２のハッシュデータ構造の第２の特定のビンに割り当てる工程を包含する。

【００７０】

一例として、上記第１のヌクレオチド配列決定リードを割り当てる工程は、上記第１のヌクレオチド配列決定リードおよび上記ヌクレオチド配列決定リードを含む配列決定リードのペアを、上記第１のヌクレオチド配列決定リードの第１のシグナチャーの複数のサブシーケンスおよび上記第２のヌクレオチド配列決定リードの第２のシグナチャーの複数の

サブシーケンスに基づいて、それぞれ、上記第１のハッシュデータ構造の第１の特定のビンおよび第２のデータ構造の第２の特定のビンに割り当てる工程を包含する。上記コンピューティングシステムは、上記第１のハッシュデータ構造のビンのキーおよび上記第２のハッシュデータ構造のビンのキーを格納するために、それぞれ、第１のデータ構造および第２のデータ構造を格納し得る。配列決定リードの上記ペアを割り当てる工程は、上記第１のシグナチャーおよび上記第２のシグナチャーが、上記第１のデータ構造および上記第２のデータ構造の中に格納されることを決定する工程；ならびに配列決定リードの上記ペアを、上記第１のハッシュデータ構造の第１の特定のビンおよび上記第２のハッシュデータ構造の第２の特定のビンに、それぞれ、第１の格納したキーおよび上記第２の格納したキーを使用して割り当てる工程を包含し得る。配列決定リードの上記ペアおよび上記第１の格納したキーと関連する第１の配列決定リードおよび第２の格納したキーと関連する第２の配列決定リードを含むペアのアラインメントスコアは、アラインメントスコア閾値を上回る。

【００７１】

配列決定リードの上記ペアを割り当てる工程は、上記第１のデータ構造の中に格納されかつ上記第１のシグナチャーと関連する上記第１のハッシュデータ構造の１またはこれより多くの第１のキーを決定する工程；上記第２のデータ構造の中に格納されかつ上記第２のシグナチャーと関連する上記第２のハッシュデータ構造の１またはこれより多くの第２のキーを決定する工程；第１の格納したキーと関連する第１の配列決定リードおよび第２の格納したキーと関連する第２の配列決定リードを含むペアが、任意の第１の格納したキーと関連する第１の配列決定リードおよび任意の第２の格納したキーと関連する第２の配列決定リードを含む任意のペアの最高のアラインメントスコアを有することを、配列決定リードの上記ペアとともに決定する工程；ならびに配列決定リードの上記ペアを、上記第１のハッシュデータ構造の第１の特定のビンおよび上記第２のハッシュデータ構造の第２の特定のビンに、それぞれ、第１の配列決定リードおよび第２の配列決定リードのペアと関連する上記第１の格納したキーおよび上記第２の格納したキーを使用して、上記最高のアラインメントスコアとともに割り当てる工程を包含し得る。上記第１の格納したキーと関連する第１の配列決定リードは、上記第１のシグナチャーとともに任意の第１の格納したキーと関連する上記第１の配列決定リードの最高のアラインメントスコアを有し得る。上記第２の格納したキーと関連する第２の配列決定リードは、上記第２のシグナチャーとともに任意の第２の格納したキーと関連する第２の配列決定リードの最高のアラインメントスコアを有し得る。

【００７２】

<u>リード識別</u>

図８は、局所性鋭敏型ハッシュを使用して類似のリードを識別する、例示的方法８００を示すフロー図である。方法８００は、コンピューティングシステムのコンピューター可読媒体（例えば、１またはこれより多くのディスクドライブ）に格納された実行可能なプログラム命令のセットの中に埋め込まれ得る。例えば、図９に示されかつ以下で依り詳細に記載されるコンピューティングシステム９００は、方法８００を実行するために、実行可能なプログラム命令のセットを実行し得る。方法８００が開始される場合、上記実行可能なプログラム命令は、メモリ（例えば、ＲＡＭ）に搭載され得、コンピューティングシステム９００の１またはこれより多くのプロセッサによって実行され得る。方法８００が、図９に示されるコンピューティングシステム９００に関して記載されるが、その説明は例証に過ぎず、限定であることを意図しない。いくつかの実施形態において、方法８００またはその一部は、多数のコンピューティングシステムによって直列でまたは並列で行われ得る。上記コンピューティングシステムは、配列決定リードの複数のペアを格納するために、第１のハッシュデータ構造および第２のハッシュデータ構造を格納し得る。

【００７３】

配列決定リードの各ペアは、第１のヌクレオチド配列決定リードおよび第２のヌクレオチド配列決定リードを含み得、ここで配列決定リードの各ペアは、上記第１のヌクレオチ

ド配列決定リードの第１の識別子配列の第１の識別子サブシーケンスのハッシングによって生成したペアの第１のヌクレオチド配列決定リードの第１のシグナチャーに基づいて、上記第１のハッシュデータ構造の複数の第１のビンのうちの１つに割り当てられる。配列決定リードの各ペアは、上記第２のヌクレオチド配列決定リードの第２の識別子配列のハッシングによって生成したペアの第２のヌクレオチド配列決定リードの第２のシグナチャーに基づいて、上記第２のハッシュデータ構造の複数の第２のビンのうちの１つに割り当てられ得る。

【００７４】

方法８００がブロック８０４で開始された後、方法８００は、ブロック８０８へと進み、ここでコンピューティングシステムは、第１の問い合わせヌクレオチド配列決定リードおよび第２の問い合わせヌクレオチド配列決定リードのペアを受容する。方法８００は、ブロック８０８からブロック８１２へと進み、ここで上記コンピューティングシステムは、複数の第１の問い合わせ識別子サブシーケンスおよび複数の第２の問い合わせ識別子サブシーケンスを、それぞれ、上記第１の問い合わせヌクレオチド配列決定リードおよび上記第２の問い合わせヌクレオチド配列決定リードから生成する。上記問い合わせ識別子サブシーケンスをブロック８１２において生成した後、方法８００は、ブロック８１６へと進み、ここで上記コンピューティングシステムは、上記第１のヌクレオチド配列決定リードおよび上記第２のヌクレオチド配列決定リードに関する第１の問い合わせシグナチャーおよび第２の問い合わせシグナチャーを、それぞれ、上記複数の第１の問い合わせ識別子サブシーケンスおよび上記複数の第２の問い合わせ識別子サブシーケンスに対してハッシュ化を適用することによって生成する。上記コンピューティングシステムは、図７を参照して記載されるブロック７０８～７１６を参照して記載されるように、ブロック８０８～８１６において工程を実施し得る。

【００７５】

ブロック８１６の後に、上記方法は、直交する問い合わせを含み得る。例えば、上記方法は、ブロック８１６からブロック８２０へと進み、ここで上記コンピューティングシステムは、１またはこれより多くの第１の格納したペアおよび１またはこれより多くの第２の格納したペアを、上記第１のハッシュデータ構造および上記第２のハッシュデータ構造から、それぞれ、上記第１の問い合わせシグナチャーおよび上記第２の問い合わせシグナチャーを使用して回収する。ここで上記第１のペアおよび上記第２のペアの各々は、第１の格納したヌクレオチド配列決定リードおよび第２の格納したヌクレオチド配列決定リードを含む。ブロック８２０において配列決定リードのペアを回収した後、方法８００は、ブロック８２４へと進み、ここで上記コンピューティングシステムは、配列決定リード１および配列決定リード２としての上記第１の格納したペアおよび第２の格納したペアの両方に存在する第１の格納したヌクレオチド配列決定リードおよび第２の格納したヌクレオチド配列決定リードの各ペアを、上記問い合わせ配列決定リード１および上記問い合わせ配列決定リード２に類似であるとして決定する。方法８００は、ブロック８２８で終了する。

【００７６】

配列決定リードの各ペアは、第１の識別子配列および第２の識別子配列と関連し得る。上記コンピューティングシステムは、配列決定リードの第１のペアの上記第１の識別子配列および上記第２の識別子配列、ならびに配列決定リードの第２のペアの上記第２の識別子配列および第１の識別子配列が同一であることを決定し得；そして配列決定リードの上記第１のペアおよび配列決定リードの上記第２のペアのヌクレオチド配列を決定し得る。

【００７７】

一実施形態において、方法８００は、ブロック８０８において第１の問い合わせヌクレオチド配列決定リードを受容する工程を包含し得る。上記第１の問い合わせヌクレオチド配列決定リードを受容する工程は、上記第１の問い合わせヌクレオチド配列決定リードおよび上記第２の問い合わせヌクレオチド配列決定リードのペアを受容する工程を包含し得る。方法８００は、ブロック８１２において、複数の第１の問い合わせ識別子サブシーケ

ンスを、上記第１の問い合わせヌクレオチド配列決定リードから生成する工程を包含し得る。上記複数の第１の問い合わせ識別子サブシーケンスを生成する工程は、複数の第２の問い合わせ識別子サブシーケンスを上記第２のヌクレオチド配列決定リードから生成する工程を包含し得る。方法８００は、ブロック８１６において上記複数の第１の問い合わせ識別子サブシーケンスに対してハッシュ化を適用することによって、上記第１のヌクレオチド配列決定リードに関する第１の問い合わせシグナチャーを生成する工程を包含し得る。上記第１の問い合わせシグナチャーを生成する工程は、上記複数の第２の問い合わせ識別子サブシーケンスに対してハッシュ化を適用することによって、上記第２のヌクレオチド配列決定リードに関する第２の問い合わせシグナチャーを生成する工程を包含し得る。方法８００は、ブロック８２０において上記第１の問い合わせシグナチャーを使用して、１またはこれより多くの第１の格納したヌクレオチド配列決定リードを第１のハッシュデータ構造から回収する工程を包含し得る。上記第１の格納したヌクレオチド配列決定リードの各々は、上記第１の問い合わせヌクレオチド配列決定リードに類似であり得る。１またはこれより多くの第１の格納したヌクレオチド配列決定リードを回収する工程は、１またはこれより多くの第１の格納したペアを上記第１のハッシュデータ構造から回収する工程、配列決定リードの複数のペアを、上記第１の問い合わせシグナチャーおよび上記第２の問い合わせシグナチャーを使用して格納する工程を包含し得る。上記第１のペアの各々は、それぞれ、上記第１の問い合わせヌクレオチド配列決定リードおよび上記第２の問い合わせヌクレオチド配列決定リードに類似の第１の格納したヌクレオチド配列決定リードおよび第２の格納したヌクレオチド配列決定リードを含み得る。

【００７８】

実行環境

図９は、本明細書で開示される代謝産物、註釈および遺伝子組み込みシステムを実行するように構成された例示的なコンピューティングデバイス９００の一般的構造を示す。図９に示されるコンピューティングデバイス９００の一般的構造は、コンピューターハードウェアおよびソフトウェア構成要素の配置を含む。コンピューティングデバイス９００は、図９に示されるもの以外のより多くの（またはより少ない）要素を含み得る。しかし、これら概して従来どおりの要素の全てが、可能な開示を提供するために示されることは必要ではない。図示されるように、コンピューティングデバイス９００は、処理装置９４０、ネットワークインターフェイス９４５、コンピューター可読媒体ドライブ９５０、入力／出力デバイスインターフェイス９５５、ディスプレイ９６０、および入力デバイス９６５を含み、これらは全て、コミュニケーションバスによって互いと連絡し得る。ネットワークインターフェイス９４５は、１またはこれより多くのネットワークまたはコンピューティングシステムに対する接続を提供し得る。上記処理装置９４０は、このようにして、他のコンピューティングシステムまたはサービスから、ネットワークを介して情報および命令を受容し得る。処理装置９４０はまた、メモリ９７０に、およびメモリ９７０から連絡し得、入力／出力デバイスインターフェイス９５５を介して選択肢的なディスプレイ９６０に出力情報をさらに提供し得る。入力／出力デバイスインターフェイス９５５はまた、選択肢的な入力デバイス９６５（例えば、キーボード、マウス、デジタルペン、マイクロフォン、タッチスクリーン、ジェスチャー認識システム、音声認識システム、ゲームパッド、加速度計、ジャイロスコープ、または他の入力デバイス）からの入力を受け入れ得る。

【００７９】

メモリ９７０は、処理装置９４０が１またはこれより多くの実施形態を実行するために実行するコンピュータープログラム命令（いくつかの実施形態ではモジュールまたは構成要素としてグルーピングされる）を含み得る。メモリ９７０は一般に、ＲＡＭ、ＲＯＭおよび／または他の永続的な、補助的または一時的でないコンピューター可読媒体を含む。メモリ９７０は、コンピューティングデバイス９００の一般的管理およびオペレーションにおいて、処理装置９４０による使用のためのコンピュータープログラム命令を提供するオペレーティングシステム９７２を格納し得る。メモリ９７０は、コンピュータープログ

ラム命令および本開示の局面を実行するための他の情報をさらに含み得る。
【００８０】
例えば、一実施形態において、メモリ９７０は、局所性鋭敏型ハッシュ（例えば、図７を参照して記載されるリードコラプシング方法７００）を使用して配列決定リードをコラプシングするための局所性鋭敏型ハッシュベースのリードコラプシングモジュール９７４を含む。メモリ９７０は、さらにまたは代わりに、問い合わせ配列決定リードの類似のヌクレオチド配列決定リードを識別するための局所性鋭敏型ハッシュ問い合わせモジュール９７６を含み得る（例えば、図８を参照して記載される識別方法８００）。さらに、メモリ９７０は、データストア９９０、ならびに／またはリードコラプシングおよび／もしくは類似のヌクレオチド配列決定リード特定に関するデータおよびその結果を格納する１もしくはこれより多くの他のデータストアを含み得るかまたは連絡し得る。
【００８１】
ハードウェアアクセラレーション
いくつかの実施形態において、配列決定リードのグルーピングおよびコラプシングに関して開示される方法は、汎用コンピュータープロセッサより高効率で開示される方法を計算するようにデザインまたはプログラムされたアプリケーション特異的ハードウェアにおいて実行される。例えば、処理装置９４０は、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）またはアプリケーション特異的集積回路（ＡＳＩＣ）であり得る。
【００８２】
一例において、上記局所性鋭敏型ハッシュ（ＬＳＨ）オペレーションは、ＦＰＧＡによって加速され得る。いくつかの実施形態において、ＦＰＧＡによるＬＳＨオペレーションの加速は、ＵＭＩあたりにハッシュ表を構築および問い合わせするために必要とされるメモリに、およびソフトウェアにおいてボトルネックとなるメモリハンド幅にどの程度近いかにも依存し得る。配列決定リードと関連するＵＭＩのクラスター化が、ＦＰＧＡにおいて実行される場合、それは、同じハードウェア内で上記リードコラプシング方法を加速することはまた、有益であり得る。
【００８３】
いくつかの実施形態において、１またはこれより多くのアプリケーション特異的集積回路（ＡＳＩＣ）は、本明細書で記載されるそれぞれのゲノム分析モジュール、または他のコンピューターのうちの１またはこれより多くの機能を実施するようにプログラムされ得る。ＡＳＩＣは、上記ＡＳＩＣのデジタルロジックゲートが、ＶＨＤＬのようなハードウェア記述言語を使用してプログラム可能であるという点で、本明細書で記載されるＦＰＧＡに類似の１またはこれより多くのプログラマブルロジック回路を含む集積回路を含む。しかし、ＡＳＩＣは、ＡＳＩＣが、１回だけプログラム可能であり、一旦プログラムされたら動的に再構成され得ないという点で、ＦＰＧＡとは異なる。さらに、本開示の局面は、ＦＰＧＡまたはＡＳＩＣを使用して配列決定リードのグルーピングおよびコラプシングを実行することに限定されない。代わりに、処理装置９４０のゲノム分析モジュール、または他のコンピューターのいずれも、ソフトウェア命令の実行を通じて配列決定リードのグルーピングおよびコラプシングを実行する、１またはこれより多くの中央演算装置（ＣＰＵ）、グラフィック処理装置（ＧＰＵ）、またはそのための任意の組み合わせを使用して実行され得る。
【００８４】
いくつかの実行において、配列決定リードのグルーピングおよびコラプシングを実行する、ＦＰＧＡ、ＡＳＩＣ、ＣＰＵ、ＧＰＵ、またはこれらの組み合わせのような集積回路の使用は、単一のＦＰＧＡ、単一のＡＳＩＣ、単一のＣＰＵ、単一のＧＰＵ、またはこれらの任意の組み合わせを含み得る。代わりに、またはさらに、配列決定リードのグルーピングおよびコラプシングを実行する、ＦＰＧＡ、ＡＳＩＣ、ＣＰＵ、ＧＰＵ、またはこれらの組み合わせのような集積回路の使用は、多数のＦＰＧＡ、多数のＡＳＩＣ、多数のＣＰＵ、または多数のＧＰＵ、またはこれらの任意の組み合わせを含み得る。配列決定リードのグルーピングおよびコラプシングを実行する、多数のＦＰＧＡのようなさらなる集積

回路の使用は、マッピング、整列、Ｐ－ＨＭＭ確率計算、およびバリアント呼び出しのような二次分析オペレーションを行うためにかかる時間量を低減し得る。いくつかの実行において、これらの二次分析オペレーションを実行するＦＰＧＡの使用は、これらの二次分析オペレーションを完了するのにかかる時間を、２４時間（またはこれより長い）から３０分（もしくはより短い）程度の短さへと低減し得る。いくつかの実行において、これらの二次分析オペレーションを実行する多数のＦＰＧＡの使用は、これらの二次分析オペレーションの５分程度の短さでの完了を生じ得る。

【００８５】

用語法

本明細書中の実質的に任意の複数形および／または単数形の用語の使用に関して、当業者は、その状況および／または適用に適切な場合、その複数形から単数形へおよび／またはその単数形から複数形へと解釈し得る。その種々の単数系／複数形の交換は、明瞭にする目的で、本明細書で明らかに述べられ得る。

【００８６】

一般に、本明細書で、および特に添付の請求項において使用される用語（例えば、添付の請求項の本文）は、概して、「開放系」の用語（例えば、用語「含む、包含する（ｉｎｃｌｕｄｉｎｇ）」は、「が挙げられるが、これらに限定されない（ｉｎｃｌｕｄｉｎｇ ｂｕｔ ｎｏｔ ｌｉｍｉｔｅｄ ｔｏ）」と解釈されるべきであり、用語「有する（ｈａｖｉｎｇ）」は、「を少なくとも有する（ｈａｖｉｎｇ ａｔ ｌｅａｓｔ）」と解釈されるべきであり、用語「含む、包含する（ｉｎｃｌｕｄｅｓ）」は、「が挙げられるが、これらに限定されない（ｉｎｃｌｕｄｅｓ ｂｕｔ ｉｓ ｎｏｔ ｌｉｍｉｔｅｄ ｔｏ）」と解釈されるべきである、など）と意図されることは、当業者によって理解される。導入される請求項の記載の特定の数字が意図される場合、このような意図は、その請求項の中で明示的に記載され、このような記載がない場合、このような意図は存在しないことは、当業者によってさらに理解される。例えば、理解の補助として、以下の添付の請求項は、請求項の記載を導入するために、導入句「少なくとも１（ａｔ ｌｅａｓｔ ｏｎｅ）」および「１またはこれより多くの（ｏｎｅ ｏｒ ｍｏｒｅ）」の使用を含み得る。しかし、このような語句の使用は、不定冠詞「１つの、ある（ａ）」または「１つの、ある（ａｎ）」による請求項の記載の導入が、その同じ請求項が、その導入句「１またはこれより多くの」または「少なくとも１」および不定冠詞（例えば、「１つの、ある（ａ）」または「１つの、ある（ａｎ）」を含む場合であっても、このような導入された請求項の記載を含む任意の特定の請求項を、１つのこのような記載のみを含む実施形態に限定することを意味するとは解釈されるべきではない（例えば、「１つの、ある（ａ）」および／または「１つの、ある（ａｎ）」は、「少なくとも１」または「１またはこれより多くの」を意味すると解釈されるべきである）；同じことが、請求項の記載を導入するために使用される定冠詞の使用に関してもあてはまる。さらに、導入された請求項の記載の特定の数字が、明示的に記載されているとしても、当業者は、このような記載が少なくともその記載された数字を意味すると解釈されるべきであることを認識する（例えば、「２つの記載（ｔｗｏ ｒｅｃｉｔａｔｉｏｎｓ）」という最低限の記載は、他の修飾語がなくても、少なくとも２つの記載、または２またはこれより多くの記載を意味する）。さらに、「Ａ、Ｂ、およびＣなどのうちの少なくとも１つ（ａｔ ｌｅａｓｔ ｏｎｅ ｏｆ Ａ，Ｂ，ａｎｄ Ｃ，ｅｔｃ．）」に類似の慣習が使用される場合では、一般に、このような構文は、当業者がその慣習を理解するという意味において意図される（例えば、「Ａ、Ｂ、およびＣのうちの少なくとも１つを有するシステム（ａ ｓｙｓｔｅｍ ｈａｖｉｎｇ ａｔ ｌｅａｓｔ ｏｎｅ ｏｆ Ａ，Ｂ，ａｎｄ Ｃ）」としては、Ａのみ、Ｂのみ、Ｃのみ、ＡとＢとを一緒に、ＡとＣとを一緒に、ＢとＣとを一緒に、および／またはＡとＢとＣとを一緒に、などを有するシステムが挙げられるが、これらに限定されない）。「Ａ、Ｂ、またはＣなどのうちの少なくとも１つ（ａｔ ｌｅａｓｔ ｏｎｅ ｏｆ Ａ，Ｂ，ｏｒ Ｃ，ｅｔｃ．）」に類似の慣習が使用される場合では、一般に、このような構文は、当業者がその慣習を理解するという意味において意図される（例えば、「Ａ、Ｂ、または

Ｃのうちの少なくとも１つを有するシステム（ａ　ｓｙｓｔｅｍ　ｈａｖｉｎｇ　ａｔ　ｌｅａｓｔ　ｏｎｅ　ｏｆ　Ａ，　Ｂ，　ｏｒ　Ｃ）」としては、Ａのみ、Ｂのみ、Ｃのみ、ＡとＢとを一緒に、ＡとＣとを一緒に、ＢとＣとを一緒に、および／またはＡとＢとＣとを一緒に、などを有するシステムが挙げられるが、これらに限定されない）。２またはこれより多くの代替の用語を表す実質的に任意の離節的な単語および／または語句が、説明の中であろうとも、請求項の中であろうとも、図面の中であろうとも、その用語のうちの１つ、その用語のうちのいずれか、または両方の用語を含むという可能性を企図することが理解されるべきであることは、当業者によってさらに理解される。例えば、語句「ＡまたはＢ（Ａ　ｏｒ　Ｂ）」は、「Ａ」もしくは「Ｂ」または「ＡおよびＢ（Ａ　ａｎｄ　Ｂ）」という可能性を含むことが理解される。

【００８７】

さらに、本開示の特徴または局面がマーカッシュグループの用語で記載される場合、当業者は、本開示がまた、それによって、そのマーカッシュグループの任意の個々のメンバーまたはメンバーの下位群に関して記載されていることを認識する。

【００８８】

当業者によって理解されるように、任意のおよび全ての目的に関して（例えば、書面による説明を提供することに関して）、本明細書で開示される全ての範囲はまた、任意のおよび全ての考えられる部分範囲、ならびにその部分範囲の組み合わせを包含する。任意の列挙された範囲は、十分に記載されかつその同じ範囲が少なくとも等しく１／２、１／３、１／４、１／５、１／１０などへと分離されることを可能にすると容易に認識され得る。非限定的な例として、本明細書で考察される各範囲は、下位の１／３、中央の１／３、および上位の１／３などへと容易に分離され得る。同様に当業者によって理解されるように、「まで（ｕｐ　ｔｏ）」、「少なくとも（ａｔ　ｌｅａｓｔ）」、「より大きい、を超える（ｇｒｅａｔｅｒ　ｔｈａｎ）」、「より小さい、未満（ｌｅｓｓ　ｔｈａｎ）」などのような全ての文言は、記載される数字を包含し、上記で考察されるとおりの部分範囲へとその後に分離され得る範囲に言及する。最後に、当業者によって理解されるように、範囲は、各個々のメンバーを包含する。従って、例えば、１～３個の物品を有する群は、１個の、２個の、または３個の物品を有する群に言及する。同様に、１～５個の物品を有する群は、１個の、２個の、３個の、４個の、または５個の物品を有する群に言及するなど。

【００８９】

種々の局面および実施形態が本明細書で開示されているが、他の局面および実施形態は、当業者に明らかである。本明細書で開示される種々の局面および実施形態は、例証目的であり、限定であるとは意図されず、真の範囲および趣旨は、以下の請求項によって示される。

本発明は、例えば、以下の項目を提供する。

(項目１)

ヌクレオチド配列決定リードからヌクレオチド配列を決定するためのシステムであって、前記システムは、

実行可能な命令および複数のビンの中にヌクレオチド配列決定リードを格納するための第１のハッシュデータ構造を格納するように構成された一時的でないメモリ；ならびに

以下を包含する方法を行うために、前記実行可能な命令によってプログラムされたハードウェアプロセッサ：

複数の第１のヌクレオチド配列決定リードを受容する工程；

各第１のヌクレオチド配列決定リードに関して：

複数の第１の識別子サブシーケンスを、前記第１のヌクレオチド配列決定リードの第１の識別子配列から生成する工程；

前記第１のヌクレオチド配列決定リードに関する第１のシグナチャーを、前記複数の第１の識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；および

前記第１のヌクレオチド配列決定リードを、前記第１のシグナチャーに基づいて、

前記第１のハッシュデータ構造のうちの少なくとも１つの第１の特定のビンに割り当てる工程；ならびに
　前記第１のハッシュデータ構造の各第１の特定のビンに関するヌクレオチド配列を、割り当てられた１またはこれより多くの第１のヌクレオチド配列決定リードとともに決定する工程、
を含むシステム。
(項目２)
前記第１のヌクレオチド配列決定リードを割り当てる工程は、
　前記第１のシグナチャーの複数のサブシーケンスを、前記第１のヌクレオチド配列決定リードの前記第１のシグナチャーから決定すること；および
　前記第１のヌクレオチド配列決定リードを、前記第１のシグナチャーのサブシーケンスに基づいて、複数の第１のハッシュデータ構造の各第１のハッシュデータ構造の第１の特定のビンに割り当てること、
を包含する、項目１に記載のシステム。
(項目３)
前記第１のヌクレオチド配列決定リードを割り当てる工程は、
　前記第１のシグナチャーの複数のサブシーケンスを、前記第１のヌクレオチド配列決定リードの前記第１のシグナチャーから決定すること；および
　前記第１のヌクレオチド配列決定リードを、前記第１のシグナチャーの前記複数のサブシーケンスに基づいて、前記第１のハッシュデータ構造の複数の第１の特定のビンに割り当てること、
を包含する、項目１に記載のシステム。
(項目４)
前記第１の特定のビンは、前記第１のハッシュデータ構造に存在するビンであり、前記第１のヌクレオチド配列決定リードおよび前記第１のハッシュデータ構造の前記第１の特定のビンに割り当てられた別の第１のヌクレオチド配列決定リードのアラインメントスコアは、アラインメントスコア閾値を上回る、項目１に記載のシステム。
(項目５)
前記第１の特定のビンは、前記第１のハッシュデータ構造に存在するビンであり、前記第１のヌクレオチド配列決定リードおよび前記第１のハッシュデータ構造の前記第１の特定のビンに割り当てられた任意の第１のヌクレオチド配列決定リードの最高のアラインメントスコアは、アラインメントスコア閾値を上回る、項目１に記載のシステム。
(項目６)
前記第１の特定のビンは、前記第１のハッシュデータ構造の新しいビンであり、前記第１のヌクレオチド配列決定リードおよび前記第１のハッシュデータ構造の任意に存在するビンに割り当てられた任意の第１のヌクレオチド配列決定リードのアラインメントスコアは、アラインメントスコア閾値を下回る、項目１に記載のシステム。
(項目７)
前記第１のシグナチャーは、前記第１のハッシュデータ構造の前記第１の特定のビンのキーにマッチする、項目１に記載のシステム。
(項目８)
前記第１のシグナチャーおよび前記第１のハッシュデータ構造の前記第１の特定のビンの前記キーは、同一である、項目１に記載のシステム。
(項目９)
各第１のヌクレオチド配列決定リードは、第２のヌクレオチド配列決定リードと関連し、前記第１のヌクレオチド配列決定リードおよび前記第２のヌクレオチド配列決定リードは、ペアードエンドヌクレオチド配列決定リードを形成する、項目１に記載のシステム。
(項目１０)
前記ヌクレオチド配列を決定する工程は、前記第１の特定のビンに割り当てられた前記１またはこれより多くの第１のヌクレオチド配列決定リードのコンセンサス配列を決定する

ことを包含する、項目１に記載のシステム。
(項目１１)
前記コンセンサス配列を決定することは、前記第１の特定のビンに割り当てられた最高の品質スコアを有する第１のヌクレオチド配列決定リードを、前記第１の特定のビンの前記コンセンサス配列として決定することを包含する、項目１０に記載のシステム。
(項目１２)
前記ヌクレオチド配列を決定する工程は、前記第１の特定のビンに割り当てられた前記１またはこれより多くの第１のヌクレオチド配列決定リードの配列を、前記第１の特定のビンの代表的配列として選択することを包含する、項目１に記載のシステム。
(項目１３)
前記ヌクレオチド配列を決定する工程は、前記第１の特定のビンに割り当てられた前記１またはこれより多くの第１のヌクレオチド配列決定リードのうちの２つのアラインメントスコアがアラインメントスコア閾値を上回ることを決定することを包含する、項目１に記載のシステム。
(項目１４)
前記複数のヌクレオチド配列決定リードは、同一の物理的識別子配列と関連する、項目１に記載のシステム。
(項目１５)
前記複数のヌクレオチド配列決定リードは、いかなる物理的識別子配列とも関連しない、項目１に記載のシステム。
(項目１６)
ヌクレオチド配列決定リードからヌクレオチド配列を決定するためのコンピューターで実行される方法であって、前記方法は、
　複数の第１のヌクレオチド配列決定リードを受容する工程；
　各第１のヌクレオチド配列決定リードに関して：
　　複数の第１の識別子サブシーケンスを、前記第１のヌクレオチド配列決定リードの第１の識別子配列から生成する工程；
　　前記第１のヌクレオチド配列決定リードに関する第１のシグナチャーを、前記複数の第１の識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；および
　　前記第１のヌクレオチド配列決定リードを、前記第１のシグナチャーに基づいて、第１のデータ構造の第１の特定のビンに割り当てる工程；および
　　前記第１のデータ構造の各第１の特定のビンに関するヌクレオチド配列を、割り当てられた１またはこれより多くの第１のヌクレオチド配列決定リードとともに決定する工程、
を包含する方法。
(項目１７)
前記複数の第１の識別子サブシーケンスを生成する工程は、前記配列決定リードの前記第１の識別子配列から複数のｋマーを生成することを包含する、項目１６に記載の方法。
(項目１８)
前記サブシーケンスは、ヌクレオチド挿入、ヌクレオチド欠失、ヌクレオチド置換、またはこれらの組み合わせを含む、項目１７に記載の方法。
(項目１９)
２つの連続する第１の識別子サブシーケンスが重なり合う、項目１７に記載の方法。
(項目２０)
前記複数の第１の識別子サブシーケンスは、複数の４マーを含み、前記第１の識別子配列は、約２５ヌクレオチドを含む、項目１７に記載の方法。
(項目２１)
前記第１の識別子配列は、配列決定リード１のサブシーケンスである、項目１７に記載の方法。

(項目２２)
前記第１のシグナチャーを生成することは、各第１の識別子サブシーケンスに関する複数のハッシュを決定することを包含する、項目１７に記載の方法。
(項目２３)
前記第１のデータ構造は、ハッシュ表を含む、項目１７に記載の方法。
(項目２４)
類似のヌクレオチド配列決定リードを識別するためのシステムであって、前記システムは、
以下を格納するように構成された一時的でないメモリ：
実行可能な命令、
配列決定リードの複数のペアを格納するための第１のハッシュデータ構造および第２のハッシュデータ構造；ならびに
以下を包含する方法を行うために、前記実行可能な命令によってプログラムされたハードウェアプロセッサ：
第１の問い合わせヌクレオチド配列決定リードおよび第２の問い合わせヌクレオチド配列決定リードのペアを受容する工程；
複数の第１の問い合わせ識別子サブシーケンスおよび複数の第２の問い合わせ識別子サブシーケンスを、それぞれ、前記第１の問い合わせヌクレオチド配列決定リードおよび前記第２の問い合わせヌクレオチド配列決定リードから生成する工程；
前記第１のヌクレオチド配列決定リードおよび前記第２のヌクレオチド配列決定リードに関する第１の問い合わせシグナチャーおよび第２の問い合わせシグナチャーを、それぞれ、前記複数の第１の問い合わせ識別子サブシーケンスおよび前記複数の第２の問い合わせ識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；
１またはこれより多くの第１の格納したペアおよび１またはこれより多くの第２の格納したペアを、それぞれ、前記第１のハッシュデータ構造および前記第２のハッシュデータ構造から、前記第１の問い合わせシグナチャーおよび前記第２の問い合わせシグナチャーを使用して回収する工程であって、ここで前記第１のペアおよび前記第２のペアの各々は、第１の格納したヌクレオチド配列決定リードおよび第２の格納したヌクレオチド配列決定リードを含む工程；ならびに
前記第１の格納したペアおよび第２の格納したペアの両方に存在する第１の格納したヌクレオチド配列決定リードおよび第２の格納したヌクレオチド配列決定リードの各ペアを、それぞれ、前記問い合わせ配列決定リード１および前記問い合わせ配列決定リード２に類似の配列決定リード１および配列決定リード２として決定する工程、
を含む、システム。
(項目２５)
配列決定リードの各ペアは、第１のヌクレオチド配列決定リードおよび第２のヌクレオチド配列決定リードを含み、配列決定リードの各ペアは、前記第１のヌクレオチド配列決定リードの第１の識別子配列の第１の識別子サブシーケンスのハッシングによって生成した前記ペアの第１のヌクレオチド配列決定リードの第１のシグナチャーに基づいて、前記第１のハッシュデータ構造の複数の第１のビンのうちの１つに割り当てられ、配列決定リードの各ペアは、前記第２のヌクレオチド配列決定リードの第２の識別子配列のハッシングによって生成した前記ペアの第２のヌクレオチド配列決定リードの第２のシグナチャーに基づいて、前記第２のハッシュデータ構造の複数の第２のビンのうちの１つに割り当てられる、項目２４に記載のシステム。
(項目２６)
前記ハードウェアプロセッサは、
配列決定リードの各ペアに関して：
複数の第１の識別子サブシーケンスを、配列決定リードの前記ペアのうちの前記第１のヌクレオチド配列決定リードの第１の識別子配列から生成する工程；
前記第１のヌクレオチド配列決定リードに関する第１のシグナチャーを、前記複数の第１の識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；

および
　　配列決定リードの前記ペアを、前記第１のシグナチャーに基づいて、前記第１のハッシュデータ構造のうちの少なくとも１つの第１の特定のビンに割り当てる工程；および
　　前記第１のハッシュデータ構造の各第１の特定のビンに関するヌクレオチド配列を、前記１またはこれより多くのペアのうちの前記第１のヌクレオチド配列決定リードおよび前記第２のヌクレオチド配列決定リードから割り当てられた第１のヌクレオチド配列決定リードおよび第２のヌクレオチド配列決定リードの１またはこれより多くのペアとともに決定する工程、
を包含する方法を行うために前記実行可能な命令によってプログラムされている、項目２５に記載のシステム。
(項目２７)
配列決定リードの各ペアは、第１の識別子配列および第２の識別子配列と関連し、前記ハードウェアプロセッサは、
　　配列決定リードの第１のペアの前記第１の識別子配列および前記第２の識別子配列、ならびに配列決定リードの第２のペアの前記第２の識別子配列および第１の識別子配列が同一であることを決定する工程；ならびに
　　配列決定リードの前記第１のペアおよび配列決定リードの前記第２のペアのヌクレオチド配列を決定する工程、
を包含する方法を行うために前記実行可能な命令によってプログラムされている、項目２５に記載のシステム。
(項目２８)
類似のヌクレオチド配列決定リードを識別するための方法であって、前記方法は、
　　第１の問い合わせヌクレオチド配列決定リードを受容する工程；
　　複数の第１の問い合わせ識別子サブシーケンスを、前記第１の問い合わせヌクレオチド配列決定リードから生成する工程；
　　前記第１のヌクレオチド配列決定リードに関する第１の問い合わせシグナチャーを、前記複数の第１の問い合わせ識別子サブシーケンスに対してハッシュ化を適用することによって生成する工程；および
　　１またはこれより多くの第１の格納したヌクレオチド配列決定リードを、第１のハッシュデータ構造から、前記第１の問い合わせシグナチャーを使用して回収する工程であって、ここで前記第１の格納したヌクレオチド配列決定リードの各々は、前記第１の問い合わせヌクレオチド配列決定リードに類似である工程、
を包含する方法。
(項目２９)
前記第１の問い合わせヌクレオチド配列決定リードを回収する工程は、前記第１の問い合わせヌクレオチド配列決定リードおよび第２の問い合わせヌクレオチド配列決定リードのペアを受容することを包含し、前記複数の第１の問い合わせ識別子サブシーケンスを生成する工程は、複数の第２の問い合わせ識別子サブシーケンスを前記第２のヌクレオチド配列決定リードから生成することを包含し、前記第１の問い合わせシグナチャーを生成する工程は、前記第２のヌクレオチド配列決定リードに関する第２の問い合わせシグナチャーを、前記複数の第２の問い合わせ識別子サブシーケンスに対してハッシュ化を適用することによって生成することを包含し、そして１またはこれより多くの第１の格納したヌクレオチド配列決定リードを回収する工程は、１またはこれより多くの第１の格納したペアを、前記第１のハッシュデータ構造から回収すること、配列決定リードの複数のペアを、前記第１の問い合わせシグナチャーおよび前記第２の問い合わせシグナチャーを使用して格納することを包含し、前記第１のペアの各々は、前記第１の問い合わせヌクレオチド配列決定リードおよび前記第２の問い合わせヌクレオチド配列決定リードにそれぞれ類似の第１の格納したヌクレオチド配列決定リードおよび第２の格納したヌクレオチド配列決定リードを含む、項目２８に記載の方法。

【図面】

【図１】

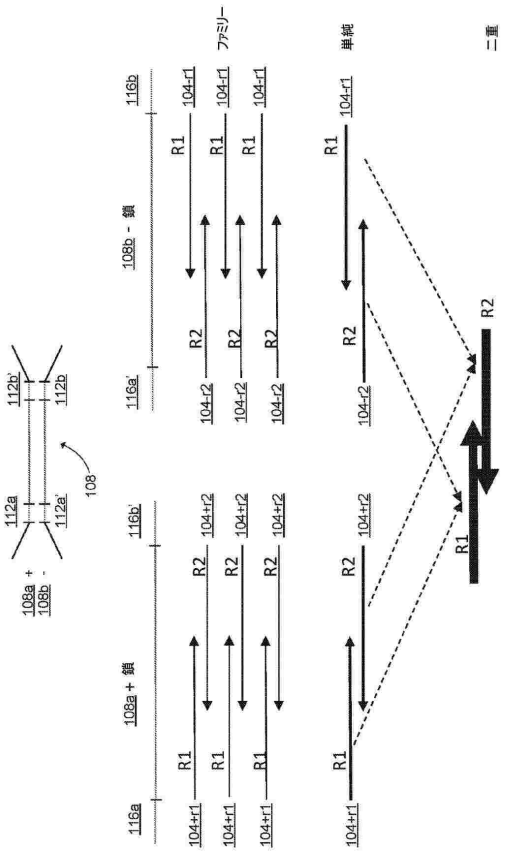

FIG. 1

【図２－１】

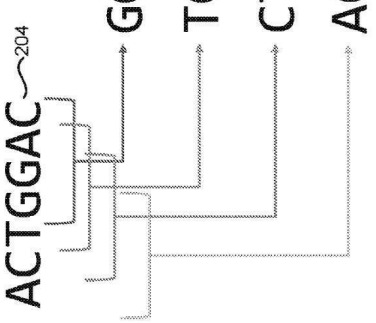

FIG. 2A

【図２－２】

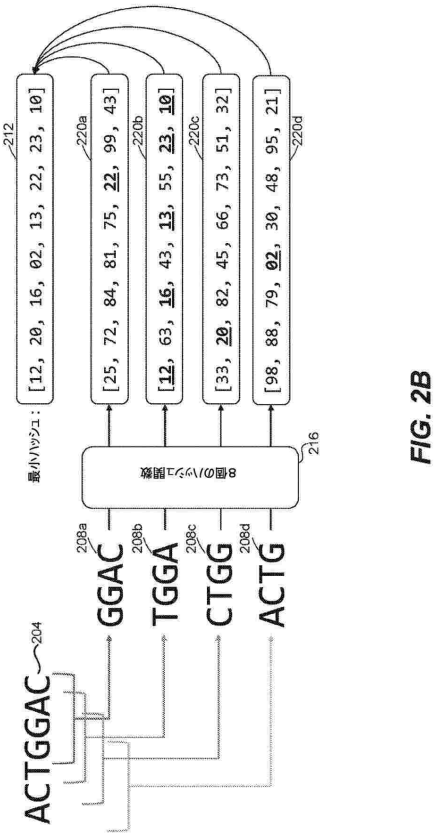

FIG. 2B

【図２－３】

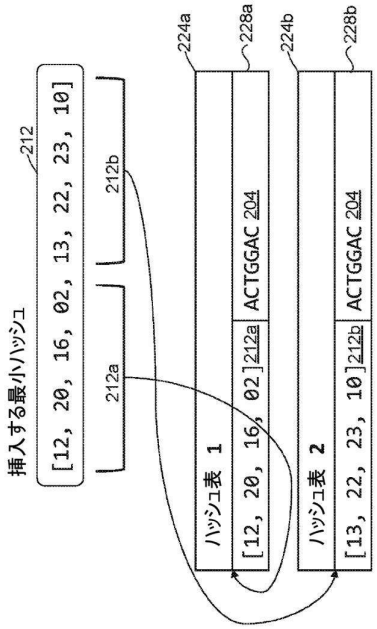

FIG. 2C

【図２－４】

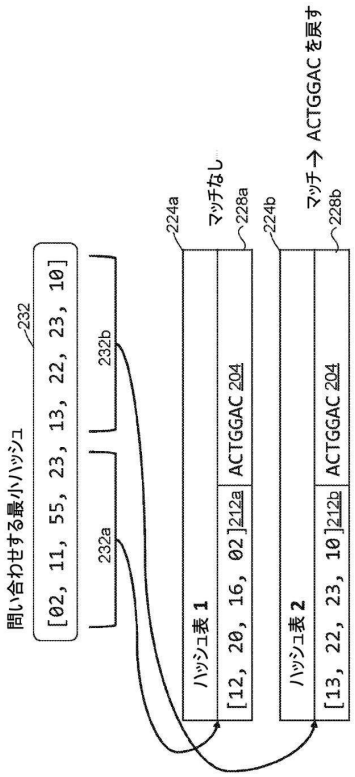

FIG. 2D

【図３】

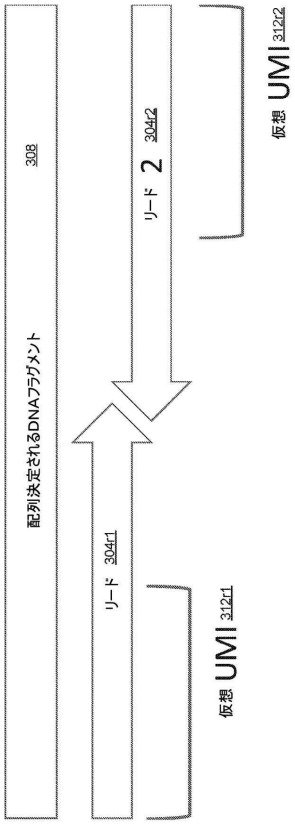

FIG. 3

【図４】

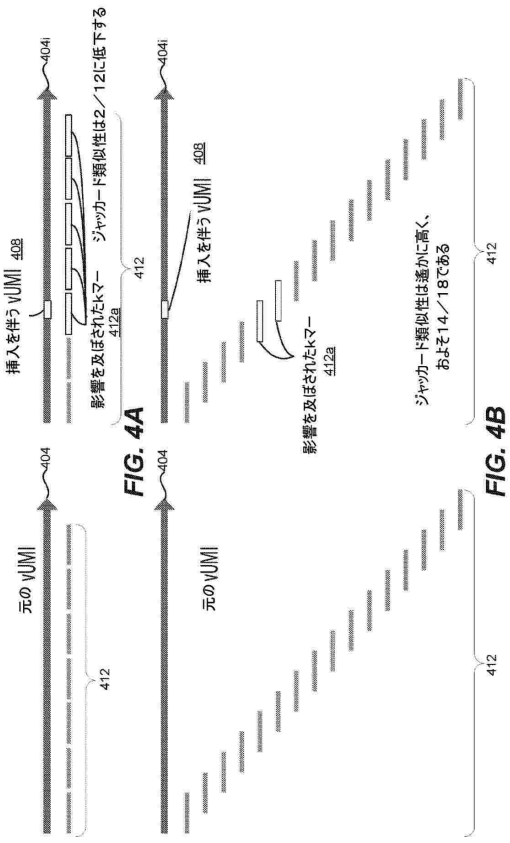

FIG. 4A

FIG. 4B

【図５】

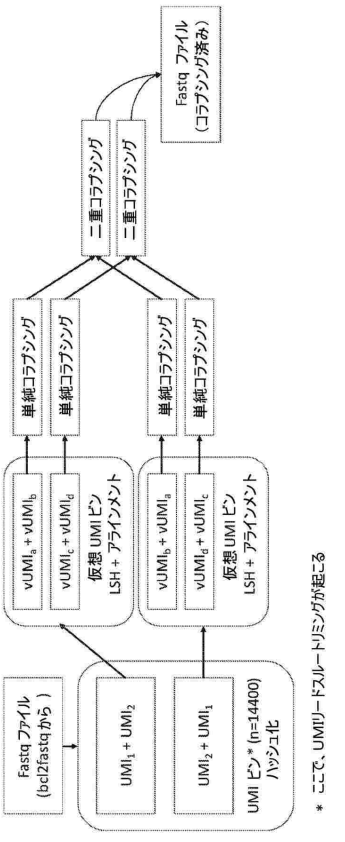

FIG. 5

【図６－１】

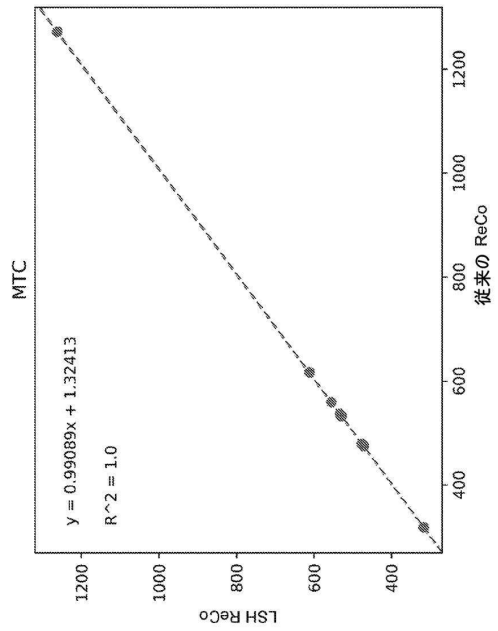

FIG. 6A

【図６－２】

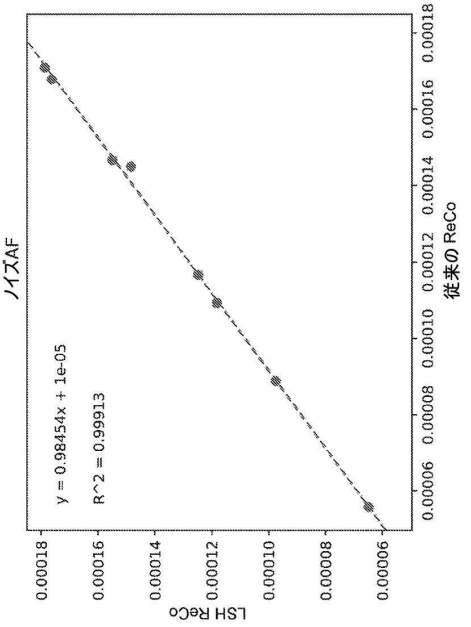

FIG. 6B

【図６－３】

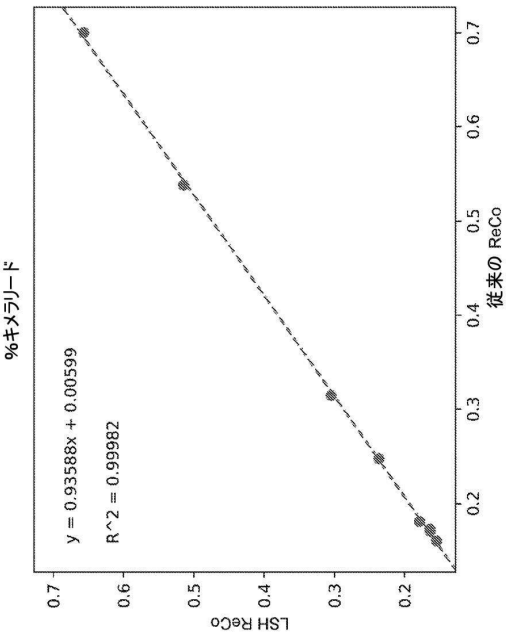

FIG. 6C

【図６－４】

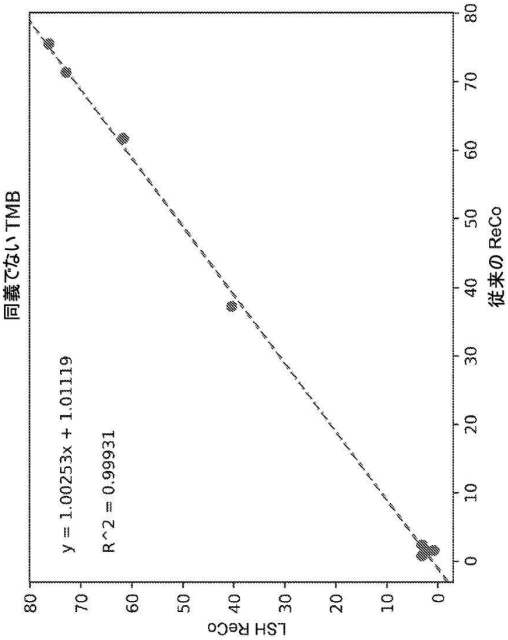

FIG. 6D

【図６－５】

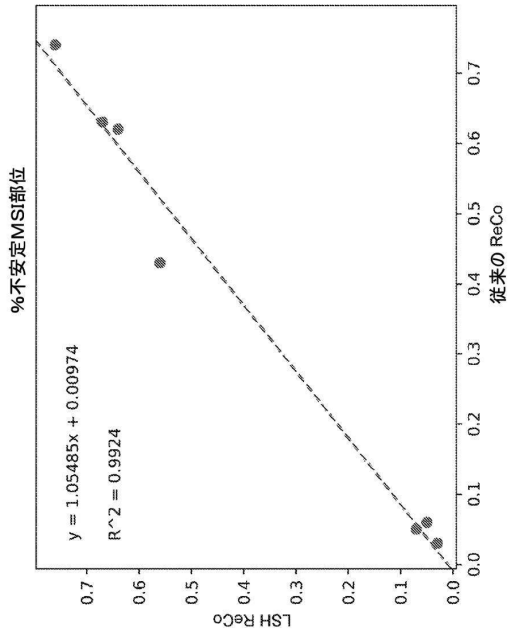

FIG. 6E

【図６－６】

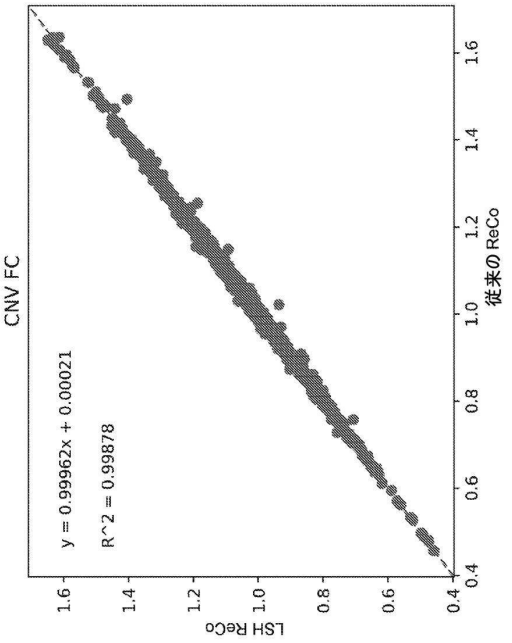

FIG. 6F

【図７】

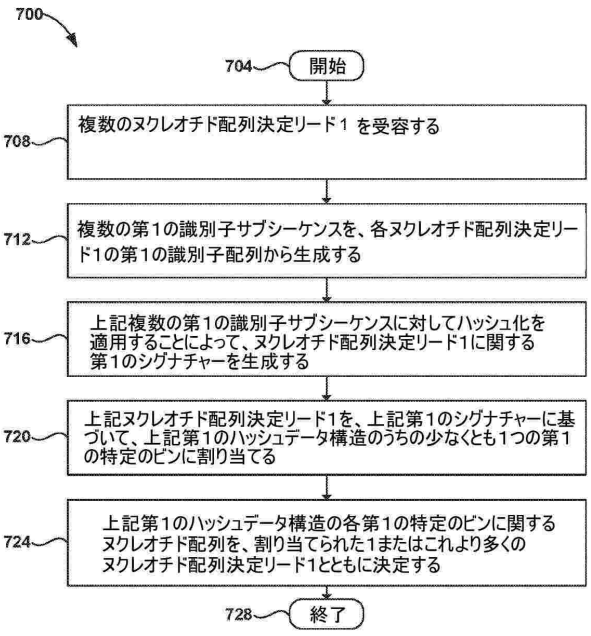

FIG. 7

【図８】

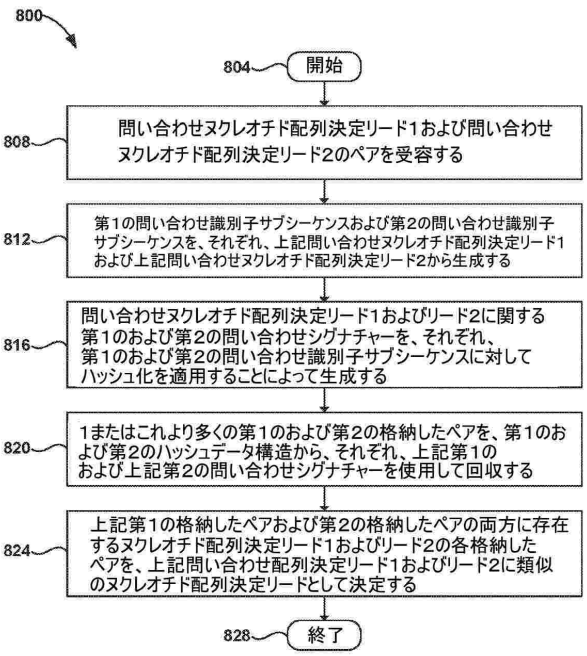

FIG. 8

【図９】

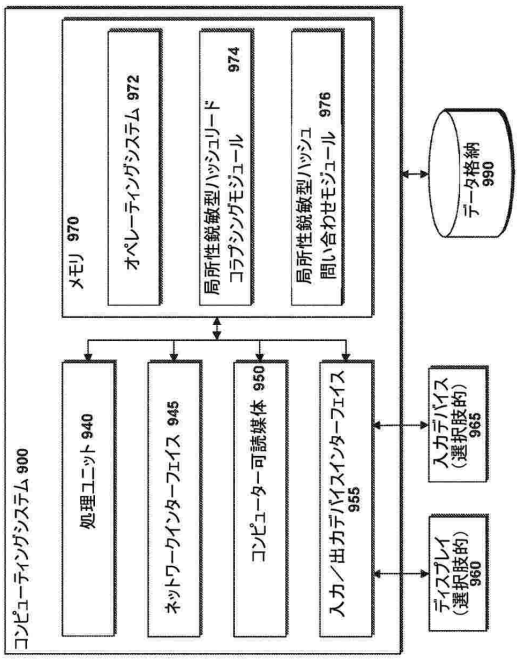
コンピューティングシステム 900
処理ユニット 940
ネットワークインターフェイス 945
コンピューター可読媒体 950
入力／出力デバイスインターフェイス 955
メモリ 970
オペレーティングシステム 972
局所性鋭敏型ハッシュリード コラプシングモジュール 974
局所性鋭敏型ハッシュ 問い合わせモジュール 976
データ格納 990
入力デバイス (選択肢的) 965
ディスプレイ (選択肢的) 960

FIG. 9

フロントページの続き

テッド 気付
(72)発明者 ウー， ケビン エリック
アメリカ合衆国 カリフォルニア ９２１２２， サンディエゴ， イルミナ ウェイ ５２００， イルミナ， インコーポレイテッド 気付
(72)発明者 ビルケ， スヴェン
アメリカ合衆国 カリフォルニア ９２１２２， サンディエゴ， イルミナ ウェイ ５２００， イルミナ， インコーポレイテッド 気付

審査官 山崎 誠也

(56)参考文献 Assemblinglarge genomes with single-molecule sequencing and locality-sensitive hashing，[online]，VOLUME 33 NUMBER 6，2015年06月，[2023年12月26日検索], インターネット〈URL:https://www.nature.com/articles/nbt.3238.pdf〉
Neil A. Miller，A 26-hour system of highly sensitive whole genome sequencing for emergency management of genetic diseases，[online]，2015年，p.1-16，[2023年12月26日検索], インターネット〈URL:https://genomemedicine.biomedcentral.com/articles/10.1186/s13073-015-0221-8〉

(58)調査した分野 (Int.Cl.，ＤＢ名)
Ｇ１６Ｂ ５／００－９９／００
Ｇ０６Ｆ １６／９０１