(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 114783523 A
(43)申请公布日 2022.07.22

(21)申请号 202210452629.4

(22)申请日 2016.10.21

(30)优先权数据
62/244,541 2015.10.21 US

(62)分案原申请数据
201680061446.2 2016.10.21

(71)申请人 相干逻辑公司
地址 美国得克萨斯

(72)发明人 M·B·多尔 J·D·加玛尼
S·V·伍德 D·G·阿拉斯塔斯
M·A·亨特

(74)专利代理机构 中国贸促会专利商标事务所有限公司 11038
专利代理师 鲍进

(51)Int.Cl.
G16B 40/00(2019.01)
G16B 30/10(2019.01)
G16B 20/20(2019.01)

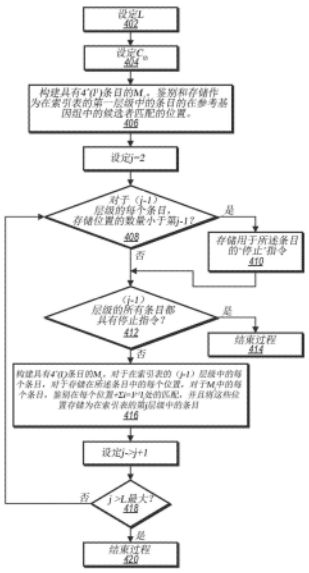
权利要求书2页 说明书20页 附图9页

(54)发明名称

使用分级反向索引表的DNA比对

(57)摘要

本发明公开涉及使用分级反向索引表的DNA比对。用于构建可用于将检索序列与参考数据匹配的分级索引表的系统和方法。所述索引表可经构建以含有与给定长度的所有子序列的穷尽性列表相关联的条目，其中每个条目含有在所述参考数据中的每个子序列的匹配的数量和位置。可以迭代方式构建所述分级索引表，其中基于匹配的数量大于一组相应阈值中的每一个，选择性地和迭代地构建用于每个延长子序列的条目。所述分级索引表可用于搜索在检索序列和参考数据之间的匹配，并且对每个相应候选匹配执行错配鉴别和表征。



CN 114783523 A

1.一种用于将检索序列与参考数据匹配的方法,所述方法包含:

通过计算装置执行:

将与参考数据相关联的分级索引表存储在非易失性存储器中;

接收指定检索序列的输入;以及

在分级索引表中检索检索序列与参考数据的一个或多个匹配,其中所述在分级索引表中检索包括:

匹配具有第一长度的检索序列的第一子序列与分级索引表的第一层级中的第一条目;

从所述第一条目中读取信息,其中所述信息指示所述分级索引表的第二层级中的多个条目;

确定第一条目是否为分级索引表的末端条目;

基于确定第一条目不是末端条目,将具有第二长度的检索序列的第二子序列添加到第一子序列;

匹配检索序列的第二子序列与分级索引表的第二层级中的第二条目,其中

第二条目是由第一条目中的信息指示的第二层级中的多个条目之一。

2.根据权利要求1所述的方法,所述方法还包括:

迭代地匹配检索序列的一个或多个附加子序列与分级索引表的后续相应层级中的具有相应长度的相应条目,直到确定相应条目中的一个条目是末端条目;

响应于确定相应条目中的一个条目是末端条目:

从第一条目中读取参考数据中的第一位置;和

对检索序列执行候选评估。

3.根据权利要求2所述的方法,

其中对检索序列执行候选评估包括确定在所述检索序列和在所述第一位置处的参考数据之间是否存在少于阈值次数的错配。

4.根据权利要求1所述的方法,

其中匹配检索序列的第一子序列与第一条目是由分级索引表的有序数值结构促进的。

5.根据权利要求1所述的方法,

其中所述第一长度是可凭经验选择的,以改善与匹配检索序列与参考数据相关联的计算参数。

6.根据权利要求1所述的方法,

其中至少部分地基于分级索引表的统计特征来选择第一长度以改善与匹配检索序列与参考数据相关联的计算参数。

7.根据权利要求1所述的方法,

其中第一长度和第二长度是不同的。

8.根据权利要求1所述的方法,

其中所述参考数据包含参考基因组,并且所述检索序列包含对所述参考基因组的短读数(SR)。

9.根据权利要求1所述的方法,

其中参考数据和检索序列包括脱氧核糖核酸(DNA)的相应序列。

10.一种计算设备,包括:

处理器;和

耦合到所述处理器的存储器,在所述存储器上存储有计算机可读指令,所述指令在由所述处理器执行时,使所述处理器执行根据权利要求1-9中任一项所述的方法。

11.一种用于将检索序列与参考数据匹配的装置,包括用于执行根据权利要求1-10中任一项所述的方法的部件。

12.一种计算机可读存储介质,包括用于将检索序列与参考数据匹配的程序指令,所述指令在由一个或多个处理器执行时,使所述一个或多个处理器执行根据权利要求1-10中任一项所述的方法。

使用分级反向索引表的DNA比对

[0001] 本申请是申请日为2016年10月21日、申请号为201680061446.2、发明名称为“使用分级反向索引表的DNA比对”的发明专利申请的分案申请。

技术领域

[0002] 本申请大体上涉及将数据模式映射到参考数据组上，且更具体地说，涉及在DNA测序和DNA比对应用中执行这类数据比对或模式匹配。

背景技术

[0003] 现代技术涉及越来越大量数据的收集和处理。其中，所谓的“大数据”的应用和使用情况范围是数据挖掘、播发、机器学习和DNA测序。在许多情况下，有必要搜索在少量样品数据和大得多的参考数据组之间的匹配。随着参考数据组的尺寸增加，样本数据与此参考数据组的比对(模式匹配)变成以指数方式更为计算密集型任务。

[0004] 数据比对的示例性案例在DNA比对领域中进行。活生物体由细胞构成并且细胞的操作和繁殖受从一代细胞传送到下一代的基因信息控制。

[0005] 物种和个体生物体的基因信息的详细知识对于更精确生命科学的保持巨大希望，从而支持改善的健康护理、农业、环境管理和犯罪解析。

[0006] 实现这些益处的障碍中的一个为对生物体的基因信息进行测序的成本。为了做到这一点的技术已经在数十年的最后十年内显著改善，使得将成本减少到小于US$1000/人表现为可实现的。然而，仍然存在数据的完整性、精确度、解释的问题，和可靠诊断疾病的问题。从生物样品获取基因信息的天数也是需要快速响应的用途的障碍，如已知供急救室患者使用的对于敏感个体具有严重副作用的医药适合性。

[0007] 因此，期望用于数据比对并且具体来说DNA测序的改善的技术和工具。

发明内容

[0008] 公开用于将数据模式映射到显著地较大数据组上的系统和方法的各种实施例。在一些实施例中，较大数据组可为参考数据组。在一些实施例中，较大数据组可为从头测序的结果，其中多个数据模式用于构建与多个数据模式自一致的大数据组。本文中呈现的许多实施例涉及DNA比对的具体使用案例，其中参考数据组为参考基因组并且数据模式为衍生自DNA链的短读数(SR)的一串DNA碱基。然而，本文中详述的方法通常适用于将任何数据模式映射到较大数据组上的问题。本文关于DNA比对所描述的方法的说明旨在有助于解释，并且不意指以任何方式限制本发明的范围。本领域中技术人员将容易地参看本文所描述的方法可如何应用于除DNA比对以外的数据比对或模式匹配方法。

[0009] 在一个实施例中，可生成基于参考数据的分级索引表。分级索引表可包含其中多个数据段中的每一个所在的在参考数据中的位置。在计算机科学中，此形式的索引表可被称作反向索引表。分级索引表可用于将检索序列与参考数据匹配。索引表可经构建以含有与给定长度的所有子序列的穷尽性列表相关联的条目，其中每个条目含有在参考数据中的

每个子序列的匹配的数量和位置。可以迭代的方式构建分级索引表，其中基于匹配的数量大于一组相应阈值中的每一个，选择性地和迭代地构建用于每个延长子序列(层次的更深层级)的条目。对于一些子序列，匹配的数量将相等或小于当前阈值，为此方法生成在表中的末端条目。有限长度的参考数据意指，可发现足够长子序列的匹配的数量将低于给定正阈值。然而，据了解，存在大于1000bp长并且出现数千次的基因组的子序列。对于完全地索引，这些序列可为或不为所感兴趣的；并且在后一种情况下，某些序列可排除掉，而非包括于分级索引表中。

[0010] 用于在参考基因组中对SR执行候选位置选择(鉴别匹配模式)的方法可包括通过以迭代方式执行以下来检索对应分级索引表。可基于一段SR，生成“印迹”(由一串DNA碱基组成)，并且其用于从与参考基因组相关联的索引表选择SR的至少一个候选位置。印迹的长度可延长以便移动到分级索引表的更深层级。一旦达到分级索引表的末端条目，就可停止迭代，并且可输出候选位置。使用分级索引表可操作以大大地增加可出现候选选择的速度。

[0011] 在一些实施例中，方法还可包括在SR中评估每个候选位置。举例来说，在一些实施例中，对于每个候选位置，在SR和参考基因组之间的匹配可基于在SR和参考基因组之间碱基误差的数量来确定。在一些实施例中，评估还可包括确定在SR中的至少一个插入或缺失(共同称为插入缺失)。举例来说，在SR中的锚定位点可基于SR的lo端部和hi端部错配的数量来确定。可使用锚定位点确定至少一个插入缺失的长度和类型。可确定至少一个插入缺失的起始位点。在一个实施例中，确定起始位点可包括计算在SR和在候选位置处的参考基因组之间的第一运行误差和，和计算在SR和从候选位置偏移处的参考基因组之间的第二运行误差和，其中偏移是基于至少一个插入缺失的类型和长度。接着可基于第一和第二运行误差和的最小值确定至少一个插入缺失的起始位置。

附图说明

[0012] 当结合以下附图考虑优选实施例的以下详细描述时，可以获得对本发明的更好理解，其中：

[0013] 图1示出根据一个实施例的两个示例性DNA序列，具体地说示出在两个链中的碱基的互补和反向排序；

[0014] 图2示出根据一个实施例的具有示例性碱基ACGTCTGATTGACAAACTGATGCA的短读数(SR)；

[0015] 图3示出被配置成实施本发明技术的实施例的示例性系统；

[0016] 图4为示出在一些实施例中用于构建分级索引表的算法的流程图；

[0017] 图5为示出在一些实施例中用于构建分级索引表的第一层级的方法的流程图；

[0018] 图6为示出在一些实施例中用于构建分级索引表的第二层级的方法的流程图；

[0019] 图7示出根据一个实施例的应用于图2的序列的本发明比对技术；

[0020] 图8为示出在一些实施例中用于使用分级索引表搜索参考数据以便匹配检索序列的子区段的方法的流程图；并且

[0021] 图9为示出在一些实施例中表征和定位插入缺失的方法的流程图。

[0022] 虽然本发明可易有各种修改以及替代形式，但在附图中借助于实例示出并且在本文中详细地描述了其具体实施例。然而，应理解，本发明的附图和详细描述并不希望将本发

明限制于所公开的特定形式，而是相反，目的是涵盖通过所附权利要求书限定的本发明的精神和范围内的所有修改、等效物和替代方案。

具体实施方式

[0023] 以引用的方式并入

[0024] 以下专利/申请以全文引用的方式并入本文中，如同完全地和彻底在本文中阐述：

[0025] 美国专利第7,415,594号，标题为“具有散置失速传播处理器和通信元件的处理系统”，2003年6月24日提交，其发明人为Michael B.Doerr、William H.Hallidy、David A.Gibson和Craig M.Chase。

[0026] 美国专利申请第13/274,138号，标题为“多处理器系统中的停用通信”，2011年10月14日提交，其发明人为Michael B.Doerr、Carl S.Dobbs、Michael B.Solka、Michael R Trocino和David A.Gibson。

[0027] 术语

[0028] 以下是本申请中使用的术语词汇表：

[0029] 存储媒体-各种类型的存储器装置或存储装置中的任一种。术语“存储媒体”旨在包括设备媒体，例如，CD-ROM、软盘104或磁带装置；计算机系统存储器或随机存取存储器如DRAM、DDR RAM、SRAM、EDO RAM、Rambus RAM、等；或非易失性存储器如磁性媒体，例如，硬盘驱动器、光学存储器或ROM、EPROM、闪存等。存储媒体同样可包括其它类型的存储器或其组合。此外，存储媒体可以位于其中执行程序的第一计算机中，和/或可以位于经由例如因特网的网络连接到第一计算机的第二不同计算机中。在后一个实例中，第二计算机可以向第一计算机提供程序指令以用于执行。术语“存储媒体”可包括两个或更多个存储媒体，其可以驻留在不同位置中，例如，在经由网络连接的不同计算机中。

[0030] 载体媒体-如上所述的存储媒体，以及物理传输媒体，如总线、网络和/或传送信号如电或光信号的其它物理传输媒体。

[0031] 可编程硬件元件-包括包含经由可编程或固线式互连件连接的多个可编程功能块的各种硬件装置。实例包括FPGA(现场可编程门阵列)、PLD(可编程逻辑装置)、FPOA(现场可编程对象阵列)和CPLD(复杂PLD)。可编程功能块的范围可为细晶(组合逻辑或查找表)到粗晶(算术逻辑单位或处理器核心)。可编程硬件元件还可被称作“可重新配置的逻辑”。

[0032] 专用集成电路(application specific integrated circuit,ASIC)-此术语旨在具有其普通含义的完全广度。术语ASIC旨在包括定制用于特定应用而非通用可编程装置的集成电路，但ASIC可含有作为结构单元的可编程处理器核心。蜂窝电话单元、MP3播放器芯片和许多其它单一功能IC为ASIC的实例。ASIC通常在硬件描述语言如Verilog或VHDL中描述。

[0033] 程序-术语“程序”旨在具有其普通含义的完全广度。术语“程序”包括1)可存储于存储器中并且可由处理器执行的软件程序或2)可用于配置可编程硬件元件或ASIC的硬件配置程序。

[0034] 软件程序-术语“软件程序”旨在具有其普通含义的完全广度，并且包括任何类型的程序指令、代码、指令码和/或数据，或其组合，其可存储于存储器媒体中并且通过处理器执行。示例性软件程序包括以文本基编程语言(例如，必要或程序语言，如C、C++、PASCAL、

FORTRAN、COBOL、JAVA、汇编语言等)编写的程序;图形程序(以图形编程语言编写的程序);汇编语言程序;已经汇编到机器语言的程序;指令码;和其它类型的可执行软件。软件程序可包含以某一方式交互操作的两个或更多个软件程序。

[0035] 硬件配置程序-程序,例如,接线对照表或比特文件,其可用于编程或配置可编程硬件元件或ASIC。

[0036] 计算机系统-各种类型计算或处理系统中的任一种,包括个人计算机系统(PC)、主机计算机系统、工作站、网络电器、互联网电器、个人数字助理(PDA)、网格计算系统或其它装置或装置的组合。一般来说,术语"计算机系统"可宽泛地限定为涵盖任何具有执行来自存储媒体的指令的至少一个处理器的装置(或装置的组合)。

[0037] 自动地-是指通过计算机系统(例如,通过计算机系统执行的软件)或装置(例如,电路、可编程硬件元件、ASIC等)执行的动作或操作,而不需用户输入直接地指定或执行所述动作或操作。因此,术语"自动地"对应于通过用户手动执行或指定的操作,其中用户提供输入以直接地执行操作。自动步骤可通过由用户提供的输入开始,但是自动地执行的后续动作不由用户指定,即,不手动执行,其中用户指定执行的每个动作。举例来说,通过选定每个字段和提供指定信息(例如,通过打字信息、选定复选框、无线电选择等)的用户填写电子表单是手动填写表单,即使计算机系统必须响应于用户动作来更新表单。可由计算机系统自动地填写表单,其中计算机系统(例如,在计算机系统上执行的软件)分析表单的字段并且填充表单,而不需指定字段应答的任何用户输入。如上文所指出,用户可调用表单的自动填写,但是不涉及表单的实际填写(例如,用户不手动指定对字段的应答,而是,自动地完成应答)。本说明书提供响应于用户已经采取的动作自动地执行的操作的各种实例。

[0038] 开发过程-是指基于方法的用于开发的寿命循环。在粗略层级下,其描述如何通过设计、实施方案、检验、部署和维护来驱动用户要求和约束。

[0039] 概述:DNA测序

[0040] 对于碳基生命形式,基因信息以长链分子脱氧核糖核酸(DNA)编码,其中交替的糖和磷酸基的两个平行主干通过碱基对的梯子保持在一起。两个主干保护碱基对并且平缓地围绕彼此旋转以形成双螺旋构形。更长链的DNA形成卷曲,和卷曲的卷曲。对于更大生物体,这些卷曲组织到成色体,其为在非常高放大倍率下用光学显微镜可见。

[0041] 每个碱基为组{腺嘌呤、胸腺嘧啶、胞嘧啶、鸟嘌呤}中的一个。每个腺嘌呤碱基将仅与胸腺嘧啶配对,并且每个胞嘧啶将仅与鸟嘌呤配对。这些成对规则支撑DNA的复制。在DNA与在溶液中的某一酶组合时,双螺旋打开成一对单链(每个由单个主干和所附接碱基构成)。然后,在碱基、糖、磷酸盐和酶的混合液中,对于每个链,可使用原始链作为模板组装互补链。净结果为原始双螺旋复制成一对双螺旋。活体外复制可重复以产生实体上需要供应到测序机的尽可能多的DNA;然而,出于速度和成本原因,测序机已经进化到使用越来越少量的DNA。

[0042] 图1示出两个示例性DNA链,具体地说示出在两个链中的碱基的互补和反向排序。如图1示出,一连串的磷酸基(小型的圆形)和脱氧核糖(五边形)充当用于单个DNA链的主干,由于连接在成对碱基之间的稳定和可预测氢键而使核碱基分开。糖环的非平面形式和在碱基中氮原子的电子构形确保DNA链的可预测组装和每个碱基与其在另一链中的互补碱基的连接。

[0043] 测序机使用许多方法以测定DNA链中的碱基序列。尽管每个方法均不同，但行业已经稳定在用于报告原始序列数据的几个实际标准格式上。对于单链读数，这些格式以5'到3'方向报告碱基序列。此标号是指连接到相邻磷酸基的在脱氧核糖部分中的碳原子。5'原子不为在糖中的5元环的一部分，而是为延伸到前一磷酸基的肘部，其中"前一"的含义按照5'到3'定则来限定。图1示出此编号定则，其中1'位提供与碱基中的可用氮轨道的键。2'位为其中已经从核糖去除氧原子以形成脱氧核糖的位置。

[0044] 如图1表示，在左侧5'到3'序列为CAATCGTCA，并且在右侧为TGACGATTG，示出在另一链中的碱基的互补和反向排序。应注意，每个核苷酸的具体键在右侧示出。

[0045] DNA的片段可由在5'到3'方向上沿主干中的一个与碱基的序列对应的一串字母表示，例如ACGTCTGATTGACAAACTGATGCA。从化学方面看，众所周知，在原始方向上的互补主干上的碱基的序列为互补链，TGCAGACTAACTGTTTGACTACGT。然而，测序仪机器将不读取双螺旋，而仅读取单链，并且其将以5'到3'方向针对所述链将互补链报告为，TGCATCAGTTTGTCAATCAGACGT，其为原始链的反互补。图2示出具有其互补链的DNA的实例链。在处理原始数据时，重要的是，知晓相同基因信息在两个中链均编码。在将短读数与参考基因组进行比对时，有必要尝试针对原始短读数和短读数的反向互补序列均进行比对。

[0046] 在现有技术中，未开发可靠机器方法来在单次注射中对超出约2000个碱基对(bp)的DNA链进行测序[Sanger方法常规地进行～800bp]。然而，较长链可附接至底物并且从两端测序，产生通过以碱基对数量可测量的靶向分离相关的一对读数。在所有情况下，读数序列比在单个人类染色体(其可含有具有至多约249百万碱基对的DNA序列)(染色体1)中的碱基对数量短得多。对于在人类基因组中所有23个染色体，存在约32亿碱基对。

[0047] 自动化高速测序仪机器为用于分析来自单个个体的生物样品以产生数十亿相对较短读数(SR)(约100bp(碱基对)/SR)。然后，高速计算机用于将数十亿短读数与彼此比较并且搜索其中序列匹配的重叠。

[0048] 从头测序

[0049] 在足够重叠的情况下，超长序列的图片可积累。这被称作从头测序。如果个体的全序列积累(在人类的情况下为23个染色体)，则我们具有所述个体的基因信息。来自许多个体(足以涵盖物种的变异体)的基因的组合被称作所述物种的基因组。对于许多物种，细胞中的线粒体也含有具有用于基因组的额外基因信息的DNA。

[0050] 重测序

[0051] 一旦建立参考基因组，获得个体的基因信息和将原始读数与参考基因组匹配的过程被称作重测序。在其中读数序列已经完全或近似匹配的参考基因组中对位置进行搜索被称作比对或映射。因此，完全重测序可需要数十亿比对。

[0052] 重测序可用于鉴别对具有基因基础的疾病的敏感性，和鉴别对药物副作用的敏感性等。在个体之间基因比较可用于示出家族关系。对于重测序，参考基因组资料库已经存在并且其可经编索引以加速每个比对过程。下文详述的实施例呈现用于基于参考基因组资料库构建分级索引表的新颖且有利方法。

[0053] 一旦构建索引表，就可以以高速来搜索所述索引表很多次数，以比对大型组的原

始数据读数。可独立于其余来处理每次比对，这使其增添在大规模并联计算机上处理。然而，可存在在比对之间共用一些信息的优点。举例来说，如果发现读数不匹配标靶基因组中的任何内容，则所述读数可置于待预丢弃的读数的列表中，而不需再次搜索。这类不可匹配的读数可来自为原始样品的一部分的细菌DNA。

[0054] DNA测序仪机器设计中的趋势为更多产地和更快速地产生更短读数。来自个体人类的典型原始数据组可为数十亿短读数(SR)，每一个平均长度为约100bp。这些可为单股读数或成对读数。成对读数可具有从几个bp到数百(可能数千)bp的靶向分离。每个SR具有一定量噪声并且执行SR的机器产生用于读数中的每个碱基的质量度量。此每一碱基的质量信息可用于改善每个SR与基因组的比对。原始数据组可含有将不匹配人类基因组的来自细菌的额外DNA，并且这些读数应当在比对处理之前或在比对处理期间过滤出。

[0055] 图3:示例性系统

[0056] 图3示出被配置成实施本发明技术的实施例的示例性系统。如所示出，在一些实施例中，系统可为计算系统82如工作站，并且可包括一个或多个处理器和至少一个存储媒体，在所述媒体上可存储根据本发明技术的实施例的一个或多个计算机程序或软件组件。举例来说，存储媒体可存储可通过一个或多个处理器执行的一个或多个程序以执行本文所描述的方法。存储媒体还可存储操作系统软件，以及用于操作计算机系统的其它软件。

[0057] 在一些实施例中，系统可包括被配置成实施本文所公开的技术的多处理器阵列(MPA)。举例来说，MPA可借助散置处理器和通信元件实施处理系统，和/或可具有HyperX架构，如美国专利第7,415,594号中所描述，所述专利以引用的方式并入在上文中。

[0058] 本发明的实施例还可通过至少一个可编程硬件元件如现场可编程门阵列(FPGA)或通过可编程硬件元件和一个或多个软件可编程处理器的组合实施。

[0059] 然而，应注意，图3中所示的计算系统82仅意为示例性，并且根据需要可使用任何其它计算系统。

[0060] 系统方面

[0061] 本发明技术可包括数据流系统，其借助核心比对功能重叠数据输入和输出重叠以及预处理和后处理。为实现此，一个实施例使用通信套接字和双缓冲技术。

[0062] 处理SR对可以在核心比对阶段期间执行。这可通过在比对个体SR和可能比对类型的层次期间考虑潜在配合对实现。

[0063] 为了加速核心比对任务，对于比对不需要的辅助信息可传送到后比对阶段，这可降低在核心比对器中带宽和存储器要求。为实现此，一个实施例使用单独的通信套接字组。

[0064] 示例性实施例和实施方案

[0065] 下文描述本文所公开的技术的各种示例性实施例和实施方案。然而，应注意，描述的特定实施例和技术不将本发明限制于任何特定形式、功能或形态。

[0066] DNA比对可指发现给定DNA片段(即，短读数或SR)出现在参考基因组内的一个或多个位置的方法。

[0067] SR可包含许多DNA碱基，其中每个碱基为在DNA中发现的简称为A、C、G或T的四个(腺嘌呤、胞嘧啶、鸟嘌呤和胸腺嘧啶)中的一个。根据使用的测序方法，每个SR的长度通常可为约100个碱基长。较新测序方法的趋势已经产生更短SR，但是每单位时间产生更多短SR。如所指出，由于来自测序方法的噪声，每个SR可具有一些碱基误差。这些误差可通过冗

余和通过使用与每个SR一起出现的每个碱基的质量信息来减低。为了表征个体人类，期望通过SR具有全基因组的约50X冗余覆盖度。鉴于参考人类基因组含有超过30亿碱基，来自一个个体的SR数据的集合在其中可具有1500-2000亿碱基。每个SR可与基因组比较以发现SR在何处适配。明显地，期望使用非常高效搜索基因组来处理来自个体的大量SR数据。

[0068] 可通过将参考基因组编码成分级反向索引表来显著地降低DNA比对所需要的时间。下文给出用于此类反向索引表且更具体地说用于分级反向索引表的新颖形式和方法。

[0069] 在一个实施例中，可使用用于将给定SR与索引表进行比对的2-阶段方法。第一阶段可被称为候选选择，接着为第二阶段，其可被称为候选评估。

[0070] 候选选择可生成随后可评估精确度的潜在位置的列表。因为候选评估阶段可能非常耗时，所以重要的是，候选选择阶段智能地减少候选位置的数量。

[0071] 应注意这些(阶段)名称仅意为说明性和示例性的，并且根据需要可使用任何其它名称。此外，在各种实施例中，这两个阶段的功能性可按不同方式组织或布置，例如，分成多于两个不同地分隔的阶段等并且仍然实施本文所公开的新颖技术。

[0072] 图4-用于构建分级索引表的算法

[0073] 图4示出用于通过参考基因组构建分级反向索引表的示例性算法。术语“反向索引表”为在计算机科学领域中的术语并且是指存储含有从内容(如数据(例如，数量或单词)到此内容在数据组中位置的映射的条目的索引数据结构。“反向索引”的一个日常实例为书的索引，其含有在书中呈现的单词或短语的列表，每个后跟着可发现此相应单词或短语的在书中的位置。为方便起见，术语“索引表”在本文中用以指在计算机文献中称为“反向索引表”的要素。在术语“分级反向索引表”中使用的术语“分级”是指具有以分级方式的各种层级的索引表的条目，如下文进一步所解释的。

[0074] 在本发明技术的一些实施例中，可使用一组k-聚体(一串k碱基)构建分级索引表以高效地确定一组候选位置。一旦针对参考基因组生成索引表，并且然后可由待与所述参考基因组比对的所有组SR使用。应了解，本文所描述的方法可更广泛地对任何参考数据执行，并且可由待与所述参考数据比对的任何组子序列使用。

[0075] 在创建分级索引表时，计算装置可被配置成将参考数据(基因组)存储到存储器中。然后，计算装置可基于参考数据通过创建处于多个层级的多个条目来创建分级索引表。如下文更详细所解释的，构建索引表开始于创建第一层级条目(层级1)，其包含在预定长度的子序列的(优选地穷尽性的)列表或组(或预定尺寸数据组的部分)上的信息。信息可包含每个相应子序列在参考基因组中的位置信息。

[0076] 对于处于每个相应层级n的每个条目(其中n为非零正整数)，计算装置可被配置成如果相应层级n条目的匹配准则为大于阈值，针对相应层级n条目在分级索引表中创建额外n+1个层级条目。每个n+1层级条目可将额外长度添加到前一层级中的子序列，因此创建更长子序列用于潜在匹配。类似于第一层级条目，每个n+1层级条目含有每个相应更长子序列在参考基因组中的位置信息。此创建额外层级的更长子序列继续，直至匹配准则不再满足为止-在末端条目(在分级分支中的最后一个条目)处的匹配的数量低于阈值。应注意，使用较低阈值可操作以减少在每个末端条目处的匹配的数量。相比之下，使用较高阈值可有助于减少在索引表中条目的总体数量。应注意，可在层次的不同层级处使用不同阈值。因此，在所得完整的分级索引表中，不同相应层级1条目可具有衍生自所述相应层级1条目的层次

的不同数量后续层级，其中层次的后续层级的数量取决于在每个后续层级处的匹配的数量。

[0077] 因此，索引表可表征为分级或多层级。在完成构建分级索引表时，计算装置可被配置成接收指定检索序列的输入，并且可使用分级索引表以在大大地加速的速率下搜索在参考数据中检索序列的子区段的匹配。

[0078] 在一些示例性实施例中，算法可用于如下构建分级索引表。可选择编码方案用于以二进制格式编码四个碱基对。在一些实施例中，可使用2-位编码方案，其中每个碱基对与唯一2-位标识符相关联(例如，A＝00，T＝01，G＝10，并且C＝11)。在其它实施例中，可使用1位编码方案，其中两个碱基对与每个唯一1位标识符相关联(例如，除其它可能性外，A和T可编码为0，而C和G可编码为1)。例如在执行快速近似候选选择过程中，1位编码方案可为有利的。在这类方案中，虽然准确的候选选择将不实现，但可裁定出大致一半的候选位置用于候选选择，显著地加速对于使用2-位编码方案的后续候选选择过程所需要的计算时间。

[0079] 一旦已经选择编码方案，就可基于编码方案将参考基因组转译成连续系列比特。举例来说，如果参考基因组包含N个碱基对并且使用2-位编码方案，则参考基因组将转译成长度2N的比特串。根据需要，经编码的参考基因组可进一步包含染色体分隔物标示物、标头字段等。

[0080] 在402处，可选择一组$L=\{l_1,_{L2},l_3,\ldots,l_{L最大}\}$碱基对长度。矢量L的要素表示与索引表的顺序层级相关联的长度。举例来说，l_1为与第一层级条目相关联的序列的长度，l_2为与第二级条目相关联的序列的长度，诸如此类。举例来说，在图2中，其示出DNA链的循序地增加的段，长度l_1可对应于标记‘初始’的区段。标记‘第1ext’的区段可对应于长度l_2，并且标记‘第2ext’的区段可对应于长度l_3，诸如此类。

[0081] 组L可经选择以改善索引表构建过程和/或短读数比对过程的计算时间和/或精确度。在一些示例性实施例中，第一长度l_1可经选择使得$4\hat{}(l_1)<N$。这可例如通过确保多数长度l_1的碱基对序列将(统计上)出现在参考基因组中的至少一个位置中来证明是有利的，因此减少致力于不成功检索的计算时间。在一些示例性实施例中，要素L的总和(我们可将其称作$L_{总}$)可经选择使得$4\hat{}(L_{总})>>N$。在此情况下，长度相等$L_{总}$的一串无规碱基将统计上以可忽略的机率出现在一串无规N碱基中。举例来说，最长人类染色体长约250百万个碱基。存在$4^{20}\approx1$万亿不同串的20个碱基对，使得设定$L_{总}=20$将引起任何特定20碱基对串出现在无规250百万碱基对串中的机率为0.025％。适当地，将$L_{总}$增加到25通过$4^5=1024$的因数，将此机率降低到0.000022％，使得对于任何实体上相关值N，无规统计一致的概率可容易地降低到可忽略的量值。由于无规相关性，这可通过例如降低SR匹配到参考基因组中的位置的概率而为有利的，并且从而增加匹配指示在SR和参考基因组中匹配位置之间的下层生物关系的机会。在其它示例性实施例中，组L可凭经验确定以改善(或可能优化)表构建过程和/或短读数比对过程的计算参数和/或精确度。

[0082] 在本实施例中，术语长度用于指碱基对序列的直线程度。然而，在其它实施例中，应了解，‘长度’可更一般化地表示与数据的集合有关的任何尺寸度量。举例来说，‘长度’可通用化以表示多维阵列、多个像素(例如，像素阵列(4×4等)的维度等。

[0083] 在404，可选择数目阈值的组$C_{th}=\{th_1,th_2,th_3,\ldots,th_{L最大-1}\}$。矢量$C_{th}$的要素表示与索引表的相应层级相关联的阈值。换句话说，矢量C_{th}的要素表示在应用在索引表的每个

相应层级处的匹配准则中使用的阈值。

[0084] 组C_{th}可经选择以改善(或可能优化)索引表构建过程和/或短读数比对过程的计算时间和/或精确度。在各种实施例中,所述改善可基于参考基因组的统计特征,或所述改善可凭经验确定。如将在下文所见,可通过迭代地构建适应性地分支树进行表构建算法。在一些实施例中,可针对树的每个分支适应性地确定数目阈值。举例来说,可设定初始数目阈值th_1,并且可在每当树分支时适应性地调节初始数目阈值,其中适应性调节是基于在所述迭代期间形成的新分支的数量。

[0085] 在406,创建具有第一长度的可能子序列的优选地穷尽性的列表。更具体地说,构建可能的每串长度l_1的碱基对的有序列表M_1,其中使用用于参考基因组的相同编码方案,将可能的每串长度l_1的碱基对同样转译成一串比特。列表可按二进制数值顺序进行排序(即,每串碱基对可转化成二进制数字并且这些数字可按递增数值顺序在M_1中排序)。对于在此有序列表中的每个串,算法可被配置成为匹配串检索参考基因组。算法可进一步被配置成存储匹配串在参考基因组中的位置和发现为数据结构中的条目的匹配串的数量,其中用于在第一有序列表中的串中的每一个的条目以与它们在第一有序列表中出现的相同顺序存储。在数据结构中的这些条目可共同包含索引表的第一层级。

[0086] 在一些实施例中,算法可进一步被配置成构建索引表的第二层级。在416,第二有序列表M_2可由可能的每一串长度l_2的碱基对构建,其中使用用于参考基因组的相同编码方案,将可能的每一串长度l_2的碱基对同样转译成比特串,并且列表以二进制数值顺序排序。

[0087] 在408,对于索引表的第一层级的每个第i条目,算法可比较存储于第i条目中的匹配位置的数量与第一计数阈值th_1。

[0088] 在410,如果存储于第i条目中的匹配位置的数量不超过th_1,则算法可不通过第i条目构建第二层级条目,并且可将与第i条目相关联的停止(STOP)指令存储在数据结构中。这可导致相应条目变成索引表中的末端条目。

[0089] 在416,如果在第i条目中发现的匹配位置的数量超过th_1,则算法可针对在M_2中的每个条目$M_2(j)$,检索存储于第i条目中的在每个相应位置处的参考基因组并且检查l_2碱基对(在每个相应位置外以距离l_1起始)是否匹配$M_2(j)$。针对被执行此检索的每个条目$M_2(j)$,算法可将其中针对$M_2(j)$发现匹配的位置以及针对$M_2(j)$发现的匹配的数量存储为数据结构中的条目,其中用于第二有序列表中的串中的每一个的条目以与它们出现在第二有序列表中的相同顺序存储。这些条目可共同包含索引表的第二层级。

[0090] 在一些实施例中,索引表的第三和所有后续层级可以迭代方式构建。应了解,第三层级可类似于第二层级构建,其中具有以下调整。在前述段落中的下标'1'和单词'第一'的所有例子可分别用'2'和'第二'替换。在前述段落中的下标'2'和单词'第二'的所有例子可分别用'3'和'第三'替换。可对于分级索引表的每个后续层级进行类似的调整,直至$L_{最大}$层级为止。此外,在构建第三层级期间,在检查M_3的每个条目$M_3(j)$是否匹配存储在索引的第二层级的每个第i条目中的在每个相应位置处的参考基因组时,算法应当在每个相应位置以外以距离l_1+l_2开始。在确定$M_n(j)$是否匹配参考基因组时,分级索引表的后续层级n将在每个相应位置以外以距离$l_1+l_2+\ldots+l_{n-1}$开始。

[0091] 在每个迭代处,在418,可确定j是否大于$L_{最大}$。如果j不大于$L_{最大}$,则算法可返回到408并且创建索引表的后续层级。如果确定j大于$L_{最大}$,则算法可在420结束。

[0092] 应了解，在构建索引表的第j层级期间，如果在第(j-1)层级的每个条目处发现的匹配位置的数量不超过第(j-1)阈值，则第(j-1)层级的所有条目可在412与停止指令相关联，并且索引表的构建可在414结束，而不需前进到后续层级。

[0093] 进一步应了解，本发明包含选择性地构建索引表的分级条目，使得仅针对实际上出现在参考基因组中的碱基对(bp)的序列构建条目。用于一串碱基对的可能碱基对序列的数量与串的长度一起以指数方式增长，使得针对甚至25个碱基对的适度串长度，具有用于所有可能串的条目的穷尽性的索引表变得过分地大($4^{25}\approx$1千万亿条目)。多数基因组含有碱基对的所有可能序列的远小得多的子集，其不可能大于基因组的长度。选择性地构建分级条目将在每一层级处的条目数量限制于参考基因组的长度，或在人类基因组的情况下32亿个。

[0094] 实际基因组具有许许多多重复序列，这减少在层次的第一层级处的条目数量。对于人类基因组，最常见重复序列被称作“LINE-I”；并且其长度在1000和6000个bp之间，并且其可存在100000个复制品，占全基因组中32亿bp的至多14.6％。

[0095] 在一些实施例中，在条目存储在分级索引表的层级中时，算法可被配置成穿过存储在索引表的前述层级的对应条目中的位置，使得这些位置从前述层级的对应条目中删除(位置的数量将仍然存储在前述层级的对应条目中)。在这些实施例中，在参考基因组中的位置将仅存储在索引‘树’的每个‘分支’的末端条目中。因为索引表可具有非常大量分支，所以这可显著地降低索引表的尺寸，这可减少计算开销。

[0096] 在一些实施例中，在构建过程期间或经由后处理，索引表可分成两个数据结构，其中指示与每个条目相关联的位置的数量的数据和指示与每个条目相关联的在参考基因组中的位置的数据各自存储在单独的数据结构中。

[0097] 索引表的每一层级的条目可另外存储与在索引表的后续层级处的相关条目的关联。举例来说，如果在层级n处的相应条目不为末端条目，则其可包含表示条目(衍生自相应条目)在n+1层级处的位置的指标。相应条目可进一步存储与索引表的后续层级相关联的长度l_{n+1}。指标和长度信息可例如使索引表的检索被快速地引导到对应于任何特定检索序列的在索引表中的条目。

[0098] 图5-分级索引表的第一层级的构建

[0099] 图5为示出可由参考数据构建分级索引表的第一层级的示例性过程的流程图。

[0100] 在502，可构建具有第一长度的第一子序列。第一长度可经选择，使得第一长度的任何无规子序列统计上可能在参考数据中出现至少一次。可用数字例如以二进制编码子序列。在示例性实施例中，子序列可包含一串DNA碱基对，其可以1位或2位二进制格式编码。第一子序列可经构建，使得其用数字编码时具有最低可能数值量值。举例来说，在一串DNA碱基对的2位编码情况下，第一子序列可为一串‘0’，其为第一长度的两倍。

[0101] 在504，创建在分级索引表的第一层级中的第一条目。第一条目与第一子序列相关联。

[0102] 在506，在参考数据中进行检索第一子序列的匹配。检索可为穷尽性的，从而其检索整个参考数据用于匹配。检索可记录匹配的数量和其在参考数据中的位置中的每一个。

[0103] 在508，匹配的数量和其位置中的每一个可存储在分级索引表的第一层级中的第一条目中。在一些实施例中，匹配的数量可存储在第一数据结构中，而匹配的位置可存储在

第二数据结构中，其中第一数据结构进一步包含相应条目在第二数据结构中的指标。

[0104] 在510，步骤502至508中的每一个可对于第一长度的每个子序列重复。举例来说，如果子序列为一串DNA碱基对，则步骤502至508可针对第一长度的碱基对的每一组合进行重复。每个顺序子序列可经选择，使得用数字编码的子序列以递增数值顺序出现。举例来说，在上文给出的实例中，对于其中2位(二进制)编码方案用于一串DNA碱基对的步骤5102，第二子序列可经选择为‘000...00010’，其中椭圆包含适当数量的‘0’，使得二进制串的长度为第一长度的两倍。在此情况下，在索引表的第一层级中的条目可以递增数值顺序创建，使得可很一般地确定与第一长度的任何给定子序列对应的条目在索引表中的位置。

[0105] 图6-分级索引表的第二层级的构建

[0106] 图6为可构建分级索引表的第二层级的示例性过程的流程图。

[0107] 在602，可构建具有第二长度的第一子序列。可凭经验或基于参考数据的统计参数确定第二长度，其方式为以便改善将数据模式映射到参考数据的过程的计算参数和/或精确度。类似于第一长度的第一子序列，具有第二长度的第一子序列可用数字编码，并且可经选择以具有最小数值量值。

[0108] 在604，可创建在分级索引表的第二层级中的第一条目。第一条目可与第一子序列相关联。

[0109] 在606，对于分级索引表的第一层级中的每个条目，可确定在所述条目中的匹配数量大于第一阈值。可凭经验或基于参考数据的统计参数确定第一阈值，其方式为以便改善将数据模式映射到参考数据的过程的计算参数和/或精确度。

[0110] 在608，在参考数据中进行检索第一子序列的匹配。在608，检索不为整个参考数据的穷尽性检索。相反，仅在与索引表的第一层级中的相应条目相关联的参考数据中的位置处执行检索。具体来说，对于用于在606提及的第一层级中的相应条目的参考基因组中的每个匹配位置，在所述位置加第一长度处进行检索。换句话说，进行检索：第一长度的每个匹配是否另外匹配后续第二长度。

[0111] 在610，匹配的数量和在608发现的每个匹配的位置存储在索引表的第二层级的第一条目中。在一些实施例中，匹配的数量可存储在第一数据结构中，而匹配的位置可存储在第二数据结构中，其中第一数据结构进一步包含相应条目在第二数据结构中的指标。

[0112] 在612，步骤602至610中的每一个可针对第二长度的每个可能子序列进行重复。举例来说，如果子序列为一串DNA碱基对，则步骤602至610可针对第二长度的碱基对的每一组合进行重复。每个顺序子序列可经选择，使得用数字编码的子序列以递增数值顺序出现。在此情况下，，将对应于在第一层级中条目(其中在606确定超过第一阈值)中的每一个创建在第二层级中的一组条目，并且这些组中的每一个将在内部以递增数值顺序排序，使得可很一般地确定与第二长度的任何给定子序列对应并且与第一长度的任何给定子序列相关联的条目在索引表中的位置。

[0113] 示例性候选过程

[0114] 在本文中我们详述用于使用分级索引表针对SR执行候选选择的示范性算法。首先，SR的长度K的印迹F可经选择用于与参考基因组比较。在一些实施例中，长度K可经选择使得在长度K内的错配或插入缺失的机率小于预定阈值。在一些实施例中，长度K可被选择为L中的第一条目，即，l_1。

[0115] 在一些实施例中，长度K可迭代地调适以改善DNA比对过程的计算要求。举例来说，初始长度K可经选择并且可在索引表中查找相应印迹。如果在查找条目处的候选位置的数量高于位置的预定阈值数量，则长度K可递增地增加预定长度并且可在索引表中查找延长印迹。新查找的条目的位置的数量可大于位置的预定阈值数量，并且长度K可再次递增地增加预定长度并且可在索引表中查找延长印迹等。此过程可迭代地重复，直至发现候选位置的数量不超过位置的预定阈值数量为止。候选评估为计算密集型任务，并且将候选位置的数量减少到可管理量可显著地降低DNA比对过程的计算负担。

[0116] 在一些实施例中，位置的预定阈值数量可包含在一组阈值C_{th}中，其在构建分级索引表中使用。举例来说，候选选择过程可被配置成仅当与印迹对应的索引表中的条目为索引表中的末端条目时，即，仅当条目与停止命令相关联时，才复位参考基因组中的匹配位置。在一些实施例中，如果与印迹相关联的条目不与停止命令相关联，则候选选择过程可被配置成将印迹的长度增加与索引表的后续层级相关联的长度，并且在用于延长印迹的索引表中查找对应条目。

[0117] 图7-借助分级索引表执行检索的实例实施方案

[0118] 递增地增加K的过程的实例实施例在图7中示出，其示出应用于图2的SR的实例检索算法。在图7中，初始长度K表示为‘初始’，并且算法涉及索引表中的amt_loc表0中的第一条目。在amt_loc表0中的‘f’字段描述停止命令是否与所述条目相关联(S)或不相关联(/)。发现，第一条目不为索引表中的末端条目，在其上长度K递增地增加量“第1ext”并且算法被引导至在amt_loc表1中的新对应条目。此处再次发现，新对应条目同样不为末端条目，并且长度K随后再次增加量“第2ext”并且算法被引导至在amt_loc表2中的另一对应条目。此处发现，对应条目为末端条目，使得算法被引导至bas_loc表2中的特定行，其中匹配延长串F的基因组中的位置(图7中的ppp)从与延长串F对应的bas_loc表2中的第一条目开始读出。

[0119] 应了解，根据一些实施例，表，amt_loc表0、amt_loc表1和amt_loc表2包含分级索引表的顺序层级。应进一步了解，根据一些实施例，在‘amt_loc’表和‘bas_loc’表组之间的区别说明关于所发现匹配的数量和那些匹配的位置的信息分离成两个不同数据结构。

[0120] 在各种实施例中，印迹可选自SR的起点、中间或末端。在一些实施例中，长度K的单独印迹可选自SR的起点、中间和/或末端中的每一个，并且从每个印迹中发现的匹配可在候选选择过程中聚合在一起。

[0121] 在DNA比对中，通常必需考虑相对于参考基因组的SR的正向方向和“反向互补”方向。反向互补方向为SR(读数反向)并且将所有A碱基转译成T，T碱基转译成A，C碱基转译成G，并且G碱基转译成C。举例来说，以下SR片段与以下反向互补片段相关联：

[0122]

ACGTCTGATTGACAAACTGATGCA	原始
TGCATCAGTTTGTCAATCAGACGT	反向互补

[0123] 因此，可在两个方向上选择印迹以确定候选位置。在一些实施例中，进一步可为有利的是，使用反向印迹(非互补的)和/或互补印迹(非反向的)执行候选选择。

[0124] 在一些实施例中，有效性或质量评分可指派给SR中的每个碱基对，其中有效性或质量评分涉及SR中碱基对的预期可靠度。举例来说，以高置信度或保真度读取的SR中的碱基对可分配有较高有效性或质量评分。在这些实施例中，印迹F可优先选自包含定位于F内

的最高合计质量评分的碱基对的SR的一部分。在其它实施例中，具有低质量评分的碱基对存在于印迹中可导致在所述碱基对处检索裂开。举例来说，如果印迹的第5碱基对表示为具有低质量评分的C，则候选检索继续进行4次独立检索，其中对于所述4次独立检索，第5碱基对经取代为A、C、G和T中的每一个。对于这4次检索中的每一个发现的匹配接着可在检索过程的最终输出中合计在一起。应理解，此实施例可扩展到多个碱基对与低质量评分相关联的情况。

[0125] 如果检索到F和参考基因组之间的准确匹配，则算法可被配置成使用在索引表的后续层级中存储的与条目的关联来查找在索引表中对应于F的条目。接着可读出包含于与F对应的条目中的位置并且随后用于候选评估。需要许多现存DNA比对算法以执行二进制或其它计算密集型检索以发现在参考索引表中匹配F的条目。本发明中的索引表的系统性组织可显著地减少计算工作量和时延，因为为了发现在索引表中与F对应的条目，需要读取并且处理仅索引表的相关条目的指针信息，而非整个存储在索引表中的信息。

[0126] 在一些实施例中，可期望在参考基因组中发现大致匹配F的候选位置。举例来说，可期望在参考基因组中存储具有少于x个与F的错配的所有位置。在这些情况下，用于执行候选选择的算法可被配置成如下。

[0127] 算法可首先比较F的第一l_1碱基对与M_1中的每个条目。算法可存储含有少于x个与F的错配的M_1中的每个条目。对于每个存储的长度l_1的条目，算法可将每个长度l_2的串附加在索引表的第二层级中并且比较所得串与F的第一l_1+l_2碱基对。对于产生少于x个错配的每次这类比较，算法可类似地进行到索引表的第三和后续层级。此过程可迭代直至针对在索引表的特定层级中的特定条目遇到停止指令为止，或直至所发现错配的数量超过x。接着可读出与所遇到停止指令相关联的与索引表中每个条目相关联的位置并且用于候选评估。

[0128] 在一些实施例中，算法可被配置成使用在参考基因组和F之间错配的数量的运行和，同时确定用于候选选择的近似匹配。在此情况下，算法可被配置成类似于前述段落执行，但是存在若干修改。算法首先可比较F的第一l_1碱基对与M_1中的每个条目。算法可存储含有少于x个与F的错配的M_1中的每个条目，并且可存储针对每个存储条目记录的错配的数量的运行和。对于每个存储的长度l_1的条目，算法可比较每个长度l_2的串与F的后续l_2碱基对，并且将所发现错配的数量添加到运行和。对于小于x的每个运行和，算法可类似于索引表的第三和后续层级来进行。此过程可迭代，直至对于在索引表的特定层级中的特定条目遇到停止指令为止，或直至发现运行和超过x(无论哪个首先发生)。接着可读出与所遇到停止指令相关联的与索引表中每个条目相关联的位置并且用于候选评估。

[0129] 在一些实施例中，应了解，前述候选选择算法为自身提供迅捷并行度。举例来说，因为每个SR的候选选择过程是以编程方式独立的并且使用常见索引表，所以用于许多SR的候选选择算法可同时执行，从而显著地加速计算性能。

[0130] 图8-将检索序列与参考数据匹配

[0131] 图8为示出使用分级索引表将检索序列与参考数据匹配的示例性过程的流程图。

[0132] 在802，在分级索引表中查找检索序列的子区段。检索序列的子区段的长度可凭经验或基于分级索引表的统计表征选择，以改善匹配过程的计算参数。由于索引表的有序数值结构，所以查找过程可被快速地被引导到在索引表中对应于检索序列的子区段的条目。接着可读取此条目以确定包含于条目中的信息。

[0133] 在804，可确定在802中发现的条目是否为索引表的末端条目。末端条目可为与停止命令相关联的条目，即，末端条目可不与索引表的更高层级中的任何条目相关联。

[0134] 在806，基于确定条目不为末端条目，检索序列的子区段的长度可增加第一长度。可将第一长度选择成等于与索引表的后续层级相关联的长度。

[0135] 在808，使用分级索引表，进行参考数据的检索，用于匹配所增加长度的检索序列的子区段。808可类似于802进行，但是具有检索序列的延长子区段。

[0136] 在810，步骤804至808可迭代地重复，直至确定查找条目为索引表的末端条目。可根据需要借助唯一第一长度执行步骤804至808的每次迭代。如果发现查找到条目为索引表的末端条目，则匹配过程可读取从分级索引表中发现的匹配的位置并且进行到候选评估。

[0137] 候选评估

[0138] 一旦一组候选位置已经选用于特定SR，候选评估过程就可实施以比较SR的全部内容与参考基因组。在候选评估过程中，错配(其中碱基相比于参考基因组中不同)可被允许，并且实际上在生物学上是极其令人感兴趣的。一般来说，有效比对为其中在比对位点处SR和参考基因组之间错配的数量在可接受极限(即，错配数量可大于零但是小于某一预定阈值)内的比对。在此情况下，在比对中错配的表征和位置可为计算的重要输出。

[0139] 错配可采取取代的形式，其中一个碱基对经另一个碱基对取代。错配还可为插入，其中SR具有不出现在参考基因组中比对位点处的所插入1或更多个碱基。同样，错配可为缺失，其中一个或多个碱基从SR中失去，但是在参考基因组中存在。插入和缺失共同称为‘插入缺失’。

[0140] 举例来说，24碱基SR ACGTCTGATTGACAAACTGATGCA可具有这些类型的比对：

[0141]

```
参考：ACGTCTGATTGACAAACTGATGCA
SR：ACGTCTGATTGACAAACTGATGCA
错配：·····························        0错配；“一致”匹配

参考：AGGTCAGATTGACAAACTGATGCA
SR：ACGTCTGATTGACAAACTGATGCA
错配：.!...!..................        在“准确”匹配中2个错配

参考：ACGTC___TGATTGACAAACTGATGCA
SR：ACGTCGGATGATTGACAAACTGATGCA
错配：.....!!!.....................        由于插入的3个错配

参考：ACGTCTGATTTACGGACAAACTGATGCA
SR：ACGTCTGATT____GACAAACTGATGCA
错配：.........!!!!.............        由于缺失的4个错配
```

[0142] 也可出现以上内容的组合，例如：

参考:: ACGTATGATTTACGGACAAACTGCGGCA

[0143] SR: ACGTCTGATT____GACAAACTGATGCA

错配: ....!.....!!!!.........!!... 总计 6 个错配

[0144] 在准确匹配中定位错配相对地容易:仅需要逐个地比较碱基。定位插入和/或缺失(统称为插入缺失)要求多得多的处理,并且因此为更复杂的方法。

[0145] 鉴别可能含有插入缺失的SR

[0146] 作为第一步骤,可确定SR是否可能相对于基因组中通过候选选择鉴别的位置含有插入缺失。此步骤将根据F是选自SR的起点、中间还是末端而不同地进行。在其中F选自SR的起点情况下,可记录在SR和参考基因组之间的错配的数量的运行和(从最后位点到SR中的第一位点)。如果插入缺失存在于SR中,则相比于插入缺失的位置而距SR的起点较远地定位的碱基对可含有与参考基因组平均$3/_4$时间的错配(因为存在4个碱基对并且偏移将导致它们随机匹配,机率为25%)。从SR的末端开始,可记录每碱基对的错配的数量的运行平均值。如果对于预定数目的后续碱基对,运行平均值在预定阈值$^3/_4$内,则可确定可能存在插入缺失。应了解,DNA序列不包含无规碱基对序列,并且其它实施例可使用非$^3/_4$的平均错配比率。举例来说,可凭经验或通过DNA序列的统计考虑因素确定比率。

[0147] 在其中F选自SR的末端的情况下,类似的过程可用于确定可能存在插入缺失。在此情况下,从SR的起点开始,可记录每碱基对的错配数量的运行平均值。如果对于预定数目的后续碱基对,运行平均值在预定阈值$^3/_4$内,则可确定可能存在插入缺失。

[0148] 在一些实施例中,根据需要,预定阈值和后续碱基对的预定数目可调适以改善插入缺失识别过程的运行时间和精确度。

[0149] 在一些实施例中,可通过使用错配的运行和计算错配频率的梯度变化来确定可能存在插入缺失。举例来说,可使用来自SR的末端的错配的运行和计算平均错配频率。如果发现平均错配频率在SR中的特定位点处变化,其中发现变化大于预定阈值量值并且此外发现所述变化持续预定范围的碱基对,则可确定插入缺失可能存在于SR中。通过计算错配频率在SR长度上的梯度,可类似地鉴别可能的错配,并且如果在SR的范围内梯度尖峰超过预定阈值,则考虑可能现存插入缺失。

[0150] 表征并且定位插入缺失

[0151] 本发明技术的一个实施例使用4步骤式方法来评估用于插入缺失的候选位点:

[0152] 步骤1:Lo-末端和Hi-末端错配

[0153] 步骤2:长度和类型

[0154] 步骤3:误差和

[0155] 步骤4:插入缺失位点

[0156] 此方法可引起发现4种类型的插入缺失:使用lo-末端锚点的缺失,使用lo-末端锚点的插入,使用hi-末端锚点的缺失,和使用hi-末端锚点的插入。在下文中关于实例论述这些类型中的一个(使用hi-末端锚点的缺失)。应注意本文所描述的其它示例性情况为类似的。

[0157] 术语“lo-末端”和“hi-末端”是指SR的相应末端。举例来说,lo-末端碱基为在SR的lo-末端处的碱基。使用lo-末端锚定位点意指将lo-末端保持在恒定位点处,并且使hi-末

端位点变化,如同碱基缺失或插入到SR的中间中。同样,使用hi-末端锚定位点意指将hi-末端保持在恒定位点处,并且使hi-末端位点变化,如同碱基缺失或插入到SR的中间中。

[0158] 步骤1:Lo-末端或Hi-末端锚点的选择

[0159] 在此步骤中,S碱基的检测器尺寸可用于检查SR的lo-末端和hi-末端。可计数在SR的lo-末端S碱基和在SR候选比对位点处参考基因组的lo-末端S碱基之间的错配的数量;这些为lo-末端错配。同样,可计数在SR的hi-末端S碱基和在SR候选比对位点处参考基因组的hi-末端S碱基之间的错配的数量;这些为hi-末端错配。应注意S可为可调的参数;一个实施例使用24个碱基。

[0160] 如果lo-末端错配的数量大于阈值U并且hi-末端错配小于阈值M,则可表示(建议)使用hi-末端锚定位点。同样,如果hi-末端错配大于阈值U并且lo-末端错配小于阈值M,则可表示(建议)使用lo-末端锚定位点。如果这些条件均不成立,则可停止针对此候选位点的插入缺失搜索。应注意,U和M为可调的参数;一个实施例分别使用4和6。

[0161] 在一些实施例中,如果通过比较lo-末端印迹与参考基因组来选择候选位置,则可自动地选择lo-末端锚点,因为lo-末端已知与参考基因组中的候选位置适当对准。类似地,如果通过比较hi-末端印迹与参考基因组来选择候选位置,则可自动地选择hi-末端锚点。在其中选自SR中间的印迹用于选择特定候选位置的情况下,可为有利的是执行上述锚点选择过程。在这些情况下,SR的中间可与候选位置进行比对,但是可能未知潜在的插入缺失是否存在于SR的lo或hi侧上。

[0162] 步骤2:长度和类型

[0163] 如果疑似有插入缺失,下一步骤可确定插入缺失的长度和类型(插入或缺失)。用户可提供搜索的最高插入缺失长度(IDIST)。根据其中疑似有插入缺失的位置,一个实施例使用SR的lo-末端或hi-末端部分(与锚点末端相对)执行一系列错配计数。计数可使用B碱基,并且可通过D=[-IDIST..+IDIST]变化候选位点。应注意B为可调的参数;一个实施例使用值8。可记录偏移值D,其可引起最小数量的错配。在一个实施例中:

[0164] 如果D=0,则没有插入缺失,并且对于此候选,搜索停止。

[0165] 如果D<0,则发现可能插入,长度=|D|。

[0166] 如果D>0,则发现可能缺失,长度=D。

[0167] 鉴于插入缺失的长度和类型,应当发现起始位点。这可在接下来两个步骤(误差和和插入缺失位点)中执行。

[0168] 步骤3:误差和

[0169] 在此步骤中,可计算两个运行误差和。第一运行和可在SR和在候选位点处的参考基因组之间,并且第二运行和可在SR和在(候选+D)位点处的参考基因组之间;应注意,对于插入情况,D可为负。

[0170] 步骤4:插入缺失位点

[0171] 最终步骤是定位插入缺失的开始。这可通过使用基于类型的调节来发现两个计数阵列的总和的最小值来执行。

[0172] 在插入的情况下,可计算以下(注意:标号在1处开始,而非0):

[0173] 对于在[1..碱基-长度]中的i,总和[i]=(计数2[碱基]-计数2[长度+i])+计数1[i]

[0174] 接下来,发现总和[i]的最小值,其中i在[2..碱基-长度]中。插入的起始位点为i。

[0175] 在缺失的情况下,计算:

[0176] 对于在[1..碱基]中的i,总和[i]＝(计数2[碱基]-计数2[i])+计数1[i]

[0177] 然后,发现总和[i]的最小值,其中i在[2..碱基]中。缺失的起始位点为i。

[0178] 实例:

[0179] 在以下实例中,使用以下SR和SR候选位点执行每个步骤:

[0180] 参考:ACGTCTGATTTACGGACAAACTGATGCA

[0181] SR:ACGTCTGATTGACAAACTGATGCA

[0182] 步骤步骤1:Lo-末端和Hi-末端错配

[0183] 应注意,因为使用24-碱基SR,而非更为典型的100-碱基SR,所以出于这些实例的目的,检测器尺寸和阈值调节到S＝6,U＝1,M＝2:

[0184]
```
参考:          ACGTCTGATTTACGGACAAACTGATGCA
SR:            ACGTCTGATTGACAAACTGATGCA
lo-末端错配:    ......----------------------    lo-末端错配=0
hi-末端错配:    ----------------------!.!!!.    hi-末端错配=4
```

[0186] lo-末端错配为0(ACGTCT与ACGTCT);hi-末端错配为4(GATGCA与AACTGA)。因为hi-末端错配>U并且lo-末端错配<M[(4>1)&&(0<2)],所以将使用lo-末端锚定位点搜索插入缺失。

[0187] 实例步骤2:长度和类型

[0188] 因为已经确定lo-末端锚点,所以可使用hi-末端碱基确定插入缺失的类型和长度。

[0189] 应注意,对于此实例,B＝8和IDIST＝4:

	D=-4 参考：	----ACGTCTGATTTACGGACAAA	
	SR：	ACGTCTGATTGACAAACTGATGCA	
	错配：	----------------.!..!!!.	在 hi-末端处 4 个错配
	D=-3 参考：	---ACGTCTGATTTACGGACAAAC	
	SR：	ACGTCTGATTGACAAACTGATGCA	
	错配：	----------------!!!!!!!!	8 个错配
	D=-2 参考：	--ACGTCTGATTTACGGACAAACT	
	SR：	ACGTCTGATTGACAAACTGATGCA	
	错配：	----------------!!!.!!.!	6 个错配
	D=-1 参考：	-ACGTCTGATTTACGGACAAACTG	
	SR：	ACGTCTGATTGACAAACTGATGCA	
[0190]	错配：	----------------!!!.!!!!	7 个错配
	D=0 参考：	ACGTCTGATTTACGGACAAACTGA	
	SR：	ACGTCTGATTGACAAACTGATGCA	
	错配：	----------------.!!.!!!.	5 个错配
	D=+1 参考：	CGTCTGATTTACGGACAAACTGAT	
	SR：	ACGTCTGATTGACAAACTGATGCA	
	错配：	----------------!!!!..!!	6 个错配
	D = +2 参考：	GTCTGATTTACGGACAAACTGATG	
	SR：	ACGTCTGATTGACAAACTGATGCA	
	错配：	----------------!!!!!!!!	8 个错配
	D = +3 参考：	TCTGATTTACGGACAAACTGATGC	
	SR：	ACGTCTGATTGACAAACTGATGCA	
	错配：	----------------!!!!!!!!	8 个错配
[0191]	D = +4 参考：	CTGATTTACGGACAAACTGATGCA	
	SR：	ACGTCTGATTGACAAACTGATGCA	
	错配：	----------------........	0 错配<--最小值

[0192] 因为D>0,所以已经发现可能的缺失,长度＝+4。

[0193] 实例步骤3:误差和

[0194] 使用上述缺失案例：

[0195]

```
在候选位点处参考：ACGTCTGATTTACGGACAAACTGA
SR：          ACGTCTGATTGACAAACTGATGCA
错配：        .........!..!!..!!.!!!.
计数 1：      00000000001112333455678 8 <-- 错配的运行和
在候选+4 位点处参考：CTGATTTACGGACAAACTGATGCA
SR：          ACGTCTGATTGACAAACTGATGCA
错配：        !!.!!.!.!!.............
计数 2：      12234455677777777777777 7 <-- 错配的运行和
```

[0196] 实例步骤4:插入缺失位点

[0197] 如上所述,因为搜索缺失,所以方法可计算用于标号[1..碱基-长度]的总和[i]。

[0198] 7-计数2[i]:65543322100000000000

[0199] 计数1[i]:00000000001112333455

[0200] 总和[i]:65543322101112333455

[0201] 最小值发生在标号i＝10。此意味着在插入缺失之前碱基数量为10,后跟着4-碱基缺失(来自步骤2),后跟着在插入缺失之后匹配碱基的(碱基-10)。那么简要特质间隙比对报告(Compact Idiosyncratic Gapped Alignment Report,CIGAR)值可为10M4D14M,其中CIGAR串为碱基序列长度和相关联操作,并且用于表示项目,如哪个碱基与参考物比对(匹配/错配),哪个从参考物中缺失,以及哪个是不在参考物中的插入。

[0202] 图9-用于表征并且定位插入缺失的方法

[0203] 图9为示出用于表征并且定位插入缺失的示例性方法的流程图。可以对在图8中示出的方法的结束时发现的每个匹配执行图9的方法。

[0204] 在902,确定用于检索序列的锚定位点。锚定位点可确定为在检索序列的hi-末端或lo-末端处。可通过比较来自hi-末端和lo-末端中的每一个的第一长度位点与参考数据并且选择末端处具有较少错配的锚定位点来确定锚定位点。

[0205] 在904,可使用锚定位点确定插入缺失的长度和类型。举例来说,如果在702确定lo-末端锚定点,则可从在适当的位置具有各种偏移的检索序列的hi-末端计数错配,并且可选择引起最小数量错配的偏移。

[0206] 在906,可计算第一运行误差和。可通过锚定位点计算第一运行误差和。对于lo-末端锚点,运行误差和可从lo-末端锚定位点正向前进。对于hi-末端锚点,运行误差和可从hi-末端锚定位点以反向前进。

[0207] 在908,可计算在偏移处的第二运行误差和。偏移可确定为在704选择的偏移。可通过锚定位点同样计算第二运行误差和。

[0208] 在910,可基于第一和第二运行误差和的总和的最小值确定插入缺失的起始位置。举例来说,可将第一和第二运行和加在一起。构建第一运行误差和,使得将可能不产生对于在锚定位点和插入缺失位置之间检索序列区域的错配。构建第二运行误差和,使得将可能不产生对于在与锚定位点相对的末端和插入缺失的位置之间检索序列区域的错配。通过对第一和第二运行和进行求和,具有最低数量错配的位置可确定为插入缺失的位置。

[0209] 本发明的其它实施例在以下编号段落中呈现：

[0210] 1.一种用于将检索序列的子区段与参考数据匹配的方法，包含：

[0211] 存储衍生自所述参考数据的分级索引表，其中所述分级索引表包含在多个分级层级处的多个条目，

[0212] 基于确定与所述检索序列相关联的在所述分级索引表中的第一条目不为所述索引表的末端条目，增加所述检索序列的所述子区段的长度，和

[0213] 基于确定所述第一条目不为所述分级索引表的末端条目，查找匹配增加长度的所述检索序列的所述子区段的所述分级索引表中的第二条目。

[0214] 2.一种用于评估检索序列与参考数据的匹配的方法，所述方法包含：

[0215] 基于鉴别在所述检索序列和所述参考数据之间的碱基误差，确定在所述检索序列和所述参考数据之间的错配，

[0216] 确定在所述检索中序列的至少一个插入缺失，包含：

[0217] 基于所述检索序列的lo-末端和hi-末端错配的数量，确定在所述检索序列中的锚定位点；

[0218] 使用所述锚定位点，确定所述至少一个插入缺失的长度和类型；

[0219] 确定所述至少一个插入缺失的起始位点，包含：

[0220] 计算在所述检索序列和在所述匹配位置处的所述参考数据之间的第一运行误差和；

[0221] 计算在所述检索序列和在从所述匹配位置的偏移处的所述参考数据之间的第二运行误差和，其中所述偏移基于所述至少一个插入缺失的所述类型和长度；和

[0222] 基于所述第一和第二运行误差和的最小值，确定所述至少一个插入缺失的起始位置。

[0223] 尽管上文已相当详细地描述了实施例，但所属领域的技术人员一旦完全理解以上公开内容便将容易明白众多变化型式及修改。旨在将以下权利要求书解释为包涵所有此类变化型式和修改。

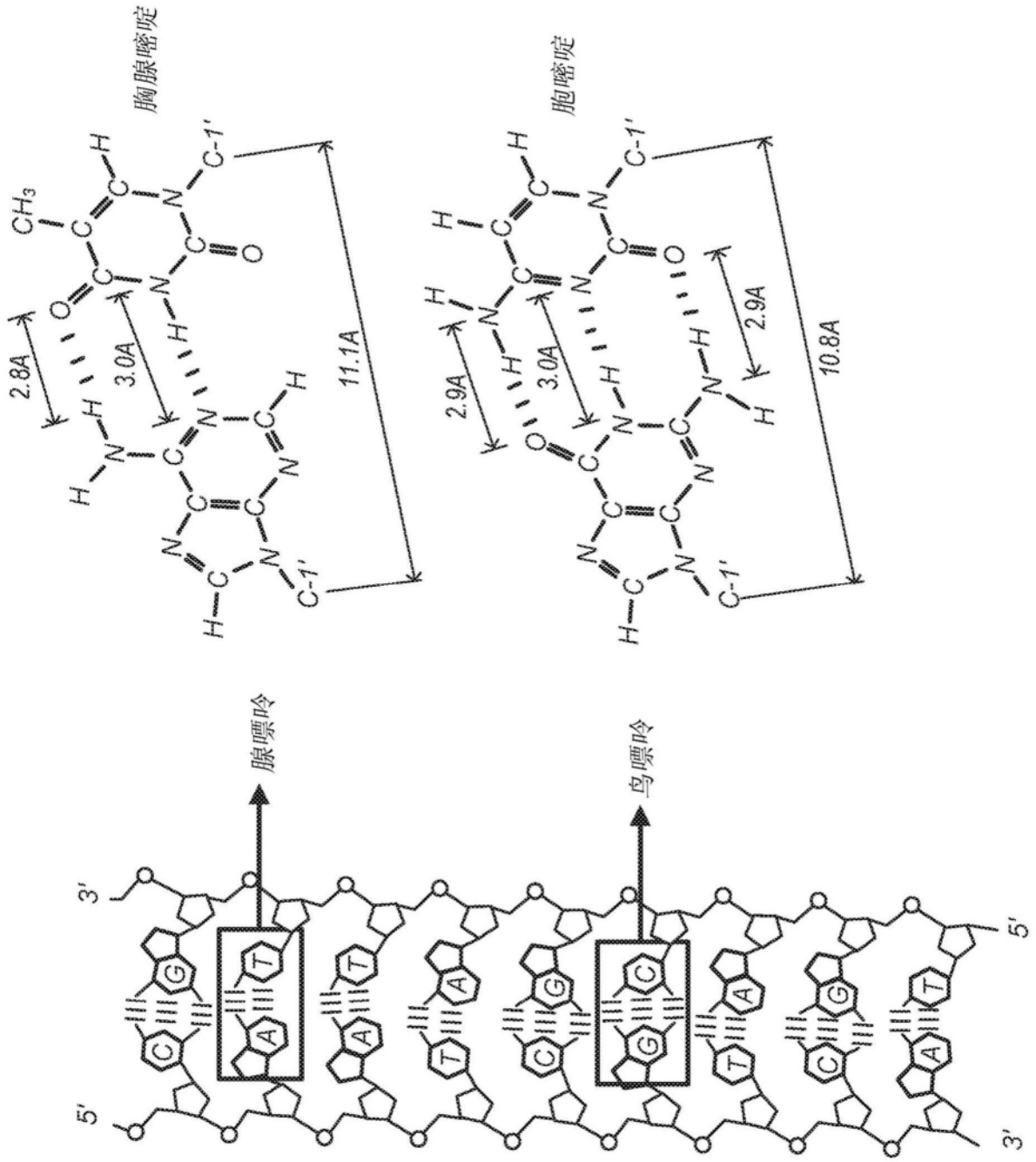


图1

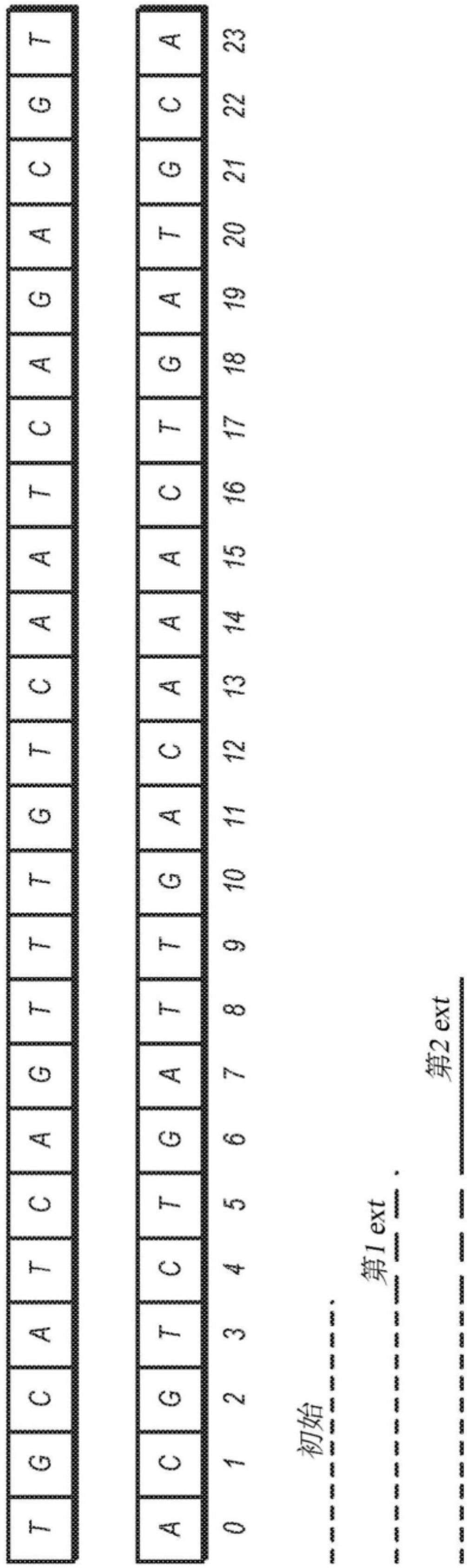

T G C A T C A G T T T G T C A A T C A G A C G T
A C G T C T G A T T G A C A A A C T G A T G C A
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
初始
第1 ext
第2 ext

图2

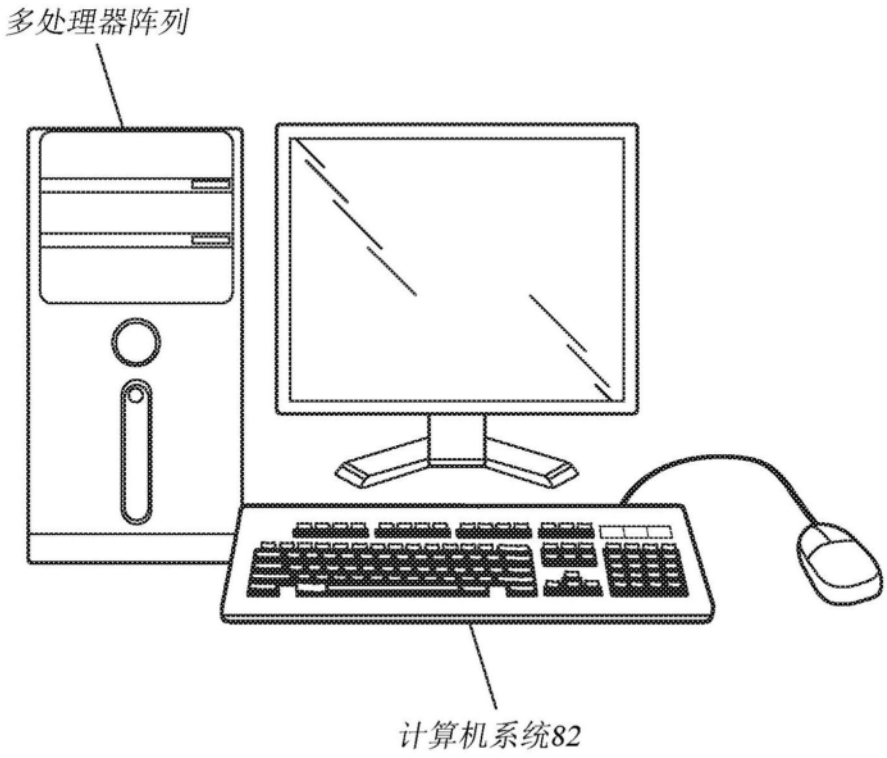


图3

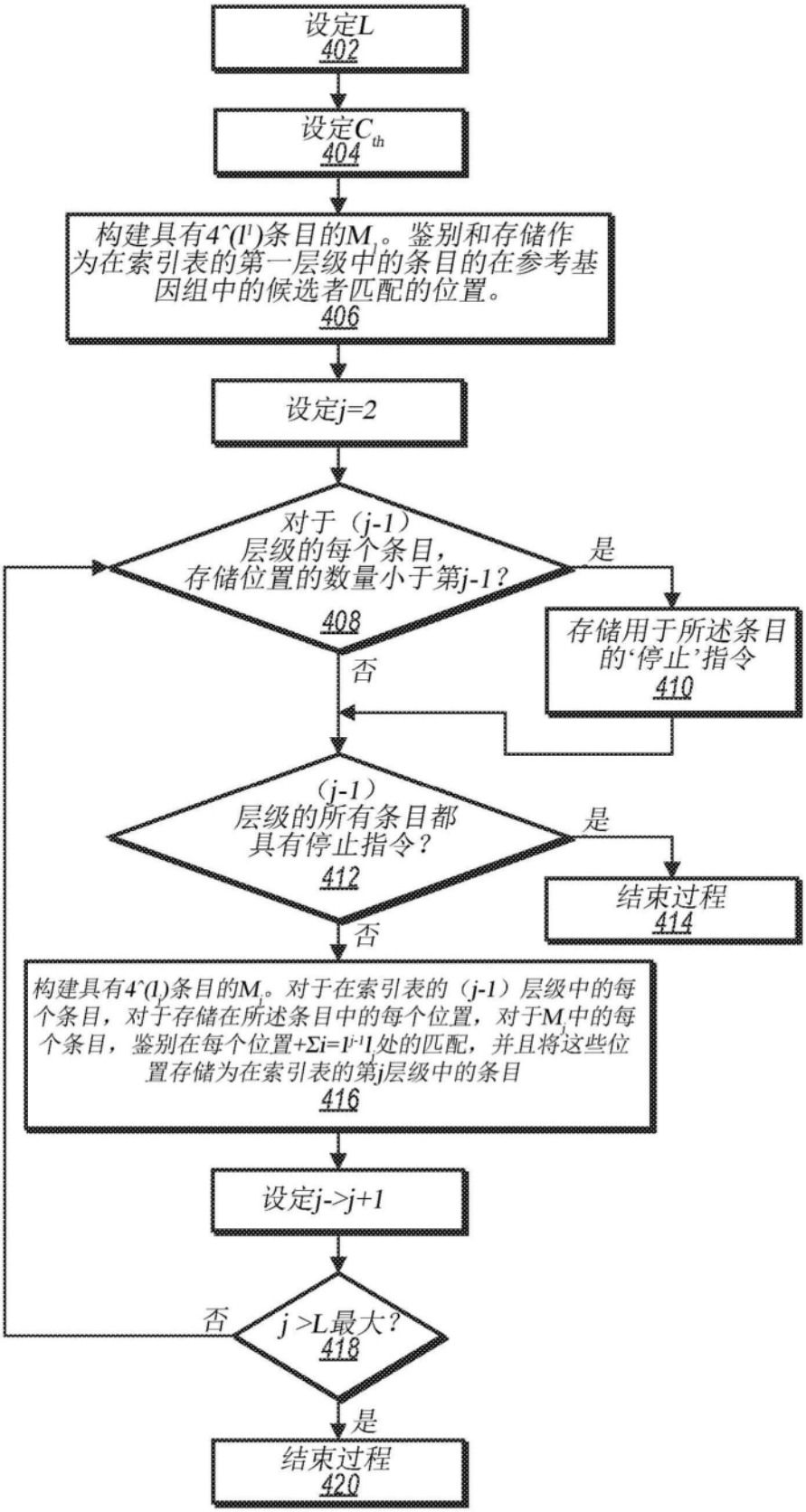

设定L
402
设定C_{th}
404
构建具有4^(l')条目的$M_{l'}$。鉴别和存储作为在索引表的第一层级中的条目的在参考基因组中的候选者匹配的位置。
406
设定j=2
对于（j-1）层级的每个条目，存储位置的数量小于第j-1？
408
是
存储用于所述条目的'停止'指令
410
否
（j-1）层级的所有条目都具有停止指令？
412
是
结束过程
414
否
构建具有4^(l)条目的M_l。对于在索引表的（j-1）层级中的每个条目，对于存储在所述条目中的每个位置，对于$M_{l'}$中的每个条目，鉴别在每个位置+Σi=1^(l'-1)l处的匹配，并且将这些位置存储为在索引表的第j层级中的条目
416
设定j->j+1
否
j >L最大？
418
是
结束过程
420

图4

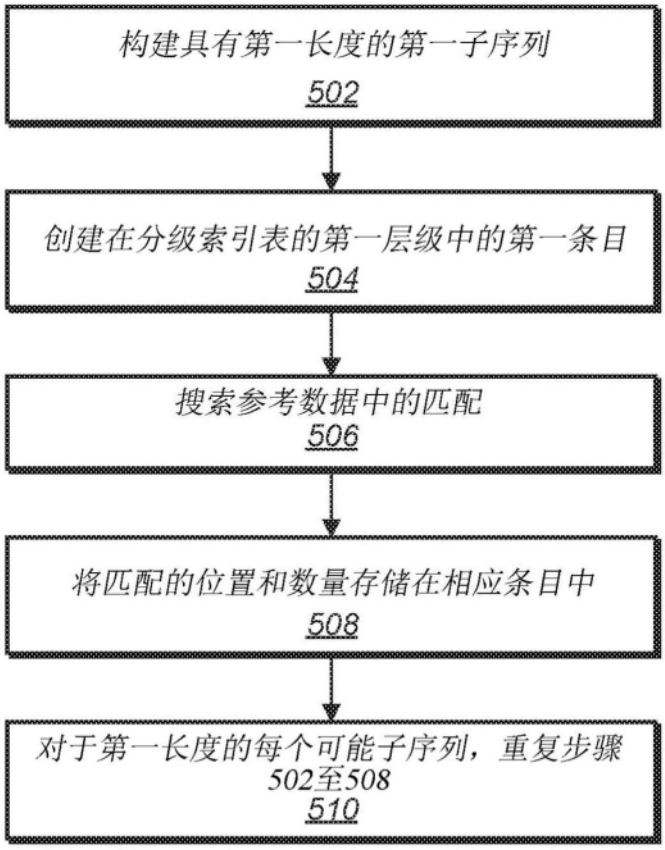

构建具有第一长度的第一子序列
502
创建在分级索引表的第一层级中的第一条目
504
搜索参考数据中的匹配
506
将匹配的位置和数量存储在相应条目中
508
对于第一长度的每个可能子序列，重复步骤502至508
510

图5

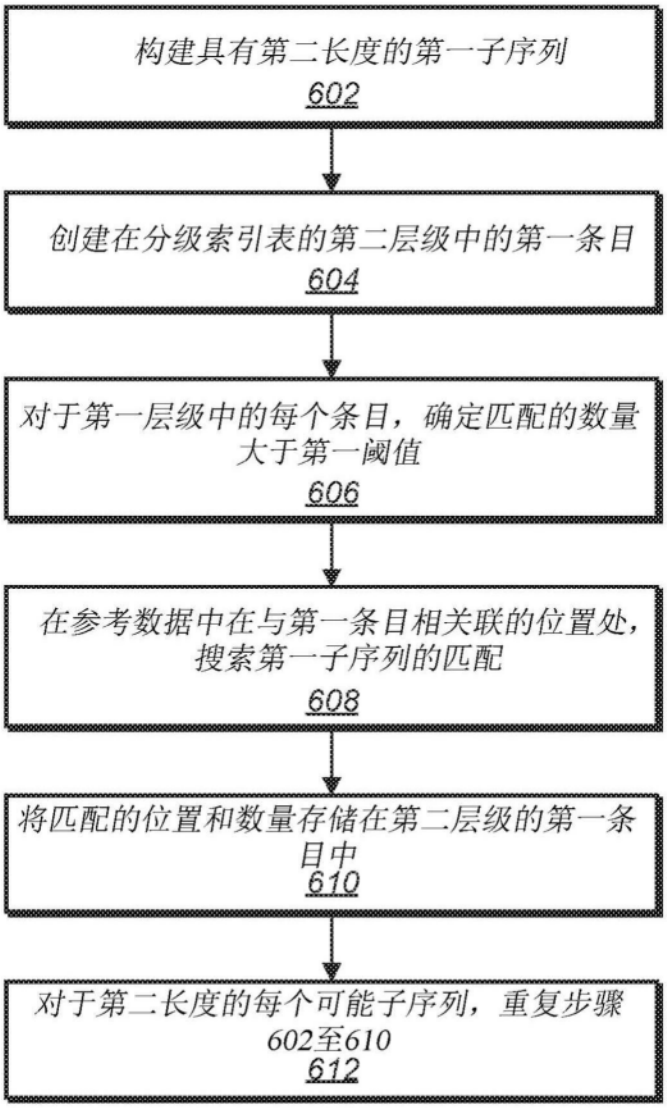

构建具有第二长度的第一子序列
602
创建在分级索引表的第二层级中的第一条目
604
对于第一层级中的每个条目，确定匹配的数量大于第一阈值
606
在参考数据中在与第一条目相关联的位置处，搜索第一子序列的匹配
608
将匹配的位置和数量存储在第二层级的第一条目中
610
对于第二长度的每个可能子序列，重复步骤602至610
612

图6

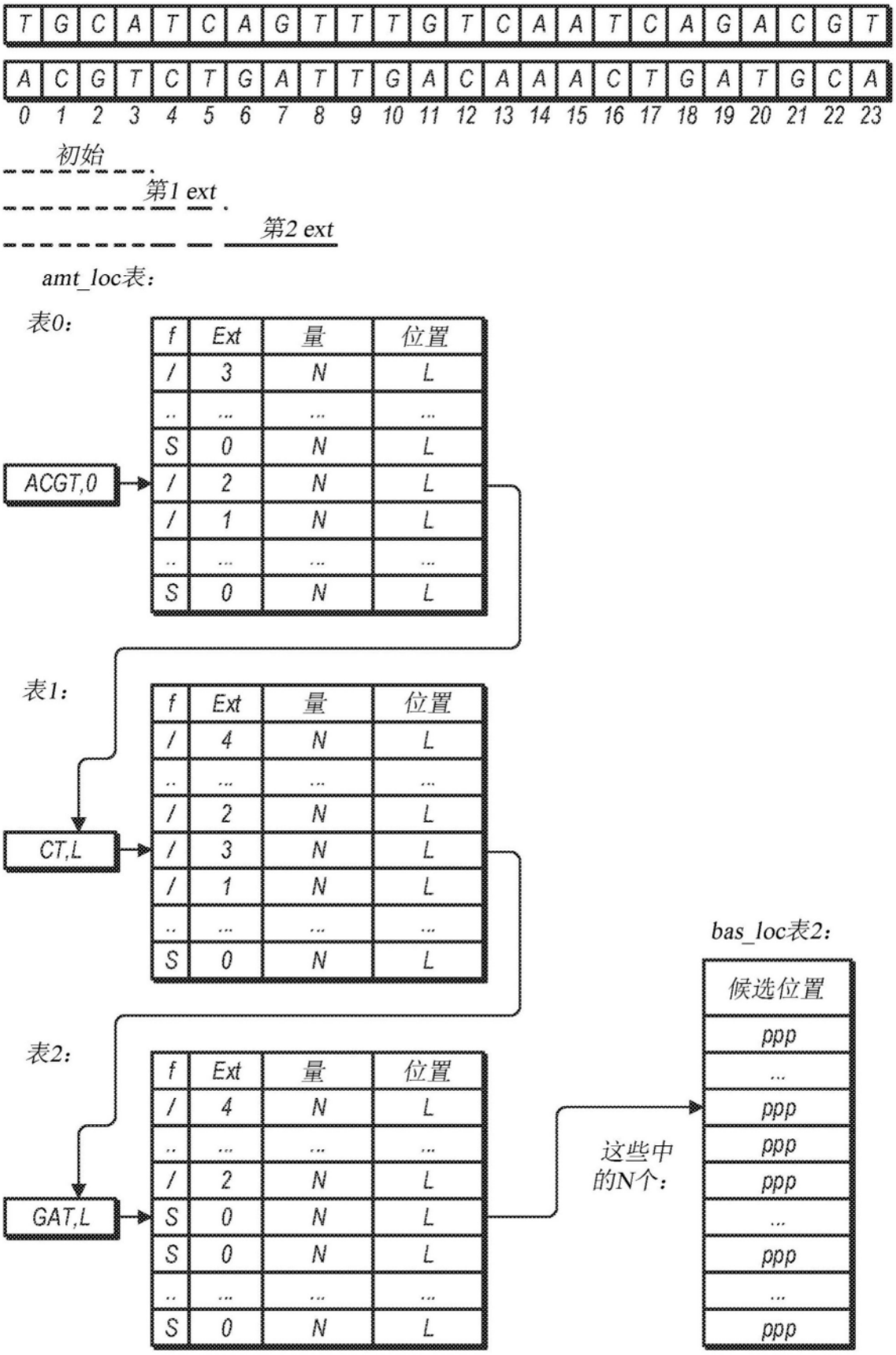

T G C A T C A G T T T G T C A A T C A G A C G T
A C G T C T G A T T G A C A A A C T G A T G C A
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
初始
第1 ext
第2 ext
amt_loc表：
表0：
f Ext 量 位置
/ 3 N L
.. ..
S 0 N L
/ 2 N L
/ 1 N L
.. ..
S 0 N L
ACGT,0
表1：
f Ext 量 位置
/ 4 N L
.. ..
/ 2 N L
/ 3 N L
/ 1 N L
.. ..
S 0 N L
CT,L
表2：
f Ext 量 位置
/ 4 N L
.. ..
/ 2 N L
S 0 N L
S 0 N L
.. ..
S 0 N L
GAT,L
这些中
的N个：
bas_loc表2：
候选位置
ppp
...
ppp
ppp
ppp
...
ppp
...
ppp

图7

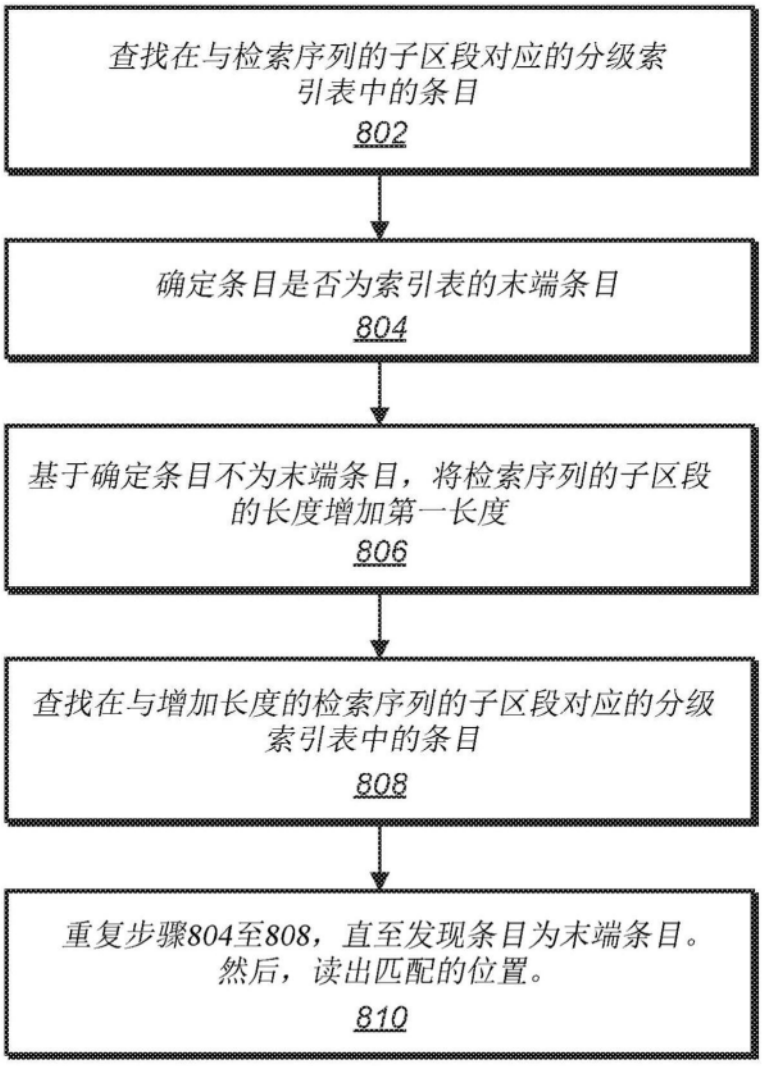

查找在与检索序列的子区段对应的分级索引表中的条目
802
确定条目是否为索引表的末端条目
804
基于确定条目不为末端条目，将检索序列的子区段的长度增加第一长度
806
查找在与增加长度的检索序列的子区段对应的分级索引表中的条目
808
重复步骤804至808，直至发现条目为末端条目。然后，读出匹配的位置。
810

图8

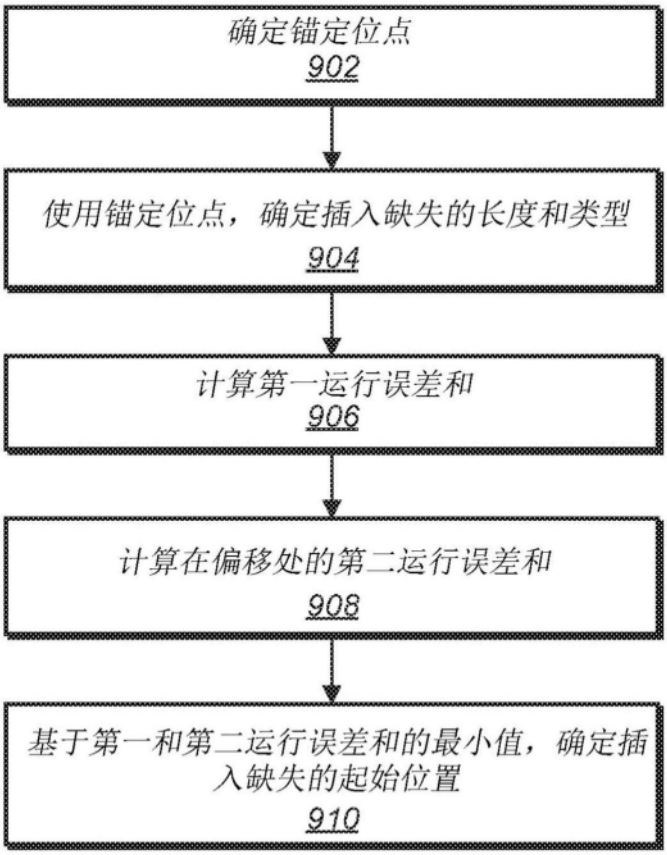

确定锚定位点
902
使用锚定位点，确定插入缺失的长度和类型
904
计算第一运行误差和
906
计算在偏移处的第二运行误差和
908
基于第一和第二运行误差和的最小值，确定插入缺失的起始位置
910

图9