(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 102591930 B
(45)授权公告日 2015.04.29

(21)申请号 201110437649.6
(22)申请日 2011.12.14
(30)优先权数据
12/968,618 2010.12.15 US
(73)专利权人 微软公司
地址 美国华盛顿州
(72)发明人 C·W·拉曼纳 M·H·甘地 J·E·布鲁尔
(74)专利代理机构 上海专利商标事务所有限公司 31100
代理人 钱孟清
(51)Int.Cl.
G06F 17/30(2006.01)
(56)对比文件
CN 101095310 A,2007.12.26, 全文.
CN 1716958 A,2006.01.04, 全文.
US 7689530 B1,2010.03.30, 全文.
US 2010114811 A1,2010.05.06, 全文.
审查员 张伯

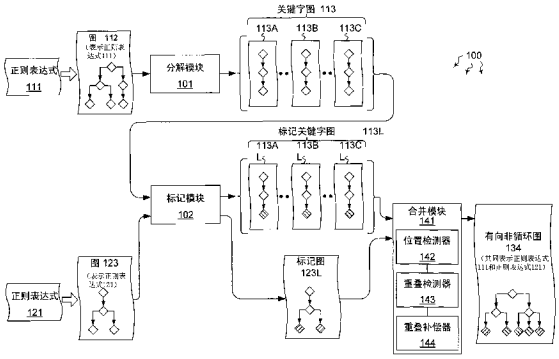
权利要求书3页 说明书8页 附图7页

(54)发明名称

分解和合并正则表达式

(57)摘要

本发明涉及用于分解和合并正则表达式的方法、系统和计算机程序产品。本发明的各个实施例将正则表达式分解成多个简单关键字图、将那些关键字图以紧凑和有效的格式合并，并产生可执行简化的正则表达式字母表的有向非循环图(DAG)。若干这些正则表达式DAG然后能合并在一起以产生表示整个集合的正则表达式的单个DAG。可在多轮方法中组合DAG以及其它文本处理算法和堆集合以扩展正则表达式字母表。



CN 102591930 B

1. 一种在利用正则表达式的计算机环境中减少硬件资源消耗的方法，所述方法在有向非循环图 (134) 中表示一个或多个正则表达式 (111、112)，所述方法包括 ：

访问从第一正则表达式 (111) 分解的一个或多个关键字图 (113) 的动作，一个或多个关键字图的每一个具有根节点、一个或多个中间节点、以及叶节点，所述一个或多个中间节点以及所述叶节点的每一个标识部分地匹配所述第一正则表达式 (111) 的字符模式，所述一个或多个中间节点的每一个以及所述根节点具有单个子节点，所述中间节点之一具有所述叶节点作为子节点，每个叶节点被标记为所述第一正则表达式 (111) 的匹配状态 ；

访问表示第二正则表达式 (121) 的至少一部分的第二图 (123) 的动作，所述第二图 (123) 具有根节点、一个或多个中间节点、以及一个或多个叶节点，所述一个或多个中间节点以及所述一个或多个叶节点的每一个标识部分地匹配所述第二正则表达式 (123) 的字符模式 ；

将所述一个或多个关键字图 (113) 和所述第二图 (123) 合并成有向非循环图 (134) 的动作，所述有向非循环图 (134) 共同表示所述第一正则表达式 (111) 和所述第二正则表达式 (121) 两者，所述合并动作包括对于所述一个或多个关键字图的每一个进行 ：

单独选择所述关键字图 (113A、113B、113C) 的动作 ；

标识所选关键字图 (113A, 113B, 113C) 和第二图 (123) 内具有至少部分重叠字符模式的任何相似定位的中间节点的动作 ；以及

对于相似定位且具有部分重叠字符模式的所选关键字图 (113A, 113B, 113C) 中的任何所标识中间节点和所述第二图 (123) 中的所标识中间节点，在所标识中间节点处合并所选关键字图和所述第二图以在所述有向非循环图 (134) 中表示来自所选关键字图 (113A, 113B, 113C) 和所述第二图 (123) 中的等效匹配状态的动作，所述合并使得所述关键字图变成所述第二图的一部分。

2. 如权利要求 1 所述的方法，其特征在于，标识所选关键字图和第二图内具有至少部分重叠字符模式的任何相似定位的中间节点的动作包括标识完全重叠的所选关键字图中的中间节点和所述第二图中的中间节点的动作。

3. 如权利要求 2 所述的方法，其特征在于，在所标识中间节点处合并所选关键字图和所述第二图的动作包括将所选关键字图中的中间节点与所述第二图中的中间节点组合成表示完全重叠字符模式的单个节点的动作。

4. 如权利要求 1 所述的方法，其特征在于，标识所选关键字图和第二图内具有至少部分重叠字符模式的任何相似定位的中间节点的动作包括标识部分重叠的所选关键字图中的中间节点和所述第二图中的中间节点的动作。

5. 如权利要求 4 所述的方法，其特征在于，在所标识中间节点处合并所选关键字图和所述第二图的动作包括更改所标识中间节点的至少之一的字符模式以消除部分重叠的字符模式的动作。

6. 如权利要求 4 所述的方法，其特征在于，在所标识中间节点处合并所选关键字图和所述第二图的动作包括在所选关键字图与所述第二图之间添加边以对更改所标识中间节点的至少之一的字符模式进行补偿的动作。

7. 如权利要求 1 所述的方法，其特征在于，所述利用正则表达式的计算机环境是用于防垃圾邮件或防数据泄漏的。

8. 一种在利用正则表达式的计算机环境中减少硬件资源消耗的方法，所述方法用来在有向非循环图中表示一个或多个正则表达式，所述方法包括以下动作：

访问从第一正则表达式 (111) 分解的一个或多个关键字图 (113)，一个或多个关键字图的每一个具有根节点、一个或多个中间节点、以及叶节点，所述一个或多个中间节点以及所述叶节点的每一个标识部分地匹配所述第一正则表达式 (111) 的字符模式，所述一个或多个中间节点的每一个以及所述根节点具有单个子节点，所述中间节点之一具有所述叶节点作为子节点，每个叶节点被标记为所述第一正则表达式 (111) 的匹配状态；

访问表示第二正则表达式 (121) 的至少一部分的第二图 (123)，所述第二图 (123) 具有根节点、一个或多个中间节点、以及一个或多个叶节点，所述一个或多个中间节点以及所述一个或多个叶节点的每一个标识部分地匹配所述第二正则表达式 (123) 的字符模式；

将所述一个或多个关键字图 (113) 和所述第二图 (123) 合并成有向非循环图 (134)，所述有向非循环图 (134) 共同表示所述第一正则表达式 (111) 和所述第二正则表达式 (121) 两者，所述合并包括对于所述一个或多个关键字图的每一个进行：

标识所选关键字图 (113A, 113B, 113C) 和第二图 (123) 内具有至少部分重叠字符模式的任何相似定位的中间节点；以及

对于相似定位且具有部分重叠字符模式的所选关键字图 (113A, 113B, 113C) 中的任何所标识中间节点和所述第二图 (123) 中的所标识中间节点，在所标识中间节点处合并所选关键字图和所述第二图以表示有向非循环图 (134) 中来自所选关键字图 (113A, 113B, 113C) 和所述第二图 (123) 中的等效匹配状态，所述合并使得所述关键字图变成所述第二图的一部分。

9. 如权利要求 8 所述的方法，其特征在于，还包括将所述一个或多个关键字图的每一个的叶节点标记为所述第一正则表达式的匹配状态。

10. 如权利要求 8 所述的方法，其特征在于，还包括将所述第二图的每个端节点标记为所述第二正则表达式的匹配状态。

11. 如权利要求 8 所述的方法，其特征在于，所述利用正则表达式的计算机环境是用于防垃圾邮件或防数据泄漏的。

12. 一种在利用正则表达式的计算机环境中减少硬件资源消耗的方法，所述方法在有向非循环图中表示一个或多个正则表达式，所述方法包括：

访问从第一正则表达式分解的一个或多个关键字图 (301A, 301B)(501) 的动作，一个或多个关键字图的每一个具有根节点、一个或多个中间节点、以及叶节点，所述一个或多个中间节点以及所述叶节点的每一个标识部分地匹配所述第一正则表达式的字符模式，所述一个或多个中间节点的每一个以及所述根节点具有单个子节点，所述中间节点之一具有所述叶节点作为子节点，每个叶节点被标记为所述第一正则表达式的匹配状态；

访问表示第二正则表达式的第二图 (302)(502) 的动作，所述第二图具有根节点、一个或多个中间节点、以及一个或多个叶节点，所述一个或多个中间节点以及所述一个或多个叶节点的每一个标识部分地匹配第二正则表达式的字符模式，所述第二图具有被标记为所述第二正则表达式的匹配状态的一个或多个端节点；以及

将一个或多个关键字图 (301A, 301B)(501) 和第二图 (302)(502) 合并成有向非循环图 (304) 的动作，所述有向非循环图共同表示所述第一正则表达式和所述第二正则表达式两

者，所述合并动作包括：

在所述一个或多个关键词图 (312)(511) 和所述第二图 (313)(512) 内标识具有至少部分重叠的字符模式的任何相似定位的中间节点的动作；

对于相似定位且具有部分重叠字符模式的关键字图 (511) 中的任何所标识中间节点和所述第二图 (512) 中的所标识中间节点，进行：

更改所标识中间节点 (511) 的至少之一的字符模式以消除部分重叠的字符模式的动作；以及

在所述关键字图 (501) 和所述第二图 (502) 之间添加边 (514) 以对更改所标识中间节点 (511) 的至少之一的字符模式进行补偿的动作；

对于相似定位且具有完全重叠字符模式的关键字图 (301B) 中的任何所标识中间节点和所述第二图 (302) 中的所标识中间节点，进行：

通过将所述关键字图 (301B) 中的中间节点 (312) 与所述第二图 (302) 中的中间节点 (313) 组合成表示完全重叠的字符模式的单个节点 (314) 来将所述关键字图和所述第二图组合在一起的动作。

13. 如权利要求 12 所述的方法，其特征在于，所述利用正则表达式的计算机环境是用于防垃圾邮件或防数据泄漏的。

14. 一种在利用正则表达式的计算机环境中减少硬件资源消耗的系统，所述系统用来在有向非循环图中表示一个或多个正则表达式，所述系统包括：

用于访问从第一正则表达式 (111) 分解的一个或多个关键字图 (113) 的装置，一个或多个关键字图的每一个具有根节点、一个或多个中间节点、以及叶节点，所述一个或多个中间节点以及所述叶节点的每一个标识部分地匹配所述第一正则表达式 (111) 的字符模式，所述一个或多个中间节点的每一个以及所述根节点具有单个子节点，所述中间节点之一具有所述叶节点作为子节点，每个叶节点被标记为所述第一正则表达式 (111) 的匹配状态；

用于访问表示第二正则表达式 (121) 的至少一部分的第二图 (123) 的装置，所述第二图 (123) 具有根节点、一个或多个中间节点、以及一个或多个叶节点，所述一个或多个中间节点以及所述一个或多个叶节点的每一个标识部分地匹配所述第二正则表达式 (123) 的字符模式；

用于将所述一个或多个关键字图 (113) 和所述第二图 (123) 合并成有向非循环图 (134) 的装置，所述有向非循环图 (134) 共同表示所述第一正则表达式 (111) 和所述第二正则表达式 (121) 两者，所述合并包括对于所述一个或多个关键字图的每一个进行：

标识所选关键字图 (113A, 113B, 113C) 和第二图 (123) 内具有至少部分重叠字符模式的任何相似定位的中间节点；以及

对于相似定位且具有部分重叠字符模式的所选关键字图 (113A, 113B, 113C) 中的任何所标识中间节点和所述第二图 (123) 中的所标识中间节点，在所标识中间节点处合并所选关键字图和所述第二图以表示有向非循环图 (134) 中来自所选关键字图 (113A, 113B, 113C) 和所述第二图 (123) 中的等效匹配状态，所述合并使得所述关键字图变成所述第二图的一部分。

15. 如权利要求 14 所述的系统，其特征在于，所述利用正则表达式的计算机环境是用于防垃圾邮件或防数据泄漏的。

分解和合并正则表达式

技术领域

[0001] 本发明涉及用于分解和合并正则表达式的方法、系统和计算机程序产品。

背景技术

[0002] 计算机系统和相关技术影响社会的许多方面。的确，计算机系统处理信息的能力已转变了人们生活和工作的方式。计算机系统现在通常执行在计算机系统出现以前手动执行的许多任务（例如，文字处理、日程安排和会计等）。最近，计算机系统彼此耦合并耦合到其他电子设备以形成计算机系统和其他电子设备可以在其上传输电子数据的有线和无线计算机网络。因此，许多计算任务的执行跨多个不同的计算机系统和 / 或多个不同的计算环境分布。

[0003] 在一些计算环境中，正则表达式用来匹配文本串，诸如举例而言特定字符、词、或字符模式。正则表达式可用能通过正则表达式处理器解释的形式语言来编写。正则表达式处理器是用作解析器发生器或检查文本并标识与所提供规范相匹配的部分的程序。

[0004] 正则表达式由许多文本编辑器、实用程序和编程语言使用以基于模式来搜索和操纵文本。例如，防垃圾邮件服务可利用正则表达式来确定电子消息中是否包含已知指示 SPAM 的文本串。类似地，防数据泄漏服务可利用正则表达式来检测并防止机密信息的未获授权使用和传输。

[0005] 在利用正则表达式的环境中，顺序地执行大量正则表达式并非是不常见的。例如，在确定电子消息是否包含 SPAM 时，防垃圾邮件服务可使用数以万计的正则表达式。正则表达式集合中的正则表达式可针对每个接收到的电子消息来顺序地运行。正则表达式的顺序执行限制可缩放性，并且随着检查出匹配的正则表达式和 / 或文本部分的数量增多能消耗相当多的资源。

发明内容

[0006] 本发明涉及用于分解和合并正则表达式的方法、系统和计算机程序产品。访问一个或多个关键字图。该一个或多个关键字图从第一正则表达式分解。该一个或多个关键字图的每一个具有根节点、一个或多个中间节点、以及叶节点。该一个或多个中间节点以及叶节点的每一个标识与第一正则表达式部分地相匹配的字符模式。该一个或多个中间节点的每一个和根节点具有单个子节点。中间节点之一具有叶节点作为子节点。每个叶节点被标记为第一正则表达式的匹配状态。

[0007] 访问第二图。第二图表示第二正则表达式。该第二图具有根节点、一个或多个中间节点、以及一个或多个叶节点。该一个或多个中间节点以及一个或多个叶节点的每一个标识与第二正则表达式部分地相匹配的字符模式。该第二图具有标记为第二正则表达式的匹配状态的一个或多个端节点。

[0008] 该一个或多个关键字图和第二图被合并成有向非循环图，该有向非循环图共同表示第一正则表达式和第二正则表达式二者。合并包括在一个或多个关键字图和第二图内标

识具有至少部分重叠的字符模式的任何相似定位的中间节点。对于任何所标识的具有部分重叠的字符模式的中间节点，所标识的中间节点的至少之一的字符模式被更改以消除部分重叠的字符模式。边被添加在关键字图和第二图之间以对更改所标识中间节点的至少之一的字符模式作出补偿。对于任何所标识的具有完全重叠的字符模式的中间节点，关键字图中的中间节点和第二图中的中间节点被组合成表示完全重叠的字符模式的单个节点。

[0009] 提供本发明内容以便以简化的形式介绍将在以下的具体实施方式中进一步描述的一些概念。本发明内容并非旨在标识所要求保护主题的关键特征或必要特征，也不旨在用于帮助确定所要求保护主题的范围。

[0010] 本发明的附加特征和优点将在以下描述中叙述，且其一部分根据本说明书将是显而易见的，或可通过对本发明的实践来获知。本发明的特征和优点可通过在所附权利要求书中特别指出的工具和组合来实现和获得。本发明的这些和其他特征将通过以下描述和所附权利要求书变得更加显而易见，或可通过对下文中所述的本发明的实践来领会。

附图说明

[0011] 为了描述可获得本发明的上述和其它优点和特征的方式，将通过引用附图中示出的本发明的具体实施例来呈现以上简要描述的本发明的更具体描述。可以理解，这些附图仅描述本发明的典型实施例，从而不被认为是对其范围的限制，本发明将通过使用附图用附加特征和细节来描述和说明，在附图中：

[0012] 图 1 示出便于分解和合并正则表达式的示例计算机体系结构。

[0013] 图 2 示出分解表示正则表达式的图的示例。

[0014] 图 3 示出合并表示不同正则表达式的图的示例。

[0015] 图 4 示出分解表示正则表达式的图的另一示例。

[0016] 图 5 示出合并表示不同正则表达式的图的另一示例。

[0017] 图 6 示出用于分解和合并正则表达式的示例方法的流程图。

具体实施方式

[0018] 本发明涉及用于分解和合并正则表达式的方法、系统和计算机程序产品。访问一个或多个关键字图。该一个或多个关键字图从第一正则表达式分解。该一个或多个关键字图的每一个具有根节点、一个或多个中间节点、以及叶节点。该一个或多个中间节点以及叶节点的每一个标识与第一正则表达式部分地相匹配的字符模式。该一个或多个中间节点的每一个和根节点具有单个子节点。中间节点之一具有叶节点作为子节点。每个叶节点被标记为第一正则表达式的匹配状态。

[0019] 访问第二图。第二图表示第二正则表达式。该第二图具有根节点、一个或多个中间节点、以及一个或多个叶节点。该一个或多个中间节点以及一个或多个叶节点的每一个标识与第二正则表达式部分地相匹配的字符模式。该第二图具有标记为第二正则表达式的匹配状态的一个或多个端节点。

[0020] 该一个或多个关键字图和第二图被合并成有向非循环图，该有向非循环图共同表示第一正则表达式和第二正则表达式二者。合并包括在一个或多个关键字图和第二图内标识具有至少部分重叠的字符模式的任何相似定位的中间节点。对于任何所标识的具有部分

重叠的字符模式的中间节点，所标识的中间节点的至少之一的字符模式被更改以消除部分重叠的字符模式。边被添加在关键字图和第二图之间以对更改所标识中间节点的至少之一的字符模式作出补偿。对于任何所标识的具有完全重叠的字符模式的中间节点，关键字图中的中间节点和第二图中的中间节点被组合成表示完全重叠的字符模式的单个节点。

[0021] 本发明的各实施例可包括或利用专用或通用计算机，该专用或通用计算机包括诸如例如一个或多个处理器和系统存储器的计算机硬件，如以下更详细讨论的。本发明范围内的各实施例还包括用于携带或存储计算机可执行指令和 / 或数据结构的物理介质及其他计算机可读介质。这些计算机可读介质可以是通用或专用计算机系统可访问的任何可用介质。存储计算机可执行指令的计算机可读介质是计算机存储介质（设备）。携带计算机可执行指令的计算机可读介质是传输介质。由此，作为示例而非限制，本发明的各实施例可包括至少两种完全不同类型的计算机可读介质 :计算机存储介质（设备）和传输介质。

[0022] 计算机存储介质（设备）包括 RAM、ROM、EEPROM、CD-ROM 或其他光盘存储、磁盘存储或其他磁存储设备、或可用于存储计算机可执行指令或数据结构形式的所需程序代码装置的且可由通用或专用计算机访问的任何其他介质。

[0023] “网络”被定义为允许在计算机系统和 / 或模块和 / 或其他电子设备之间传输电子数据的一个或多个数据链路。当信息通过网络或另一通信连接（硬连线、无线、或者硬连线或无线的组合）传送到或提供给计算机时，该计算机将该连接适当地视为传输介质。传输介质可包括可用于携带计算机可执行指令或数据结构形式的所需程序代码装置且通用或专用计算机可访问的网络和 / 或数据链路。上述的组合也应当被包括在计算机可读介质的范围内。

[0024] 此外，在到达各种计算机系统组件之后，计算机可执行指令或数据结构形式的程序代码装置可从传输介质自动传输到计算机存储介质（设备）（或反之亦然）。例如，通过网络或数据链接接收到的计算机可执行指令或数据结构可被缓存在网络接口模块（例如，“NIC”）内的 RAM 中，然后最终被传输到计算机系统 RAM 和 / 或计算机系统处的较不易失性的计算机存储介质（设备）。因而，应当理解，计算机存储介质（设备）可被包括在还利用（甚至主要利用）传输介质的计算机系统组件中。

[0025] 计算机可执行指令例如包括，当在处理器处执行时使通用计算机、专用计算机、或专用处理设备执行某一功能或某组功能的指令和数据。计算机可执行指令可以是例如二进制代码、诸如汇编语言之类的中间格式指令、或甚至源代码。虽然用结构特征和 / 或方法动作专用的语言描述了本主题，但是应当理解，所附权利要求书中定义的主题不必限于上述特征或动作。相反，所述特征和动作是作为实现权利要求的示例形式而公开的。

[0026] 本领域的技术人员将理解，本发明可在具有许多类型的计算机系统配置的网络计算环境中实践，这些计算机系统配置包括个人计算机、台式计算机、膝上型计算机、消息处理器、手持式设备、多处理器系统、基于微处理器的或可编程消费电子设备、网络 PC、小型计算机、大型计算机、移动电话、PDA、寻呼机、路由器、交换机等。本发明也可在其中通过网络链接（或者通过硬连线数据链路、无线数据链路，或者通过硬连线和无线数据链路的组合）的本地和远程计算机系统两者都执行任务的分布式系统环境中实施。在分布式系统环境中，程序模块可位于本地和远程存储器存储设备两者中。

[0027] 在本说明书和所附权利要求中，“正则表达式”是用来匹配文本串，诸如举例而言

特定字符、词或字符模式的结构。在一些实施例中，正则表达式具有有限字母表。正则表达式可用能通过正则表达式处理器解释的形式语言来编写。正则表达式处理器用作解析器发生器，或检查文本并标识与所提供正则表达式相匹配的文本部分。

[0028] 一般而言，图可用来表示正则表达式及其匹配状态。例如，暂时参看图 2，图 201 表示正则表达式“(\d\d)|(a(b|c))”。类似地，暂时参看图 4，图 401 表示正则表达式“([a，b，c]x)|(\d(cd|[1,3,5]([a，c，d]|ea)))”。图可通过用输入文本执行状态机来“运行”，这允许多个图的并行化。

[0029] 图 1 示出便于分解和合并正则表达式的示例计算机体系结构 100。参考图 1，计算机体系结构 100 包括分解模块 101、标记模块 102、以及合并模块 141。所描绘的组件中的每一个可通过诸如举例而言局域网（“LAN”）、广域网（“WAN”）和甚至因特网等网络（或作为网络的一部分）彼此连接。因此，所描绘的组件中的每一个以及任何其他连接的计算机系统及其组件都可创建消息相关数据并通过网络交换与消息相关数据（例如，网际协议（“IP”）数据报和利用 IP 数据报的其他更高层协议，诸如传输控制协议（“TCP”）、超文本传输协议（“HTTP”）、简单邮件传输协议（“SMTP”）等）。

[0030] 一般而言，分解可用来从表示正则表达式的更复杂的图产生表示正则表达式的一组简单图。因此，分解模块 101 被配置成将诸如举例而言表示正则表达式的图的图分解成对应的多个关键字图。分解模块 101 实质上能去除更复杂的正则表达式的转折部分，以将该更复杂的正则表达式分成多个更简单的正则表达式。每个关键字图的叶节点表示来自更复杂图的端条件（在更复杂图中其可以在中间节点或叶节点处）。分解模块 101 可分解标记或非标记图。

[0031] 标记模块 102 被配置为标记图或关键字图的节点以指示所表示正则表达式的匹配状态。标记模块 102 可在分解之前或之后标记节点。

[0032] 再次参看图 2，图 2 示出分解表示正则表达式的图的示例。如图所示，分解模块 101 接收图 201 作为输入。图 201 先前被标记（由对角阴影线表示）以指示正则表达式“(\d\d)|(a(b|c))”的匹配状态。分解模块 101 分解图 201 并输出关键字图 202。图 201 中的标记被携带至关键字图 202。由此，当文本与图 201 或任一个关键字图 202 作比较（碰上）时，任一匹配被指示为与“(\d\d)|(a(b|c))”的匹配。

[0033] 再次参看图 4，图 4 示出分解表示正则表达式的图的另一示例。如图所示，分解模块 101 接收图 401 作为输入。图 401 先前被标记（由对角阴影线表示）以指示正则表达式“([a，b，c]x)|(\d(cd|[1,3,5]([a，c，d]|ea)))”的匹配状态。分解模块 101 分解图 401 并输出关键字图 402。图 401 中的标记被携带至关键字图 402。由此，当文本与图 401 或任一个关键字图 402 作比较（碰上）时，任一匹配被指示为与“([a,b,c]x)|(\d(cd|[1,3,5]([a，c，d]|ea)))”的匹配。

[0034] 在一些实施例中，根据以下算法将图分解成关键字图：

[0035] 在根节点处开始。

[0036] 标识该根节点的所有子节点。

[0037] 对于这些节点的每一个：

[0038] a. 复制该节点之上的父节点（称此为“prefix.i”（前缀.i））。

[0039] b. 添加此节点及其子树作为“prefix.i”的子。

[0040] c. 再次从 (2) 开始，但是使用当前节点作为根节点。

[0041] 该算法能产生表示该图的关键字图（例如 DAG）集合。每个关键字图具有作为叶节点的单个端节点。在每个图内，每个节点具有单个子节点。

[0042] 一般而言，可使用合并来产生表示正则表达式的集合的单个有向非循环图（“DAG”）。相应地，合并模块 101 被配置成接收两个图作为输入，并将这两个图合并成共同表示两个输入图的匹配状态的单个 DAG。为了消除处理冗余，合并模块 101 可将两个输入图中相似定位节点处的重叠字符模式组合成单个 DAG 中的单个节点。当字符模式部分重叠时，合并模块 101 可更改一个输入图中一节点处的字符模式。合并模块 101 然后可通过在该节点与另一输入图中对应节点之间添加附加边来进行补偿。添加附加边便于两个输入图与单个 DAG 之间匹配状态的等价。

[0043] 在一些实施例中，合并模块 141 将两个关键字图合并成单个 DAG。在其它实施例中，合并模块 141 将关键字图和另一图合并成单个 DAG。合并模块 141 的功能可按需重新使用以将更大集合的图合并在一起。

[0044] 参看图 3，合并模块 141 将关键字图 301（例如先前从另一图分解的）和图 302 合并成有向非循环图 304。合并模块 141 将图 302 和关键字图 301A 用作输入。合并模块 141 将图 302 和关键字图 301A 合并成中间图 303。随后，合并模块 141 利用中间图 330 和关键字图 301B。合并模块 141 将中间图 303 和关键字图 301B 合并成有向非循环图 304。因为字符模式节点 312 和 313 重叠，所以节点 312 和 313 合并成有向非循环图 304 中的单个节点 314。

[0045] 标记（如不同对角阴影线所指示）在整个合并过程中维持。因而，端节点指示相匹配的正则表达式。节点 316 和 317 指示与正则表达式“\d\d|um”（它们从中分解的正则表达式）的匹配，而节点 318 指示与正则表达式“un”的匹配。

[0046] 如图 3 所示，合并模块 141 的输入是外部的。在其它实施例中，合并模块 141 接收一组图作为输入并输出 DAG。在处理期间，中间图在合并模块 141 内保持并进行内部处理。

[0047] 如图所示，合并模块 141 包括位置检测器 142、重叠检测器 143 以及重叠补偿器 144。在合并位置期间，位置检测器 142 被配置成标识不同图内的相似定位节点。相似定位节点可基于离根节点的距离来标识。例如，在图 3 中，节点 312 和 313 被相似地定位。在合并期间，重叠检测器 143 被配置成检测不同节点的字符模式是否至少部分地重叠。例如，字符模式 [1,3,5] 部分地匹配字符模式 \d。另一方面，字符模式 [a,b,c] 和字符模式 [a,b,c] 完全重叠。在合并期间，重叠补偿器 144 被配置成在具有部分重叠字符模式的节点被合并成单个节点时进行补偿。补偿可包括在正在被合并的输入图之间添加边。附加边便于输入图的匹配状态与所得 DAG 的匹配状态之间的等价。

[0048] 图 5 示出合并表示不同正则表达式的图的另一示例。关键字图 501 和图 502 可作为输入来接收（例如在合并模块 141 处）。位置检测器 142 能检测节点 511 和节点 512 分别在关键字图 501 和图 502 中相似地定位。重叠检测器 143 能标识部分重叠的模式 503（或公共边）。即，字符模式 \d 与字符模式 [2,3] 部分地重叠。重叠补偿器 144 可通过将节点 511 的字符模式更改为“\d-[2,3]”来去除部分重叠（去除公共边）。重叠补偿器还可添加从节点 512 至节点 513 的边 514。合并模块 114 然后能组合根节点以将（经更改的）关键字图 501 添加至图 502。重叠补偿允许图进行合并，但仍表示等效的匹配状态。例如，即使

在节点 512 处作比较（且绕过节点 511），文本串“2cd”也仍然匹配关键字图 501。

[0049] 如图所示，端节点内的不同阴影线分别指示关键字图 501 和图 502 的匹配状态。

[0050] 在一些实施例中，图根据以下算法来合并：

[0051] 仅用根节点创建空 DAG。将此标记为 Final.DAG（最后 .DAG）。

[0052] 对于集合中的每个 DAG(i.DAG)，进行以下操作：

[0053] a. 将 i.node(i. 节点）设置为 i.DAG 的根节点。

[0054] b. 将 final.node（最后 . 节点）设置为 Final.DAG（最后 .DAG）的根节点。

[0055] c. 只要 final.node 具有完全相同的边，就遍历 i.node 和 final.node 迭代。

[0056] d. 如果 i.node 边是 final.node 边的超集，则：

[0057] i. 在 i.node 与 final.node 之间添加表示非公共字符的边。此边指向 i.node 的子。

[0058] ii. 对于每个公共（边，节点）

[0059] 1. 只要 final.node 和 i.node 具有完全相同的边，就沿 final.node 和 i.node 迭代。

[0060] 2. 如果到达端节点，则将其标记为 i.DAG 的端节点。

[0061] 3. 如果未到达，则添加从 final.node 到 i.node 的子的边。

[0062] e. 如果 final.node 边是 i.node 边的超集，则：

[0063] i. 在 i.node 与 final.node 之间添加表示非公共字符的边。此边指向 final.node 的子。

[0064] ii 对于每个公共（边，节点）

[0065] 1. 只要 final.node 和 i.node 具有完全相同的边，就沿 final.node 和 i.node 迭代。

[0066] 2. 如果到达端节点，则将其标记为 final.DAG 的端节点。

[0067] 3. 如果未到达，则添加从 i.node 到 final.node 的子的边。

[0068] 图 6 示出用于分解和合并正则表达式的示例方法 600 的流程图。方法 600 将相关于计算机体系结构 100 的组件和数据并部分参考图 3 和 5 来描述。

[0069] 方法 600 包括访问表示第一正则表达式的图的动作（动作 601）。例如，分解模块 101 可访问表示正则表达式 111 的图 112。方法 600 包括将图分解成一个或多个关键字图的动作（动作 602），一个或多个关键字图的每一个具有根节点、一个或多个中间节点、以及叶节点，一个或多个中间节点以及叶节点的每一个标识部分地匹配第一正则表达式的字符模式，一个或多个中间节点的每一个以及根节点具有单个子节点，中间节点之一具有叶节点作为子节点。例如，分解模块 101 可将图 112 分解成关键字图 113（例如 113A、113B、113C 等）。

[0070] 方法 600 包括将一个或多个关键字图的每一个的叶节点标记为第一正则表达式的匹配状态的动作（动作 603）。例如，标记模块 102 可标记关键字图 113 的叶节点以产生标记关键字图 113AL、113BL、113BL 等。

[0071] 方法 600 包括访问表示第二正则表达式的第二图的动作（动作 604），该第二图具有根节点、一个或多个中间节点、以及一个或多个叶节点，一个或多个中间节点以及一个或多个叶节点的每一个标识部分地匹配第二正则表达式的字符模式。例如，标记模块 102 可访问表示正则表达式 121 的图 123。方法 600 包括将第二图中的一个或多个端节点标记为第二正则表达式的匹配状态的动作（动作 605）。例如，标记模块 102 可标记图 123 的端节

点以生成标记图 123L。

[0072] 方法 600 包括将一个或多个关键字图和第二图合并成有向非循环图的动作（动作 606），该有向非循环图共同表示第一正则表达式和第二正则表达式两者。例如，合并模块 141 可将标记关键字图 113L 和标记图 123L 合并成有向非循环图 134。有向非循环图 134 共同表示正则表达式 111 和正则表达式 121。

[0073] 动作 606 包括在一个或多个关键字图和第二图内标识具有至少部分重叠的字符模式的任何相似定位的中间节点的动作（动作 607）。例如，位置检测器 142 可标识一个或多个标记关键字图 113L 和标记图 123L 中的相似定位中间节点。相似定位节点可以是与其根节点等距的节点。例如，参照图 3，节点 312 和 313 是相似定位的（两者均距其相应根节点一条边）。类似地，在图 5 中，节点 511 和 512 被相似地定位。在图 5 中，节点 513 和 514 也被相似定位。

[0074] 在相似定位的中间节点中，重叠检测器 143 可检测何时节点具有至少部分重叠的字符模式。在图 3 中，节点 312 和 313 完全重叠。在图 5 中，节点 511 和 512 部分重叠，而节点 513 和 514 不重叠。

[0075] 对于相似定位且具有部分重叠字符模式的关键字图中的任何所标识中间节点和第二图中的所标识中间节点，动作 606 包括更改所标识中间节点的至少之一的字符模式以消除部分重叠的字符模式的动作（动作 608）。例如，重叠补偿器 144 可更改中间节点处的字符模式以消除与另一节点的部分重叠。参照图 5，节点 511 处的字符模式“\d”可被更改成“\d-[2,3]”（其等效于 [0,1,4,5,6,7,8,9]）以消除与节点 512 的部分重叠。

[0076] 对于相似定位且具有部分重叠字符模式的关键字图中的任何所标识中间节点和第二图中的所标识中间节点，动作 606 包括在关键字图与第二图之间添加边以对更改所标识中间节点的至少之一的字符模式进行补偿的动作（动作 609）。例如，重叠补偿器 144 可添加从非经更改节点到该经更改节点之下的节点的边以对更改经更改节点的字符模式进行补偿。参照图 5，可添加从节点 512 至节点 513 的边 514 以对更改节点 511 的字符模式进行补偿。

[0077] 对于相似定位且具有完全重叠字符模式的关键字图中的任何所标识中间节点和第二图中的所标识中间节点，动作 606 包括通过将关键字图中的中间节点与第二图中的中间节点组合成表示完全重叠字符模式的单个节点来将关键字图和第二图组合在一起的动作（动作 610）。例如，重叠补偿器 144 可组合标记关键字图 113L 的中间节点和标记图 123L 的中间节点。参照图 3，节点 312 和节点 313 可被组合成节点 314。

[0078] 在创建 DAG 之后，DAG 可针对文本的一部分在状态机上运行，以确定该文本部分是否与 DAG 中所表示的任何正则表达式相匹配。

[0079] 在一些实施例中，合并图与经由正则表达式的其它轮组合以便于扩展正则表达式句法（例如 *、+、或数集）。例如，当构建 DAG 以表示正则表达式时，整个正则表达式不能由 DAG 表示是可能的。例如，正则表达式可包括诸如？：或内嵌 * 运算符的字符。

[0080] 可构建越来越复杂的状态机来处理这些类型的运算符。另一替代是创建包括实际正则表达式和单个 DAG 的多个“文本处理器”。然后可使用以下算法来合并正则表达式：

[0081] 将正则表达式分解为可表示为复杂 DAG 和不可表示为复杂 DAG 的其组分。

[0082] a. 考虑：123\d\d\d(5.*3)*\d\d\d\d

[0083] b. 这可产生以下组分：

[0084] i. DAG：123\d\d\d|\d\d\d\d

[0085] ii. 正则表达式：(5.*3)*

[0086] 对正则表达式和单个 DAG 运行所有“文本处理器”。

[0087] 收集文本中发现这些文本处理器的位置（已分类，如由 DAG/Regex 确保）。

[0088] 基于 DAG 的结果及其正则表达式重新组装原始的正则表达式以确定是否发现它了。

[0089] 如果来自步骤 (3) 的结果被储存在堆（例如斐波纳契堆）集合中，则该步骤以 O(n) 为界。

[0090] 这样，所产生的 DAG 可与正则表达式引擎一起使用以产生整个正则表达式字母表的结果。多轮方法还允许执行前瞻或后顾正则表达式，而无需原地的反向跟踪或正向跟踪，这简化了系统的复杂性并有助于性能。

[0091] 因此，本发明的各个实施例将正则表达式分解成多个简单的关键字图，将那些关键字图以紧凑和有效的方式合并，并产生能执行简化正则表达式字母表的有向非循环图 (DAG)。若干这些正则表达式 DAG 然后能合并在一起以产生表示整个集合的正则表达式的单个 DAG。可在多轮方法中组合 DAG 以及其它文本处理算法和堆集合以扩展正则表达式字母表。

[0092] 本发明可具体化为其它具体形式而不背离其精神或本质特征。所述实施例在所有方面都应当被认为只是说明性的而非限制性的。因此，本发明的范围由所附权利要求书而非上述描述指示。落入权利要求书的等效方案的含义和范围内的所有改变都被权利要求书的范围所涵盖。

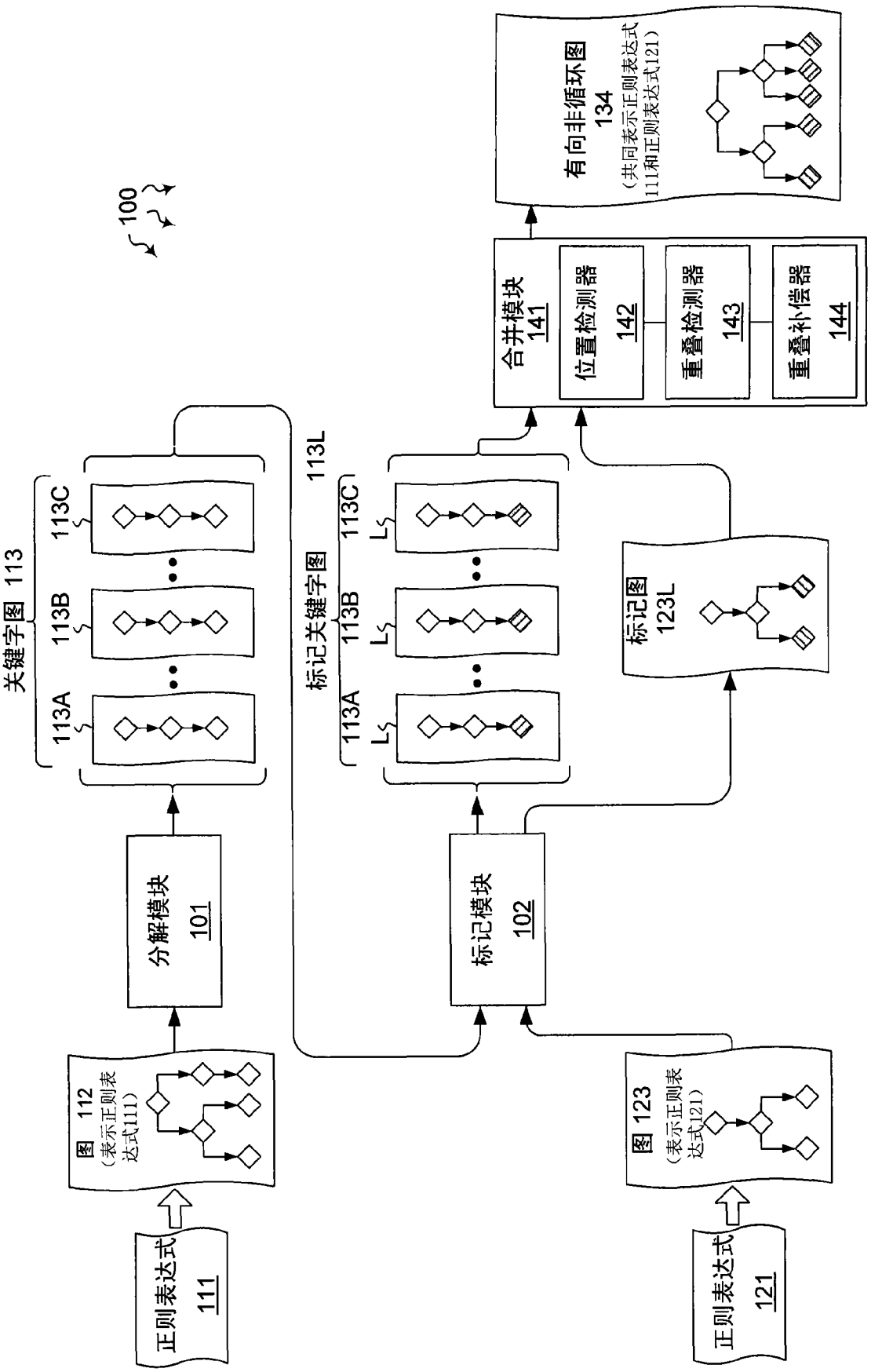

100
正则表达式
111
图 112
（表示正则表达式111）
分解模块
101
关键字图 113
113A
113B
113C
标记关键字图 113L
标记模块
102
正则表达式
121
图 123
（表示正则表达式121）
标记图
123L
合并模块
141
位置检测器
142
重叠检测器
143
重叠补偿器
144
有向非循环图
134
（共同表示正则表达式111和正则表达式121）

图 1

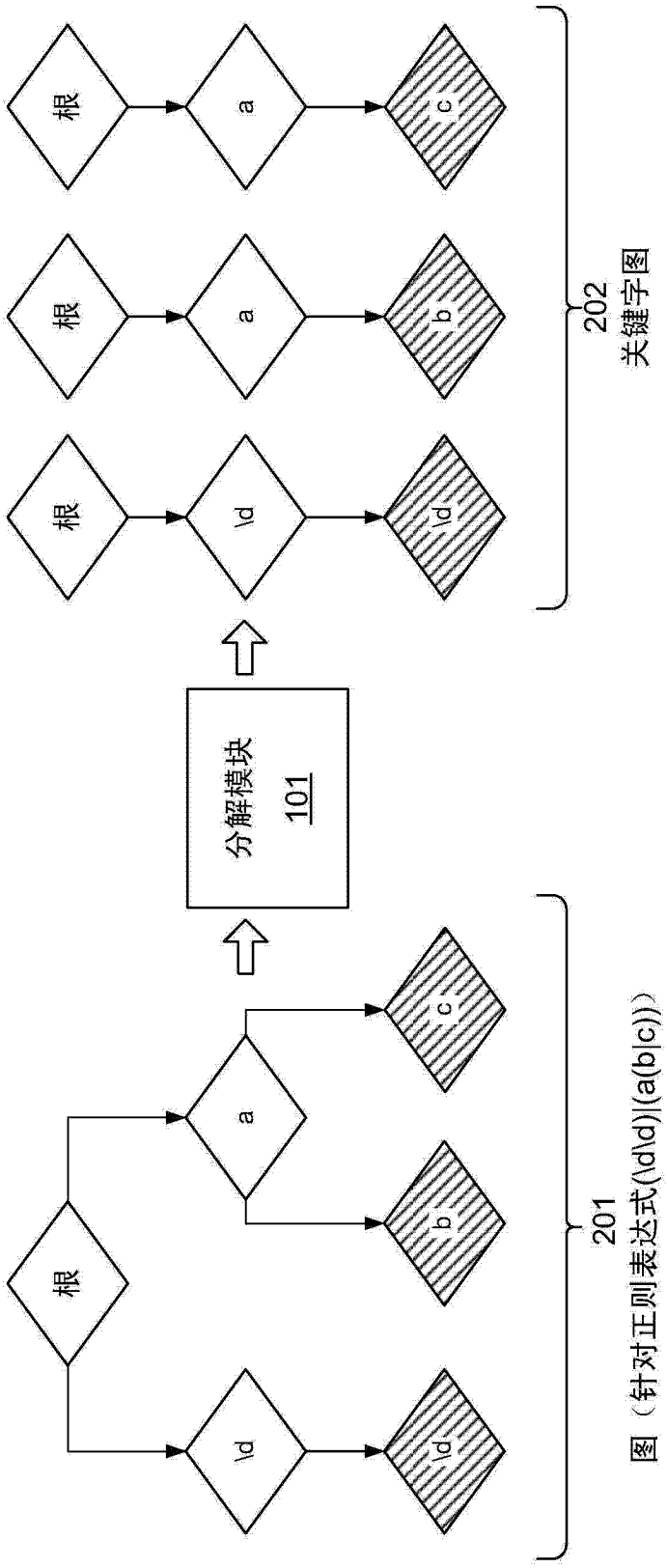


图 2

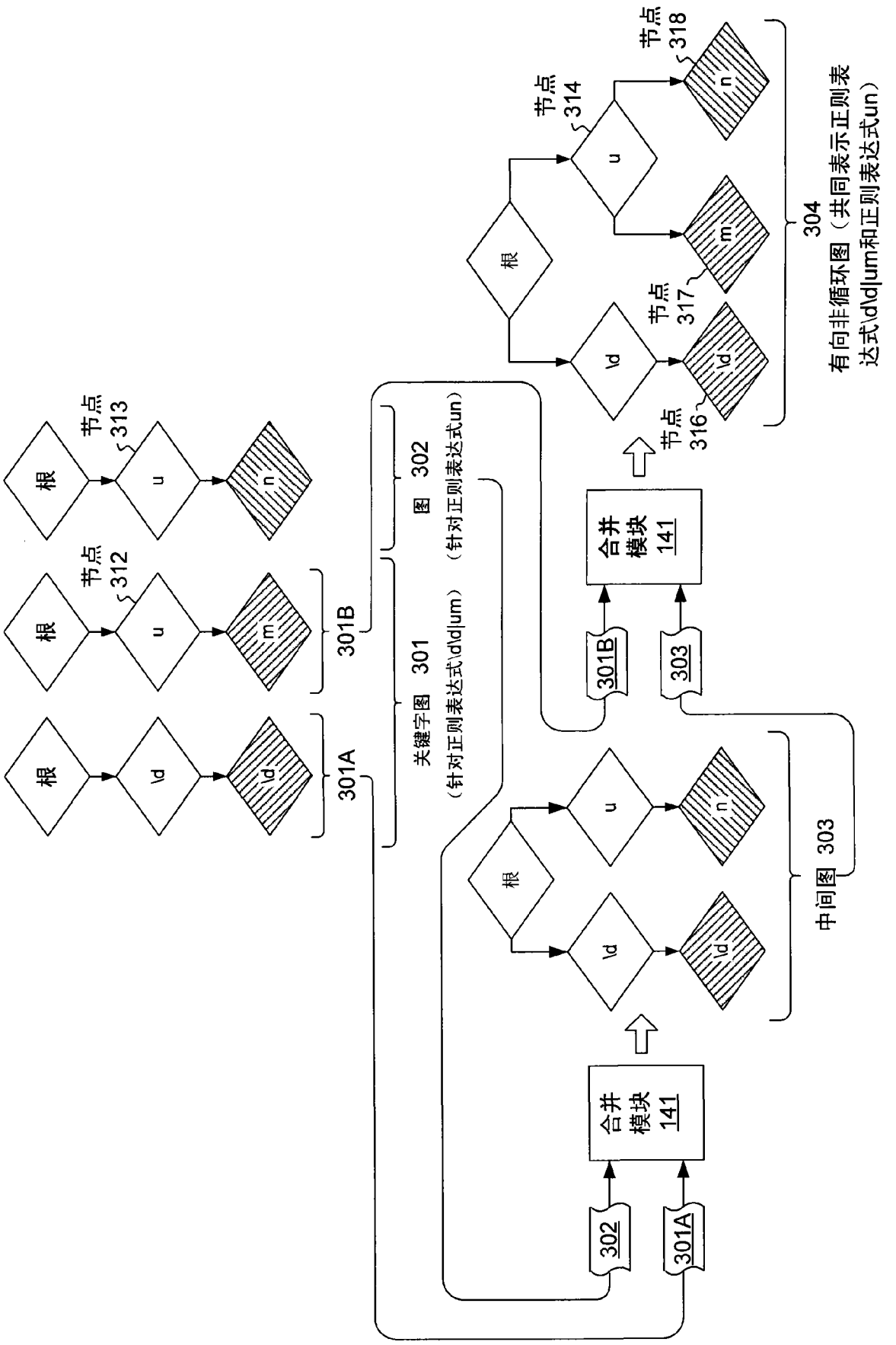

根
\d
\d
301A
根
节点
312
u
m
301B
根
节点
313
u
n
关键字图 301
（针对正则表达式\d\d|um）
图 302
（针对正则表达式un）
合并
模块
141
302
301A
根
\d
节点
316
\d
节点
317
m
根
节点
314
u
节点
318
n
304
有向非循环图（共同表示正则表达式\d\d|um和正则表达式un）
合并
模块
141
301B
303
根
\d
\d
u
n
中间图 303

图 3

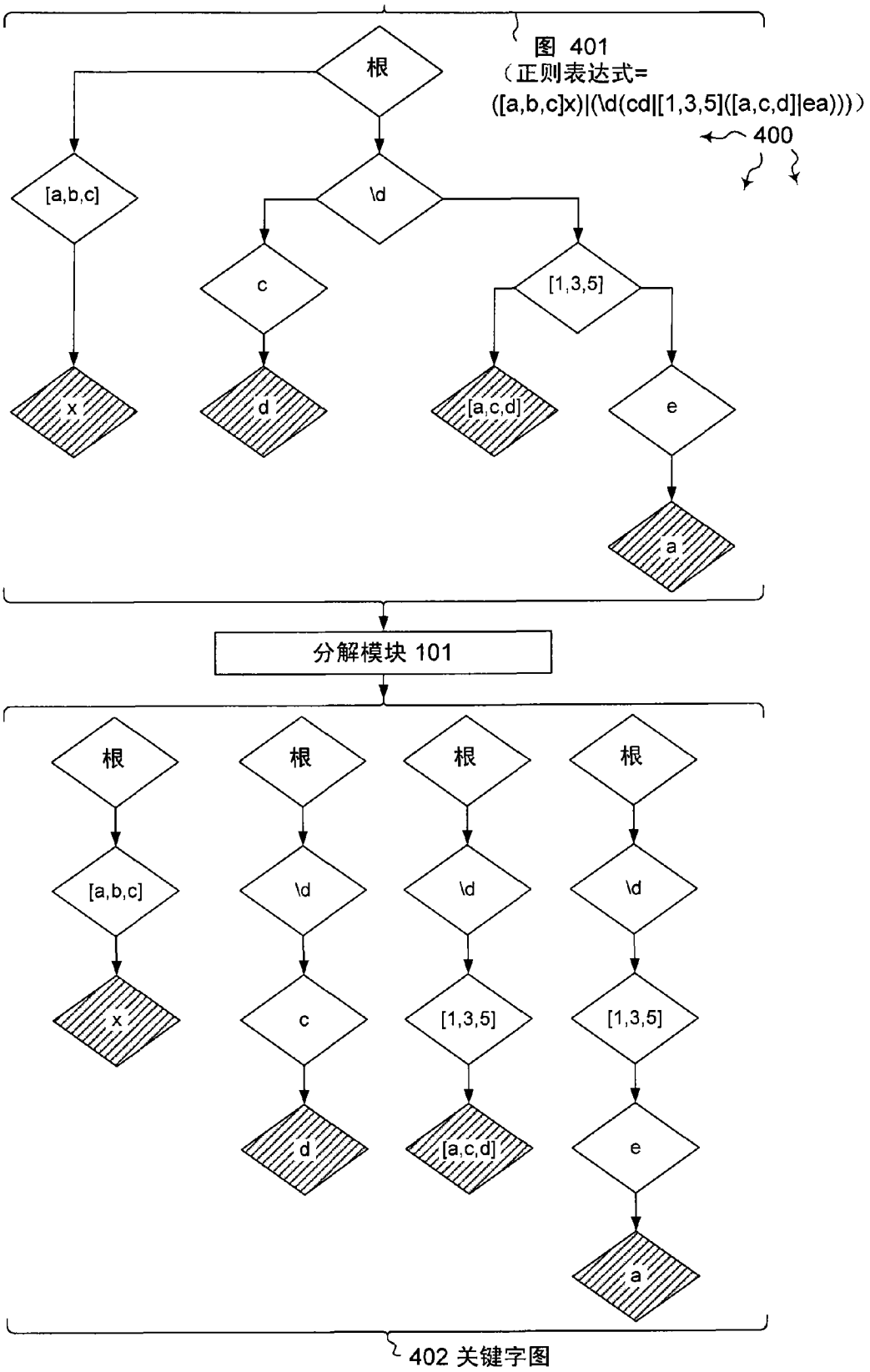


图 401
（正则表达式=
([a,b,c]x)|(\d(cd|[1,3,5]([a,c,d]|ea)))）
400
根
[a,b,c]
\d
c
[1,3,5]
x
d
[a,c,d]
e
a
分解模块 101
根
[a,b,c]
x
根
\d
c
d
根
\d
[1,3,5]
[a,c,d]
根
\d
[1,3,5]
e
a
402 关键字图

图 4

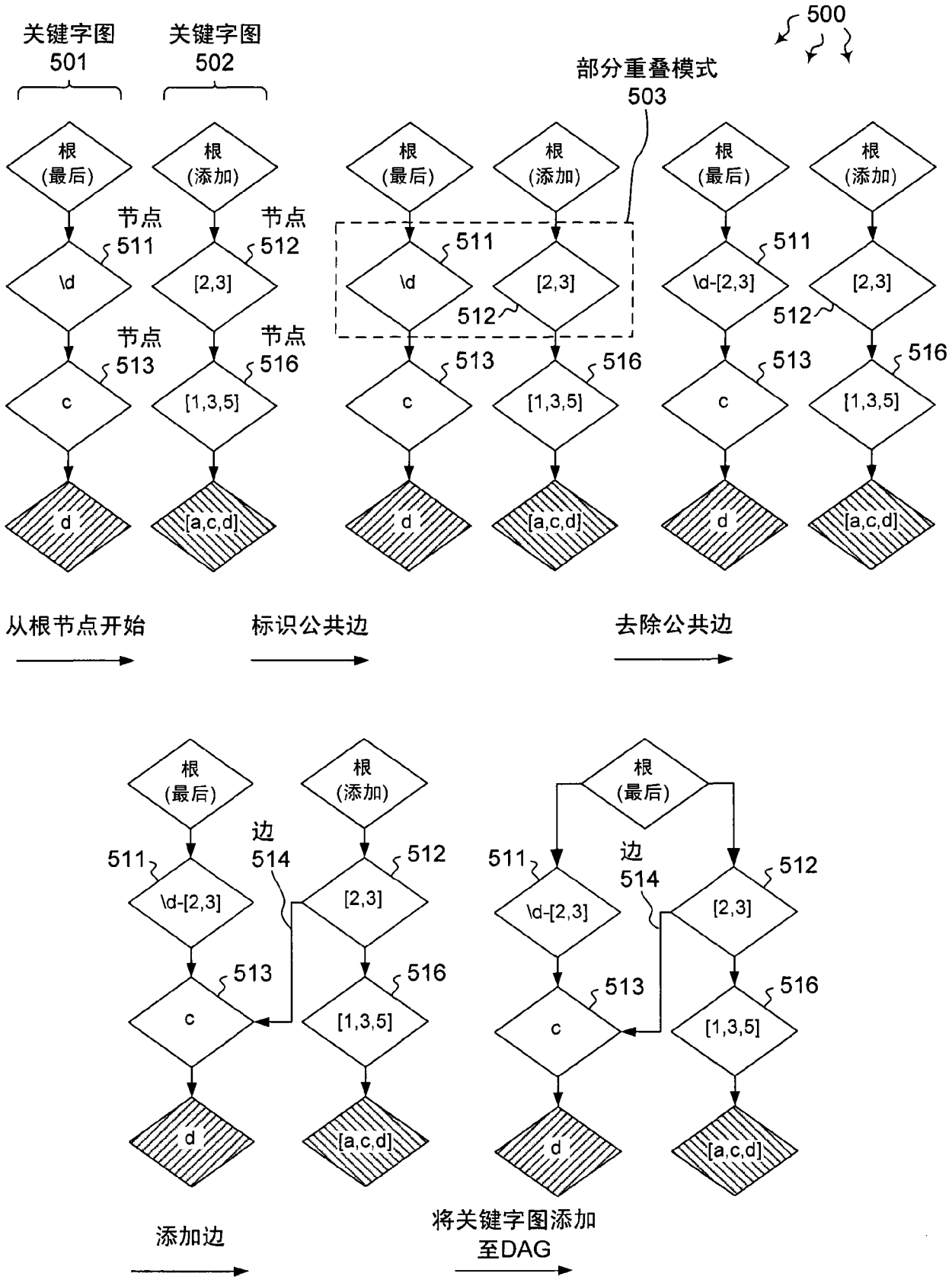

关键字图
501
关键字图
502
500
部分重叠模式
503
根
(最后)
根
(添加)
节点
511
节点
512
\d
[2,3]
节点
513
节点
516
c
[1,3,5]
d
[a,c,d]
\d-[2,3]
从根节点开始
标识公共边
去除公共边
边
514
添加边
将关键字图添加
至DAG

图 5

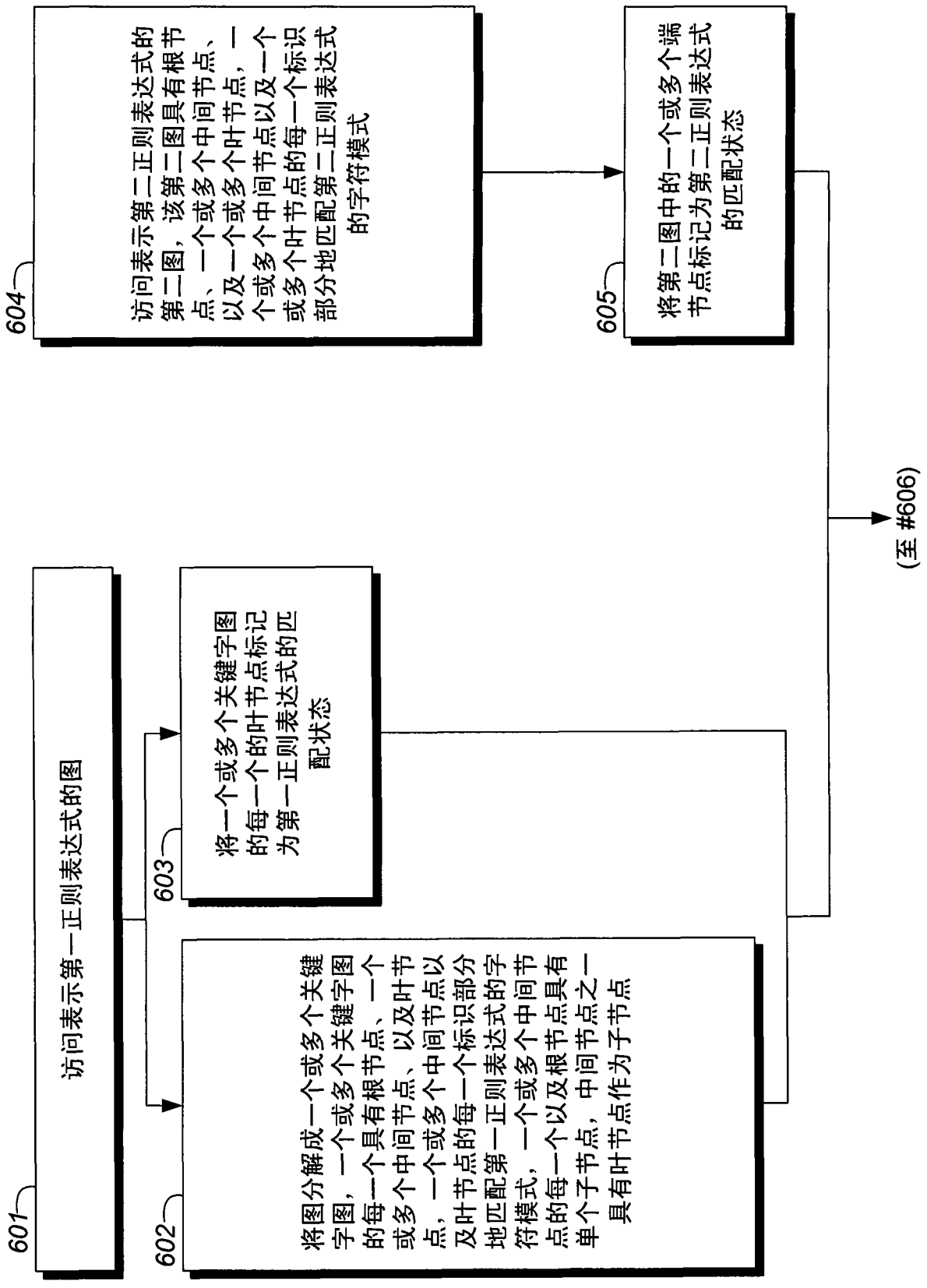

601
访问表示第一正则表达式的图
602
将图分解成一个或多个关键字图，一个或多个关键字图的每一个具有根节点、一个或多个中间节点、以及叶节点，一个或多个中间节点以及叶节点的每一个标识部分地匹配第一正则表达式的字符模式，一个或多个中间节点的每一个以及根节点具有单个子节点，中间节点之一具有叶节点作为子节点
603
将一个或多个关键字图的每一个的叶节点标记为第一正则表达式的匹配状态
604
访问表示第二正则表达式的第二图，该第二图具有根节点、一个或多个中间节点、以及一个或多个叶节点，一个或多个中间节点以及一个或多个叶节点的每一个标识部分地匹配第二正则表达式的字符模式
605
将第二图中的一个或多个端节点标记为第二正则表达式的匹配状态
(至 #606)

图 6

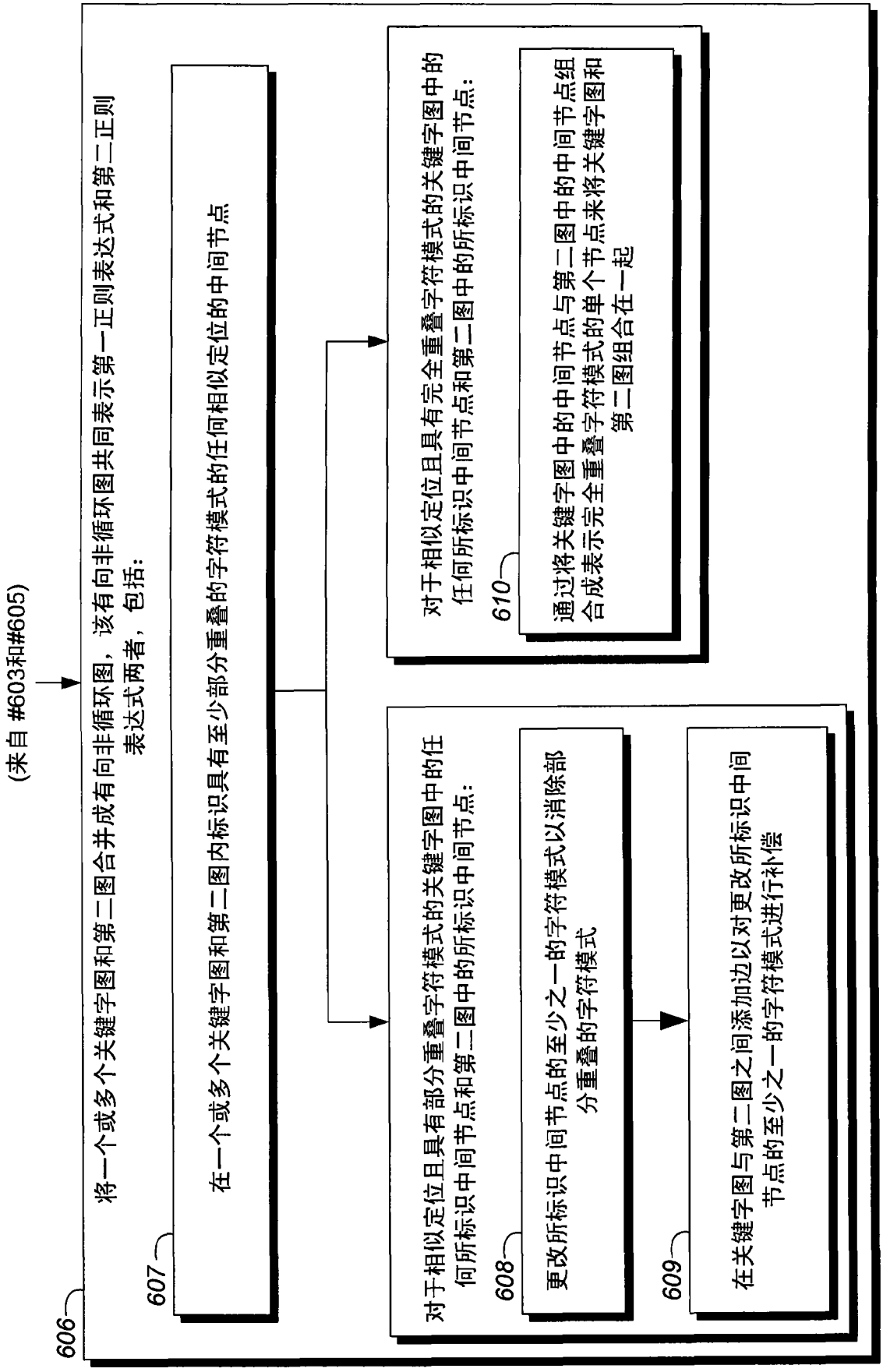

(来自 #603和#605)
606
将一个或多个关键字图和第二图合并成有向非循环图，该有向非循环图共同表示第一正则表达式和第二正则表达式两者，包括：
607
在一个或多个关键字图和第二图内标识具有至少部分重叠的字符模式的任何相似定位的中间节点
对于相似定位且具有部分重叠字符模式的关键字图中的任何所标识中间节点和第二图中的所标识中间节点：
608
更改所标识中间节点的至少之一的字符模式以消除部分重叠的字符模式
609
在关键字图与第二图之间添加边以对更改所标识中间节点的至少之一的字符模式进行补偿
对于相似定位且具有完全重叠字符模式的关键字图中的任何所标识中间节点和第二图中的所标识中间节点：
610
通过将关键字图中的中间节点与第二图中的中间节点组合成表示完全重叠字符模式的单个节点来将关键字图和第二图组合在一起

图 6（续）