



(12) 发明专利

(10) 授权公告号 CN 1842764 B

(45) 授权公告日 2010.06.23

(21) 申请号 200480024358.2

(22) 申请日 2004.06.22

(30) 优先权数据

60/482,391 2003.06.25 US

(85) PCT申请进入国家阶段日

2006.02.24

(86) PCT申请的申请数据

PCT/US2004/020834 2004.06.22

(87) PCT申请的公布数据

W02005/001687 EN 2005.01.06

(73) 专利权人 起元科技有限公司

地址 美国马萨诸塞州

(72) 发明人 克雷格·W·斯坦菲尔

(74) 专利代理机构 隆天国际知识产权代理有限公司

公司 72003

代理人 王玉双 高龙鑫

(51) Int. Cl.

G06F 9/44 (2006.01)

(56) 对比文件

US 5819021 A, 1998.10.06, 全文.

US 5999729 A, 1999.12.07, 全文.

WO 0042518 A1, 2000.07.20, 全文.

US 5117489 A, 1992.05.26, 全文.

审查员 周述虹

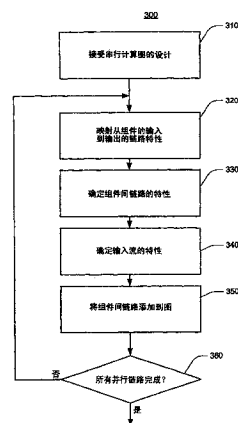
权利要求书 3 页 说明书 10 页 附图 3 页

(54) 发明名称

用于计算图的计算机辅助并行化的方法及其系统

(57) 摘要

一种用于并行计算图的自动设定或辅助设计的方法,该方法包括确定与该图的数据处理单元连接的链路单元的数据处理特性。根据与该链路单元相关的上游和/或下游数据处理单元的特性确定所述链路单元的特性,以便例如利用与相关的串行图的计算等效的并行计算图进行计算。



1. 一种用于并行计算图的自动设计的方法,该方法包括:

接受计算图的设计,在该计算图中通过链路单元连接数据处理单元,每个所述链路单元与从相关的一个上游数据处理单元到相关的一个下游数据处理单元的数据流相关联;以及

对于该计算图中并行特性不清楚的一个链路单元或多个链路单元中的每一个链路单元,根据与该链路单元关联的该上游和/或下游数据处理单元的特性来确定表示该链路单元的并行组件间链路的数据处理特性,其中存储表示一个或多个上游和/或下游数据处理单元的特性的元数据,所述元数据包括上游数据处理单元在输出端口提供的作业项的数据特性,和/或下游数据处理单元在输入端口接收的作业项的数据特性。

2. 如权利要求1所述的方法,其中将所述计算图中的每个数据处理单元表示为顶点,并且将每个链路单元表示为所述计算图的弧。

3. 如权利要求1所述的方法,还包括对于所述一个链路单元或多个链路单元中的每一个链路单元,利用数据路径上的数据处理单元来表示所述链路单元,该数据路径连接与所述链路单元相关的所述上游和下游数据处理单元。

4. 如权利要求3所述的方法,其中利用数据处理单元表示所述链路单元的步骤包括根据所述链路单元的数据处理特性配置所述数据处理单元。

5. 如权利要求4所述的方法,其中所述数据处理单元包括来自于由分割单元、集合单元以及排序单元构成的集合中的单元。

6. 如权利要求1所述的方法,还包括形成串行计算图的步骤,其中将所述计算图中的一个或多个数据处理单元分别表示为串行数据处理单元的多个实体,并且利用一个或多个串行数据处理单元,将所述一个链路单元或多个链路单元中的每一个链路单元表示为通过数据流连接到所述串行数据处理单元的多个实体。

7. 如权利要求1所述的方法,其中所述上游和/或下游数据处理单元的特性包括一个或多个所述数据处理单元的并行度。

8. 如权利要求1所述的方法,其中该数据特性包括排序特性。

9. 如权利要求1所述的方法,其中该数据特性包括分割特性。

10. 如权利要求1所述的方法,其中根据与所述链路单元关联的所述上游和/或下游数据处理单元的特性来确定所述链路单元的数据处理特性的步骤包括:确定基于所述确定的特性而形成所述链路单元的互连网络。

11. 如权利要求8所述的方法,其中所述排序特性指示:排序所述上游数据处理单元在输出端口提供的作业项的流。

12. 如权利要求8所述的方法,其中所述排序特性指示:必须排序所述下游数据处理单元在输入端口接收的作业项。

13. 如权利要求11或12所述的方法,其中所述排序特性包括定义排列次序的一个或多个关键字。

14. 如权利要求9所述的方法,其中所述分割特性指示:所述上游数据处理单元在输出端口提供的作业项的流只包括所述上游数据处理单元的被分割的作业项的子集。

15. 如权利要求14所述的方法,其中所述分割特性指示:分割该流所依据的一个或多个特定关键字。

16. 如权利要求 9 所述的方法,其中所述分割特性指示:必须分割所述下游数据处理单元在输入端口接收的数据。

17. 如权利要求 16 所述的方法,其中所述分割特性指示:必须根据一个或多个特定关键字分割所述下游数据处理单元在输入端口接收的作业项。

18. 如权利要求 1 所述的方法,其中所述数据特性包括指示符,该指示符指示所述下游数据处理单元的多个实体中的每个实体必须在其输入端口接收所有作业项的拷贝。

19. 一种用于具有一个或多个并行组件的计算图的自动设计的方法,该方法包括:

访问元数据,该元数据定义了由并行特性不清楚的链路单元上的所述一个或多个并行组件中下游并行组件接收的作业项的数据流的一个或多个输入需求的特性,其中所述输入需求包括所述作业项的数据特性;以及

设定至少一个用于处理该数据流的表示所述链路单元的并行组件间链路的功能单元,以满足该下游并行组件的输入需求。

20. 如权利要求 19 所述的方法,其中至少一个功能单元包括分割单元。

21. 如权利要求 20 所述的方法,其中所述分割单元包括哈希分割单元。

22. 如权利要求 20 所述的方法,其中所述分割单元包括循环分割单元。

23. 如权利要求 20 所述的方法,其中所述分割单元包括广播单元。

24. 如权利要求 19 所述的方法,其中至少一个功能单元包括集合单元。

25. 如权利要求 20 所述的方法,其中所述集合单元包括排序单元。

26. 如权利要求 19 所述的方法,其中设定至少一个功能单元的步骤包括设定连接两个或更多功能单元的互连网络。

27. 如权利要求 19 所述的方法,还包括:

基于组件的元数据来确定该组件的输出数据流的特性。

28. 如权利要求 27 所述的方法,其中还基于所述组件的一个或多个所述输入流的特性来确定所述输出数据流的特性。

29. 如权利要求 27 所述的方法,其中确定所述输出数据流的特性的步骤包括应用一个或多个规则。

30. 如权利要求 27 所述的方法,其中确定所述输出数据流的特性的步骤包括执行一个或多个程序语句。

31. 一种用于处理对基于图的计算的设计的系统,其包括:

用于接受计算图的设计的装置,在该计算图中通过链路单元连接数据处理单元,每个所述链路单元与从相关的一个上游数据处理单元到相关的一个下游数据处理单元的数据流相关联;以及

用于确定链路单元特性的装置,包括对于该计算图中并行特性不清楚的所述一个链路单元或多个链路单元中的每一个链路单元,根据与该链路单元关联的该上游和/或下游数据处理单元的特性来确定表示该链路单元的并行组件间链路的数据处理特性,其中存储表示一个或多个上游和/或下游数据处理单元的特性的元数据,所述元数据包括上游数据处理单元在输出端口提供的作业项的数据特性,和/或下游数据处理单元在输入端口接收的作业项的数据特性。

32. 一种用于使计算图并行化的方法,包括:

接受该计算图的设计,所述计算图包括通过并行特性不清楚的链路连接的第一组件和第二组件;

接受对该第一组件和 / 或第二组件的并行度的设计;以及

形成组件间链路,该组件间链路对应于该链路且具有至少基于该设定的并行度和存储的元数据的并行特性,其中所述元数据表示该第一组件在输出端口提供的作业项的数据特性,和或该第二组件在输入端口接收的作业项的数据特性。

33. 如权利要求 32 所述的方法,其中所述计算图是串行计算图,并且其中连接所述第一组件和第二组件的所述链路是串行链路。

34. 如权利要求 33 所述的方法,其中所述串行计算图包括连接所述计算图的多个组件的多条串行链路,并且其中该方法包括根据所述串行计算图形成并行计算图。

35. 如权利要求 34 所述的方法,其中形成所述并行计算图的步骤包括:对于所述多个串行链路中的每一条串行链路,用具有并行特性的组件间链路代替该串行链路。

36. 如权利要求 32 所述的方法,其中所述链路将所述第一组件的输出流连接到所述第二组件的输入流。

37. 如权利要求 36 所述的方法,其中所述组件间链路具有至少基于所述第一组件输出流特性的并行特性。

38. 如权利要求 36 所述的方法,其中所述组件间链路具有至少基于所述第二组件输入流特性的并行特性。

39. 如权利要求 32 所述的方法,其中接受对所述并行度的设计的步骤包括访问与所述计算图的组件关联的元数据。

40. 一种由计算机实现的方法,用于处理串行计算图以形成并行计算图,该方法包括:

(a) 基于输入流中作业项的数据特性和由组件可修改作业项的指示,将该并行图中该组件的一个或多个输入流的特性映射到该组件的一个或多个输出流的特性;

(b) 确定功能单元的特性,所述功能单元基于从两个组件之间的并行特性不清楚的组件间链路接收数据的组件所需的输入特性,来实现所述组件间链路;以及

(c) 基于来自其他组件的输出流的特性确定组件的输入流的特性,并确定用于连接所述其他组件与所述组件的链路的功能单元的特性。

41. 如权利要求 40 所述的方法,还包括:

(d) 将所述组件间链路的网络表示插入所述并行计算图。

42. 如权利要求 40 所述的方法,在使所述串行计算图并行化的过程中,重复应用(a)-(c)中的每一个处理。

43. 如权利要求 40 所述的方法,其中所述由组件可修改作业项的指示包括指示该组件可修改作业项中关键字的值的指示符。

44. 如权利要求 40 所述的方法,其中所述由组件可修改作业项的指示包括关于由该组件实现的分类、重新排序或分割的指示符。

用于计算图的计算机辅助并行化的方法及其系统

[0001] 交叉参考

[0002] 本申请要求 2003 年 6 月 25 日提交的美国临时专利申请 No. 60/482,391 的优先权，在此通过参考援引该申请的全部内容。

技术领域

[0003] 本发明涉及数据的并行处理，特别涉及并行计算图 (computation graph) 的计算机辅助设计 (computer-aided specification)。

背景技术

[0004] 通常，可将复杂计算表示为流过有向图的数据流，其中计算组件与图的顶点关联，并且组件之间的数据流对应于图的链接（弧，边）。在名称为“执行用图表示的计算”的美国专利 No. 5,966,072 中描述了实现这种基于图的计算系统。

[0005] 参照图 1A，计算图 100 的实例包括输入文件 110 和输出文件 140。输入文件 110 是一系列作业项 (work element) 的源，所述作业项例如数据记录，各关联于事务处理系统中不同的事务。每个作业项先由组件 A120 处理，接着在串行链路 125 上传送，然后由组件 B130 处理。组件 B 的输出存储在输出文件 140 中。

[0006] 期望利用单个组件的多个实体 (instance) 来实现计算图。例如，组件的每个实体可以拥有各自不同的处理器，从而实现使计算能力全面提高的谷粒平行性 (coarse-grain parallelism)。参照图 1B，并行计算图 101 的设计包括输入文件 110 和输出文件 140，这一点与串行计算图 100 相同。并行组件 A121 表示组件 A120 的 m 个并行排列的实体，并且并行组件 B131 表示组件 B130 的 m 个并行表示的实体。并行链路 126 连接并行组件 A121 和并行组件 B131。在并行计算图的表示中，例如图 1B 所示的并行计算图，用粗线表示并行组件，并在组件旁边设置表示并行度（例如，图 1B 中的“m”）的可选标识符。

[0007] 参照图 1C，并行计算图 101 以显式 (explicit) 串行形式来表示，在该图中并行排列组件 A120 的 m 个实体（标记为 A1 到 Am）。为了从输入文件 110 分配多个作业项，在输入文件 110 和形成并行组件 A121 的组件 A 的 m 个实体之间插入 1:m 分割单元 (partition element) 115，其中并行组件 A121 包括组件 A120 的 m 个实体。分割单元 115 在一个输入端接收多个作业项，并以例如循环的方式将每个输入发送到 m 个输出端之一。m:1 集合单元 (gather element) 135 在 m 条输入线上接收 m 个组件 B120 的输出，并且根据例如输入的到达时间而将所述输入合并，以输出到输出文件 140。在该实例中，将并行链路 126 表示为连接组件 A 和组件 B 相应实体的串行链路的并行组合。

发明内容

[0008] 一般，在一个方案中，本发明的特征在于提供一种自动设定并行计算图的方法。接受第一计算图的设计。该图具有通过链路单元连接的数据处理单元，并且每个链路单元与上游数据处理单元和下游数据处理单元相关联。对于并行特性不清楚的一个链路单元或多

个链路单元中的每一个链路单元,根据与链路单元相关的上游和 / 或下游数据处理单元的特性来确定该链路单元的数据处理特性,其中存储表示至少一些上游和 / 或下游数据处理单元的特性的元数据,所述元数据包括上游数据处理单元在输出端口提供的作业项的数据特性,和 / 或下游数据处理单元在输入端口接收的作业项的数据特性。

[0009] 可将每个数据处理单元表示为计算图的顶点,并且可将每个链路单元表示为计算图的弧。

[0010] 一般,在另一方案中,本发明的特征在于提供一种用于设定具有一个或多个并行组件的计算图的自动方法。该方法包括使用元数据定义由并行特性不清楚的链路单元上的下游并行组件接收的作业项的数据流的输入需求的特性,并设定用于处理数据流的表示所述链路单元的并行组件间链路的至少一个功能单元,以满足下游并行组件的输入需求,其中所述输入需求包括所述作业项的数据特性。

[0011] 该功能单元可以包括分割单元。分割单元可以包括,例如,哈希分割单元、循环分割单元或广播单元。

[0012] 功能单元可以包括集合单元,还可以包括排序单元。

[0013] 互联网络可以链接所述功能单元。

[0014] 该方法还可包括基于组件的元数据确定组件输出数据流的特性。这些特性也可以或另外基于组件的一个或多个输入流的特性。确定输出流的特性的处理可以包括应用一个或多个规则,和 / 或可以包括执行一个或多个程序语句。

[0015] 一般,在另一实例中,本发明的特征在于提供一种使计算图并行化的方法。接受计算图的设计。计算图包括由并行特性不清楚的链路连接的第一组件和第二组件。同样接受对该第一组件和 / 或第二组件的并行度的设计。形成对应于串行链路的组件间链路,其具有至少基于设定的并行度和存储的元数据的并行特性,其中所述元数据表示该第一组件在输出端口提供的作业项的数据特性,和或该第二组件在输入端口接收的作业项的数据特性。

[0016] 一般,在另一方案中,本发明的特征在于提供一种由计算机实现的使串行计算图并行化的方法。该方法包括重复执行以下步骤,包括:(a) 基于输入流中作业项的数据特性和由组件可修改作业项的指示,将该并行图中该组件的一个或多个输入流的特性映射到该组件的一个或多个输出流的特性;(b) 确定功能单元的特性,所述功能单元基于从并行特性不清楚的链路接收数据的组件所需的输入特性来确定两个组件之间的链路;以及(c) 基于来自其他上游组件输出流的特性确定组件输入流的特性,并确定连接所述其他上游组件与所述组件的链路的功能单元的特性。

[0017] 通常,在另一方案中,本发明的特征在于提供一种实现根据排列次序对数据流进行排序的方法,其中除了排序的数据之外,还在流上传送与排列次序相关的一个或多个指示符。至少某些指示符标识该数据的排列次序中的一个位置,使得该流上的后续数据不会早于该排列次序中该标识的位置而出现。

[0018] 本发明的方案可以具有一个或多个以下优点:

[0019] 在没有任何用户输入或有限的用户输入的情况下,可使串行计算图并行化,从而简化了并行计算图的设计过程。

[0020] 该自动程序不易出错,这是因为该自动系统能够验证图中组件的输入需求是否被

满足,而不是依靠用户去满足所述输入需求。

[0021] 本发明的方案包括关于保证并行计算图与初始串行(或并行)计算图的功能等效的技术考虑。一个技术效果为能够对串行计算图设定的计算进行分配,以在多个独立处理器上并行执行,从而与串行执行相比增加了并行执行的吞吐量。

[0022] 通过以下的说明书和权利要求书,本发明的其它特点和优点将更显而易见。

附图说明

[0023] 图 1A 是串行计算图。

[0024] 图 1B 是并行计算图。

[0025] 图 1C 是图 1B 所示的并行计算图的串行表示。

[0026] 图 2A 是并行计算图的一部分。

[0027] 图 2B 是具有组件间链路上表示的组件的并行计算图的一部分。

[0028] 图 2C 是图 2B 所示的并行计算图部分的串行表示。

[0029] 图 3 是自动并行化程序的流程图。

具体实施方式

[0030] 参照图 1A-图 1C,图 1A 所示的串行计算图 100 为简单计算图的实例,在某些情况下其可实施为图 1B-图 1C 所示的并行计算图 101。例如,如果可以相对于所有其它作业项独立地处理来自输入文件 110 的每个作业项,则并行计算图 101 将执行与串行计算图 100 相同的计算。注意,尽管执行了相同的计算,但是串行图 100 中输出文件 140 接收到的作业项的次序不必与并行图 101 相同。在这个实例中,作业项的输出次序并不影响计算网络的功能。

[0031] 根据组件 A120 的特性,可能需要特定类型的分割单元 115 来划分输入单元,以在组件 A 的不同实体中进行处理。例如,如果计算图 100 用于处理分别与特定帐户相关联的多个交易记录,则为使并行计算图在功能上等效于串行图,可能要求由组件 A120 的同一实体来处理任一特定帐户的所有记录。由于在串行图 100 中组件 A120 只有一个实体,因而满足上述需求。在并行图 101 中,对于某些形式的 1:m 分割单元 115,例如循环分割单元,则不能保证满足特定帐户的所有记录都进入组件 A 的同一实体的需求。在该实例中,适用的分割单元 115 根据哈希函数将每个记录中帐户字段(field)的值映射为 m 个不同的值,每个值与分割单元 115 的一个输出端相关联。对应于任一特定哈希值的所有记录都被发送到相同的输出端,从而可由并行图 101 中组件 A120 的同一实体进行处理。将所述 1:m 分割单元 115 称为哈希分割单元。哈希函数被设计为使得可将均衡数量的记录平均地发送到分割单元 115 的每个输出端。

[0032] 在基于串行计算图 100 设定并行计算图 101 的过程中,用户改变串行图的设计以保留所需的串行图的功能。例如,对于图 1A 所示的串行图,用户可以设定组件 A120 和组件 B120 的并行度(m)(在该实例中二者具有相同的并行度),将分割单元 115 和集合单元 135 添加到图中,并设定所添加单元的特性。例如,如果组件 A120 要求根据账号来分割记录,则用户获知组件 A120 的需求,并人工地配置分割单元 115 以根据该账号的哈希值来划分记录。

[0033] 通常,设定分割单元 115 的用户可以根据组件 A120 的特性而在各种类型的分割单元中进行选择。这些类型的分割单元包括,但不必局限于:

[0034] • 哈希分割单元,利用分割作业项所依据的每个作业项中的一个或多个关键字来设定该哈希分割单元;

[0035] • 循环分割单元,其中划分作业项时不考虑作业项的关键字的值,典型地在不同的输出端之间循环;以及

[0036] • 广播分割单元,其中将每个输入作业项的拷贝传递到每个输出。

[0037] 设定并行图 101 的用户也必须设定集合单元 135 的特性。例如,可根据作业项的关键字的值,如交易记录中的账号,对输入作业项进行排序。在串行图 100 中可保留该次序。然而,在并行图 101 中,如果不同的支路即使以略微不同的速率处理其输出,则该次序也可能被扰乱。因而,如果要使产生的输出文件 140 中的次序与串行图 100 保持的次序一致,则用户可以对集合单元 135 进行设定,以使其根据从不同支路接收到的作业项中例如账号的特定关键字来对输入进行排序。

[0038] 根据输出集合单元 135 的期望特性以及关于对集合单元的输入而做出的任何假设,设定并行图的用户可以选择某一类型的集合单元。可用的集合单元的类型包括,但不局限于:

[0039] • 随机集合单元,其中以随机顺序,例如根据集合单元上的到达时间,将作业项从输入端传送到输出端;以及

[0040] • 排序集合单元,其中在将输入传送到输出端之前,根据关键字对输入进行排序。

[0041] 再参照图 1A,可能存在这样的情况:不能以既适合组件 A120 又适合组件 B 130 的方式分割来自输入文件 110 的作业项。例如,在交易记录的情况下,可能需要对于组件 A 利用账号来分割作业项,而对于组件 B 则利用收款人来分割。在这种情况下,通常不会接受如图 1C 所示在组件 A 和组件 B 的对应实体之间进行直接链接。在另一种情况下,即在组件 A 期望的并行度与组件 B 不同的情形下,也不能建立这种直接链接。例如,如果组件 A 需要两倍于组件 B 的源,则设定的组件 A 的实体数量也应是组件 B 的实体的两倍,因而不能在组件 A 和组件 B 的不同的对应实体之间进行直接链接。

[0042] 参照图 2A-图 2C,在更通常的实例中,以几个步骤设定在功能上等效于串行图的并行计算图。图 2A 示出包括并行组件 A210 和 B240 的并行图 200 的一部分,其中并行组件 A210 和 B240 通过组件间链路 (ICL) (inter-component link) 205 进行链接。在图 2A 的图示中,ICL205 的并行特性并不清楚。以如下方法确定这些特性,即,使得由并行图 200 进行的计算等效于组件 A 和组件 B 的并行度都为 1 的串行图。在图 2A 中,指定组件 A 具有并行度 m 且组件 B 具有并行度 n ,这里 m 不必等于 n 。ICL205 的特性取决于如下因素,其可包括并行组件 B240 的串行实体的输入需求 (例如分割或次序要求) 和 / 或并行组件 A210 的串行实体的输出特性 (例如分割或排列特性)。

[0043] 参照图 2B,第二并行图 201 将 ICL205 表示为互连单元的网络。这个网络提供并行组件 A121 与并行组件 B131 之间的链接,并对构成并行组件的串行组件之间的作业项进行适当地“混洗 (shuffling)”,以使该图的整体功能无误。ICL205 的网络表示利用并行分割单元 221、互连网络 225 以及并行集合单元 231 进行这种混洗,其中并行分割单元 221 接收并行组件 A121 的输出,并行集合单元 231 的输出向并行组件 B131 提供了输入。

[0044] 在该实例中,并行组件 A210 具有并行度 m ,而组件 B240 具有并行度 n ,这里 m 不必等于 n 。并行组件 A121 与并行分割单元 221 之间的并行链路表示 m 个串行链路。并行分割单元 221 的输出端表示 mn (n 的 m 倍) 个串行链路。互连网络 225 具有 mn 个输入端和 mn 个输出端。并行集合单元 231 具有 mn 个输入端和 n 个输出端。

[0045] 参照图 2C,在图 2B 所示的并行计算图 201 部分的串行表示中,并行分割单元 221 可实施为分割单元 220 的 m 个实体。这些分割单元中的每一个是与图 1B- 图 1C 所示的分割单元 115 类似的 1: n 分割单元。并行集合单元 231 可实施为集合单元 230 的 n 个实体。这些集合单元中的每一个是与图 1B- 图 1C 中的集合单元 135 类似的 m :1 集合单元。互连网络 225 实施为串行链路的交叉连接,其中分割单元 220 的每一个实体连接到集合单元 230 的每一个实体。在并行图的一些可选表示中,可利用单个符号或图标来表示分割单元 221 和互连网络 225 的组合,并且该组合也称之为“分割单元”。

[0046] 这样,设定 ICL205 的网络表示的用户就设定了并行分割单元 221 和并行集合单元 231 (与形成组件间链路 (ICL) 205 的互连网络 225 一起) 的特性,其中,如图 2B 所示,该 ICL205 链接了并行组件 A121 和并行组件 B131。用户对这些特性的选择基于下游组件 B130 的识别需求,以及用户对于构成上游并行组件 A121 的组件 A120 的实体的输出特性所做出的各种假设。

[0047] 作为组件间链路 205 特性设计的实例,假设组件 A210 需要根据账号分割作业项,而组件 B240 需要根据邮政编码 (postal zip code) 分割输出。假设适当地分割了对组件 A 的输入,然后也以相同的方式分割组件 A 的输出。也就是说,在这个实例中,将根据账号分割图 2B 中组件 A210 的输出。组件间链路的每个 1: n 分割单元 220 都是哈希分割单元,其利用每个作业项内的邮政编码关键字来决定将作业项传送到哪个输出端。具有相同邮政编码的作业项通常由组件 A210 的不同实体来处理,从而将通过 1: n 哈希单元 220 的不同实体。来自对应于相同哈希值的每个 1: n 哈希分割单元 220 的输出被传递到组件间链路的同一集合单元 230。在这个实例中,在组件 B 处出现的作业项的顺序并不重要,因此每个集合单元 230 都按照作业项的到达顺序将其传送到对应的组件。

[0048] 一种使串行图并行化的自动或计算机辅助方法,其可确定 ICL205 的网络表示以及该网络表示的全部链路和单元 (例如分割单元) 的特性。用户可使用如图 2A 所示的图的表示,而不必考虑如图 2B 所示的 ICL205 的网络结构。该方法添加组件间链路的单元以实现与上述方法基本相同的效果,而无需用户显式地插入组件间链路的网络表示的分割、互连和集合单元。

[0049] 该自动或计算机辅助方法的一个方案涉及定义了计算图中部分或所有组件的输入需求特性的信息的使用,以及确定部分或所有组件输出特性的方法。用于上述目的的信息存储为与组件相关的元数据 (metadata)。该元数据用于自动并程序。

[0050] 串行图 (例如如图 200) 中的一个或多个组件分别包括与组件所需的每个输入的特性相关的元数据。例如,如果需要以特定的方式分割组件的特定输入,则该输入端的输入元数据可以包含分割作业项所根据的关键字或字段的指示符 (indicator)。如果组件具有多个输入,则每个输入都分别具有与其相关的元数据。例如,一个输入可以指示必须将所有作业项的拷贝传递到该输入,而另一个输入可以表示必须利用每个作业项的账号分割所述作业项。

[0051] 组件的输入的元数据特性可以包括一个或多个：

[0052] • 指示符,指示如果分割,则必须根据一个或多个特定关键字分割输入；

[0053] • 指示符,指示组件的每个实体必须在它的输入端接收所有作业项的拷贝；以及

[0054] • 指示必须对输入排序的指示符,以及定义排列次序的一个或多个关键字。

[0055] 一个或多个所述组件的另一特性涉及基于组件输入流的特性和组件本身特性的每个输出流的特性。这种特性的一个实例为对于处理每个作业项的组件,其顺序地接收其输入流。对于这种组件,如果根据特定关键字对输入作业项进行排序,则由于在该组件内不进行重新排序,从而输出作业项也按照相同的关键字进行排序。同样地,如果组件接收到根据特定关键字进行分割的作业项,如果该组件未修改该关键字的值,则输出作业项将保持根据该关键字而进行分割。(注意,如果组件修改了该关键字的值,则输出将不必根据被修改的关键字的值而进行分割。)

[0056] 通常,这种特性定义了特定组件(component-specific)映射函数,该映射函数接受组件的每个输入流的特性,并产生每个输出流的特性。

[0057] 流特性类似于组件的输入需求特性。具体地,流特性可以包括：

[0058] • 指示符,指示该流只包括被分割的作业项的子集(subset),以及如果分割,则分割该流所依据的特定关键字或多个关键字；以及

[0059] • 指示该流被排序的指示符,以及定义排列次序的一个或多个关键字。

[0060] 可以使用很多可选方式对从组件的输入特性到输出特性的映射进行编码。例如,可以用与该组件相关的程序语句对该映射直接进行编码。对该映射进行编码的另一种方法基于指示组件可修改作业项中关键字的值的指示符,因而可能打乱排列次序或基于该关键字的分割,以及关于由该组件显式地实现的分类、重新排序或分割的显式指示符。然后,例如使用一组通用的或特定组件的规则使得该映射基于所述输入特性和这些指示符。这种映射规则的实例包括以下几点：

[0061] • 对于具有一个输入和一个输出且未指示其修改关键字 key1 的值的组件,根据 key1 进行分割的输入产生仍然根据 key1 进行分割的输出。

[0062] • 对于具有一个输入和一个输出且未指示其修改关键字 key2 的值的组件,先根据 key1 然后根据 key2 进行排序的输入产生只根据 key1 进行排序的输出。

[0063] • 对于指示其重新排列输入作业项的组件,由于组件可能已打乱了排列次序,因而已排序的输入产生未指示其再进行排序的输出；以及

[0064] • 根据关键字 key1 的值实现排序的组件将指示根据 key1 对输出进行排序,而不考虑输入的排列次序。

[0065] 某些组件可以重新命名作业项中的字段。用于该组件的元数据识别出该重新命名,并且输出特性的表示基于重新命名的变量。例如,如果根据“账户”关键字对输入进行分割,并且将该“账户”字段重新命名为“客户”字段,则将输出的特征定义为根据该“客户”字段进行分割。

[0066] 参照图 3,用于根据串行计算图设定并行计算图的自动程序 300 首先接受将被并行化的串行图的设计(步骤 310)。该设计包括图结构的设计,也称之为图的拓扑,其设定每个组件的哪个输出通过流连接到另一个组件的每个输入。该设计也包括上述每个组件的元数据：一个或多个组件的每个输入的输入需求,以及所述组件的输入特性和输出特性之间

的映射或该组件的其他输出特性。注意,因为有些组件可能未设定任何输入需求,并且有些组件可能未提供产生其输出流特性的映射,因而上述元数据是可选的。如果组件未设定任何输入特性,则程序不必强制其输入流具有任何特定的特性。如果组件未提供产生其输出流特性的映射,则程序不必做出关于这些特性的任何假设。

[0067] 在该程序中,用户设定计算图中每个组件的期望并行度。作为该程序应用的实例,将参考图 1A 所示的简单串行计算图 100。假设用户设定组件 A120 具有 $m = 3$ 个并行实体,并且组件 B 具有 $n = 5$ 个并行实体。在该实例中,输入文件 110 和输出文件 140 的并行度为 1,表示支持串行访问的物理文件。

[0068] 对于初始串行图中的每个链路,程序确定组件间链路的特性(即分割单元、互连网络以及集合单元)以实现并行计算图中的串行链接。该程序在以下四个步骤中循环:

[0069] (a) 映射从一个或多个组件的输入到输出的链路特性(步骤 320);

[0070] (b) 对于已知上游组件的输出特性的每个组件间链路,基于任何已知的下游输入的需求来确定该组件间链路的特性,包括该组件间链路的分割单元、互连网络以及集合单元(步骤 330);

[0071] (c) 对于在步骤 (b) 中处理的每个组件间链路,确定所产生的与组件间链路的输出相关联的下游组件的输入流特性(步骤 340);以及

[0072] (d) 在并行图的相应组件之间插入组件间链路的网络表示的组件(步骤 350)。

[0073] 当已对所有的并行链路进行了处理(步骤 360),则通过展开每个并行组件可形成等效的串行计算图。

[0074] 在程序的映射步骤(步骤 320)中,计算组件的一个或多个输出流的特性,所述组件通常少于所有组件。例如,在初始的迭代(iteration)中,可以计算没有输入的组件所输出的流的特性。在后面的迭代中,可将任意组件的映射用于确定该组件的输出流的特性,其中已对该组件的所有输入流计算了输入特性。

[0075] 在下一步骤(步骤 330)中,基于该链路的上游组件(m)和下游组件(n)的并行度、来自上游组件的输出流的特性以及下游组件的输入流的需求,确定一个或多个组件间链路的分割单元和集合单元的特性。有几种情况可以直接处理:

[0076] (P1) 如果 $m = n$ 且不必根据任何特定关键字对下游组件的输入流进行分割或排序,并且输入流不需要每个作业项的拷贝,则如图 1B 中的实例所示,直接连接上游组件和下游组件的相应实体。注意,这种情况主要对应于简化形式的分割单元和集合单元。

[0077] (P2) 如果 $m \neq n$ 且不必根据任何特定关键字对下游组件的输入流进行分割,并且该输入流不需要每个作业项的拷贝,则将组件间链路的分割单元定义为进行循环分配。

[0078] (P3) 如果下游组件的输入流需要作业项根据一组关键字而进行分割,其中所述关键字不同于分割上游组件输出流的关键字,则该分割单元根据所需关键字的值进行哈希分割。

[0079] (P4) 如果输入流需要每个作业项的拷贝,则将组件间链路的分割单元定义为执行广播功能。

[0080] 对于 (P2)-(P4) 中的各种情况,有许多关于组件间链路的集合单元的特殊处理情况,以适应下游组件的输入流特性:

[0081] (G1) 如果下游组件的输入流需要根据一组特定的关键字而对输入进行排序,并且

根据相同的关键字对上游组件的输出进行排序,则集合单元实现已排序的集合。在已排序的集合中,集合单元假设每个输入上的作业项已经进行了排序,并且其根据排序关键字选择哪个输入将被选为下一个输出,以便获得输出的正确排列次序。

[0082] (G2) 如果下游组件的输入流需要根据一组特定的关键字 ($key_1, \dots, key_J$) 而对输入进行排序,并且根据一组关键字 ($key_1, \dots, key_J, \dots, key_K$) 而对上游组件的输出进行排序,则该集合单元实现已排序的集合。对于 $J = K$ 的情况,这就成为了 (G1) 中的特殊情况。

[0083] (G3) 如果输入流需要根据一组特定的关键字而对输入进行排序,并且未根据的一致的一组关键字对上游组件的输出进行排序,则集合单元进行排序。

[0084] 也可以使用其它形式的分割单元和集合单元。例如,在 $m = n$ 并且下游组件不需要任何特定的分割,但是需要根据特定的关键字进行排序的情况下,可以通过组件间链路来连接相应的上游和下游组件,其中该组件间链路只具有可进行所需排序的单入/单出“集合”单元。注意,组件间链路的输入保持了分割,并且该组件间链路增加了排序的特性。

[0085] 在循环的下一步骤(步骤 340),根据上游组件输出流的特性和插入的组件间链路的特性确定下游组件输入流的特性。注意,通常,在这些流中至少将体现输入所需的特性。也可能体现在组件输出流的特性中反映的其他的特性。例如,在根据一个关键字对流进行分割且下游组件需要其输入根据另一个关键字而进行排序的情况下,即使只需要排序,也将对产生的流既进行分割也进行排序。

[0086] 在循环的最后步骤(步骤 350),将表示组件间链路的网络单元增加到图中。

[0087] 在循环(步骤 320-350)的每一次迭代结束时,计算组件的其他输入流的特性。如果初始计算图是非循环的,则当初始图中的所有链路都被处理完时终止该程序。

[0088] 例如,如果未计算出上游组件输出流的特性,则在确定组件间链路的特性时,用于在未处理完所有链路的情况下重启程序的方法将不对流的特性做出假设。在非周期的计算图中,或者在某些组件的元数据不可用的图中可能需要这种重启。例如,即使已经实际上对流进行了分割,也可在组件间链路中插入冗余的分割单元。虽然可能效率不高,但是仍能保持并行图的正确功能。

[0089] 在上述的自动程序中,可将组件间链路的组件的插入显式地指示给用户。可选地,用户不必知道引入到原始串行计算图的每个链路中的组件间链路的特性。用户接口能够隐藏这些分割单元和集合单元,或提供由用户控制的对这些单元的可选显示,而不是默认地显示组件间链路的网络表示。

[0090] 在计算机辅助(例如,半自动)模式中,用户通过显式地引入一些分割单元和集合单元来引导程序。例如,在应用自动程序之前,用户可以设定在并行链路中使用特定的分割单元和集合单元或其他形式的单元,从而取代用于该链路的自动程序。在计算机辅助模式的另一方案中,用户可以检查自动处理的结果,并可以修改组件间链路的分割单元和集合单元。注意,在用户修改了链接之后,该自动程序可选地将流特性从该链接传递到下游,并且下游的改变可能导致下游组件间链路的新的设计。

[0091] 在另一个计算机辅助模式中,自动程序验证是否满足每个组件的输入需求,并且如果不满足则通知用户。然后用户可以引入单元以满足输入需求,接着,系统能够自动地重新检查改进的设计。作为该模式的变形,系统可以建议对图进行修改(例如,可在组件间链

路上插入分割单元或集合单元)以满足输入需求,并且用户可以确认使用该修改建议,或者提供其他的修改(例如插入不同的单元,或者对插入的单元设计不同的特性)。

[0092] 在上述的并行方法中,在应用自动程序之前,用户对每个组件选择特定的并行度。在一种可选的方法中,用户只确定哪些组件将被并行化,或者与其并行度相关的变量,而不必指定期望的并行度的数值。由此产生确定了每个组件间链路的特性的“通用(generic)”并行计算图,但是与并行组件一样,仍未确定组件间链路的网络表示中并行单元的具体实现。当指定了组件的期望并行度的特定值时(例如,在图的运行时),以特定的并行形式实现该通用图。

[0093] 在上述的并行方法中,流特性主要以“泛洪(flooding)”方式传递,即数据流的特性在与数据流自身相同的方向上向“下游”传递。也可以使用其他的方法。例如,可以使用上游传递。当存在由两个组件A和B构成的一系列组件时可以使用这种上游传播的一个简单实例。如果A和B都具有相同的并行度,并且B需要根据关键字进行分割,而A不需要任何特定的分割,则可向上游传递该分割需求以使A也需要进行与B同样的分割。通过这种方式,可以在A和B的相应实体之间使用直接链接,而无需在A和B之间引入显式的分割单元和集合单元。

[0094] 串行图的组件也可以表示整个串行子网络。自动并行方法的一个方法是使子网络并行化,这就好像串行子网络在其主图(host graph)内完全展开。

[0095] 在替代的方法中,串行子网络独立于其所处于的网络而被并行化。计算从总体上定义子网络的特性并且包括子网络上的流特性的全部映射的元数据,以便在基于子图内组件的元数据而执行的主网络的并行化程序中使用。

[0096] 可在上述并行计算图中使用的一种单元类型为排序合并单元。如上所述,排序合并单元假设根据特定排列次序排列每个输入上的作业项,并且该排序合并单元必须根据相同的排列次序产生全部的合并输出。该排序合并单元所执行的基本程序考虑到在各输入端等待(pending)的每个作业项,并根据排列次序传送作为输出的下一作业项。

[0097] 然而,如果在任意一个输入端都没有等待的作业项,则该排序合并单元不能传送任何作业项,这是因为不知道后面到达该输入端的作业项在排列次序中是否出现在已经等待的输入之前。因而该作业项将会被阻塞直到在链路上接收到流结束指示符,此时排序合并单元可以假设在该流上再没有作业项将会到达。

[0098] 在某些情况下,会在并行图中发生这种阻塞行为。例如,假设m个实体之一或组件A的输出根据关键字key1进行排序,并且下游组件B要求其输入根据关键字key2排序并根据关键字key1进行分割。如果元数据未设定根据key1分割A的输出,则在并行化方法中使用组件间链路内根据key1进行分割的哈希分割单元,其后接有根据key2进行排序的排序集合。然而,如果组件A的输出已进行了哈希分割,则对于每个哈希分割单元,只有一个输出将接收所有作业项。从而,在排序合并单元处,只有每个排序合并单元的一个输入端将接收输入。从而在流结束指示符从哈希分割单元传递到排序合并单元之前将不必阻塞整个流。

[0099] 避免这种情况的方法是针对接收排序输入的哈希分割单元,重复在每个输出链路上发送排序值指示符,以指示通过至少一个输出到达的排列次序中的值。该排序值指示符通知下游组件在这条链接上将不会提供具有更早的排列次序值的作业单元。接收了该排序

值指示符的排序合并单元利用该指示符确定是否能够从另一个输入传送等待的作业项,例如,其原因可能为该作业项的排序值位于接收到的排序值指示符中的排序值之前。

[0100] 可以尽可能多地基于通过组件的作业项的流发送排序值指示符,或者也可以不这么频繁地发送,例如,可以根据被处理的作业项的数目或根据时间周期性地发送。可以在任何可分割的排序的流上发送排序值指示符,并且可在接收所述指示符的分割单元的输出上广播所述指示符。

[0101] 另一个解决下游排序合并单元的阻塞问题的方法是打乱(scramble)上游哈希分割单元的输入,从而通常至少能够在哈希分割单元的每个输出上传送一些作业项。

[0102] 可以利用在计算机上执行的软件来实现上述的自动方法。例如,该软件可在一个或多个编程的或可编程的计算机系统(该计算机系统可以具有各种结构,例如分布式、客户机/服务器或网格格式)上执行的一个或多个计算机程序中形成程序,其中每个计算机系统包括至少一个处理器、至少一个数据存储系统(包括易失性和非易失性存储器和/或存储单元)、至少一个输入设备或端口以及至少一个输出设备或端口。该软件可以形成一个或多个较大程序的模块,例如,该程序提供与计算图的设计和配置有关的其它服务。

[0103] 该软件可以设置在例如 CD-ROM 的可由通用或专用可编程计算机读取的介质上,或者通过网络传递(在传播的信号中编码)到执行该软件的计算机上。在专用计算机上或者使用例如协处理器(coprocessor)的专用硬件可以实现所有的功能。可以采用分布方式实现该软件,在分布方式中由不同的计算机来执行该软件所设定的不同计算部分。优选地,将每个这种计算机程序存储或下载到可由通用或专用可编程计算机读取的存储介质或设备(例如固态存储器或介质,或者磁性或光学介质)中,以便当计算机系统读取该存储介质或设备时,该计算机程序可以配置和操作该计算机以执行其中描述的程序。也可以将本发明的系统实施为配置有计算机程序的计算机可读存储介质,其中具有该配置的存储介质使计算机系统以特定或预定的方式运行以执行其中描述的功能。

[0104] 可以理解前面的描述只用于说明本发明,并非限制本发明的范围,本发明的范围由所附权利要求书的范围限定。其它的实施例都包含在所附权利要求书的范围内。

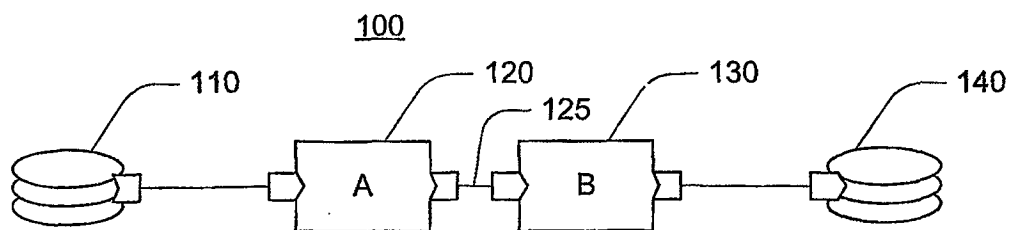


图 1A

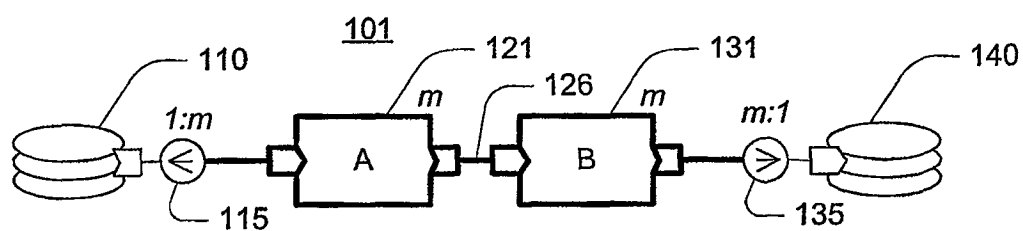


图 1B

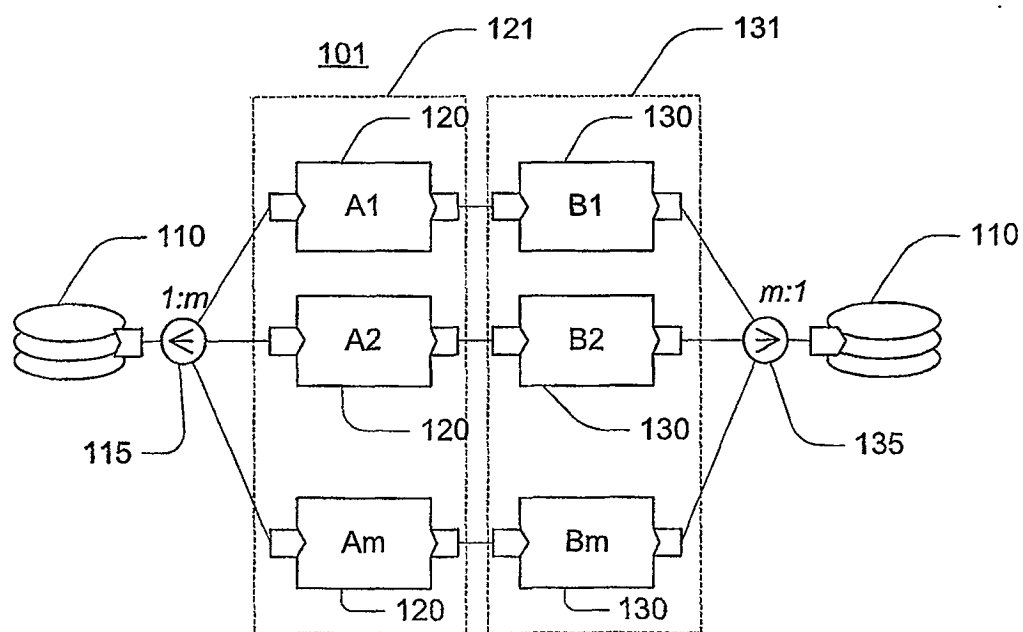


图 1C

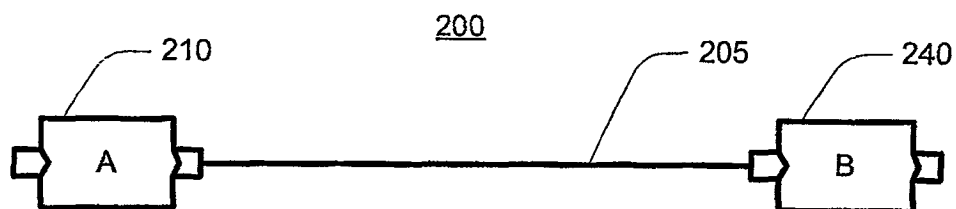


图 2A

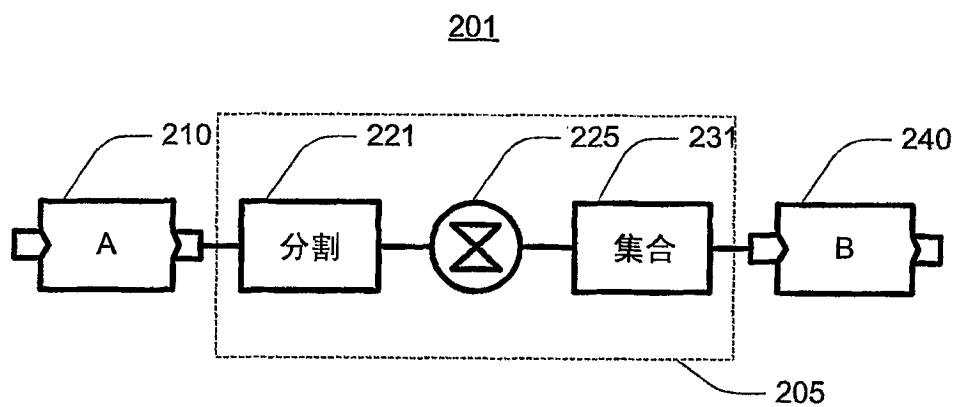


图 2B

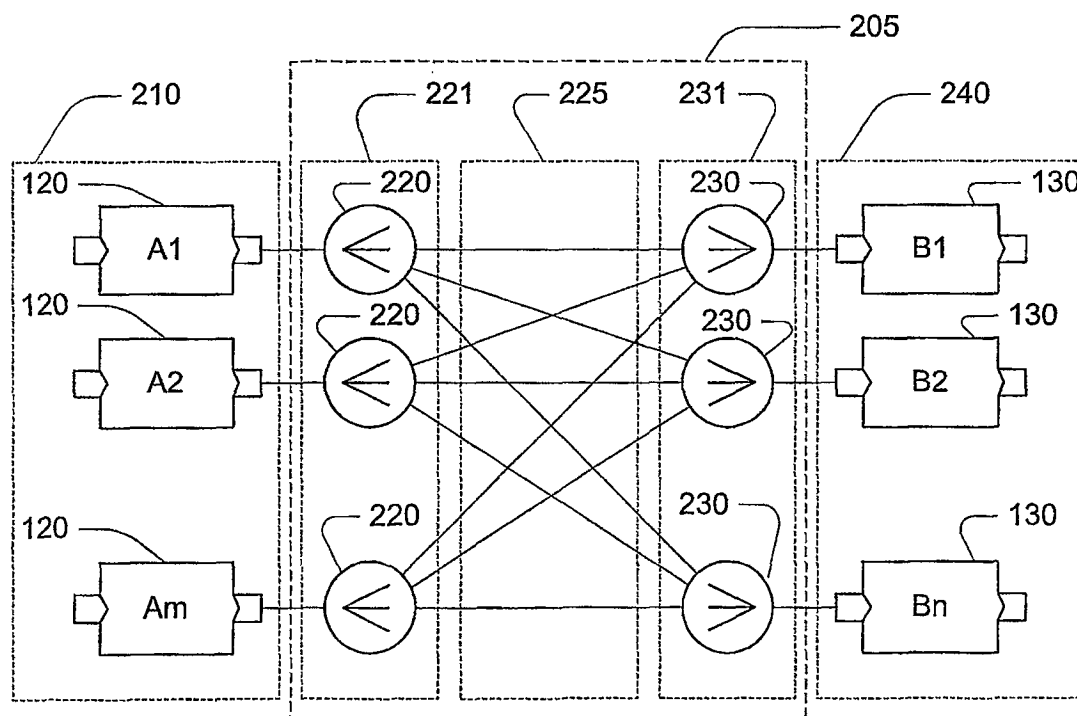


图 2C

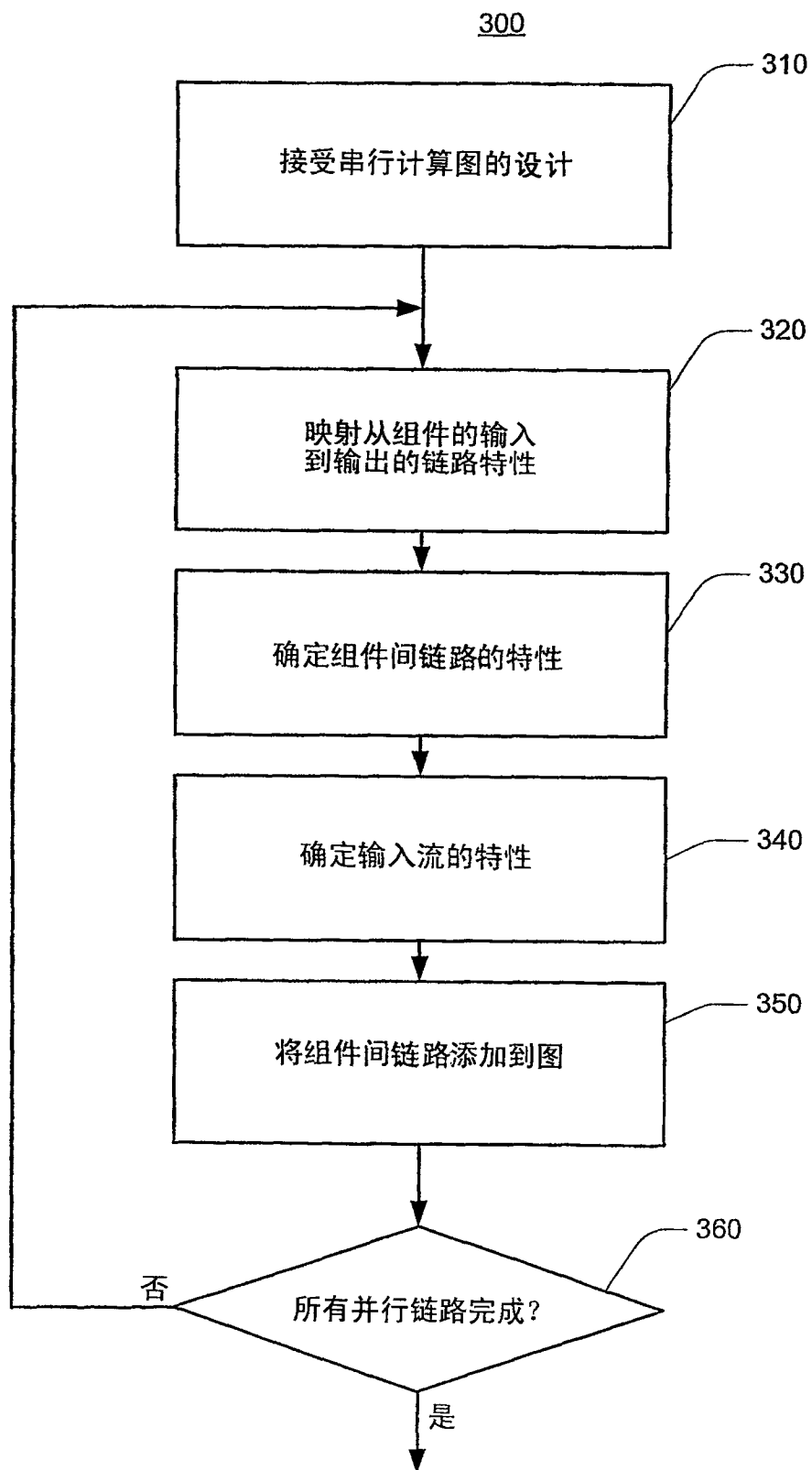


图 3