

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

H04B 10/00 (2006.01)

H04Q 11/00 (2006.01)



[12] 发明专利申请公布说明书

[21] 申请号 200610057891.X

[43] 公开日 2007年9月5日

[11] 公开号 CN 101030814A

[22] 申请日 2006.3.3

[21] 申请号 200610057891.X

[71] 申请人 中兴通讯股份有限公司

地址 518057 广东省深圳市南山区高新技术产业园科技南路中兴通讯大厦法律部

[72] 发明人 马恒 秦涛 杨静 吴庆丽

[74] 专利代理机构 北京安信方达知识产权代理有限公司

代理人 许志勇 颜涛

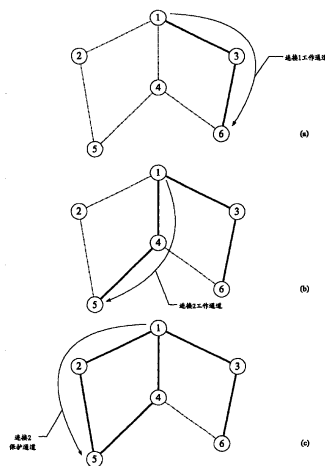
权利要求书 2 页 说明书 8 页 附图 4 页

[54] 发明名称

一种在ASON网络中为组播业务提供保护和恢复通道的方法

[57] 摘要

本发明公开了一种在ASON网络中为组播业务提供保护和恢复通道的方法，首先，为组播业务中各个目标节点的工作连接计算路由，计算中后建的连接可共享已建的连接资源；然后，根据工作连接所需的保护等级，为该工作连接确定要建立的保护连接或恢复连接，计算路由时，是在所要建立的保护连接或恢复连接不能与其对应的需要保护的工作连接使用风险共享资源的限制条件下，进行计算，并最终生成带有保护和恢复通道的组播树的生成组播树生成。本发明还进一步对所建立的具有保护功能的组播树进行重优化。本发明所述方法，既能实现通向不同的目的节点的光通道之间共享资源以提高资源的利用效率，又可以使用较少的网络资源完成对组播业务的保护功能。



1、一种在 ASON 网络中为组播业务提供保护和恢复通道的方法，其特征在于，包括如下步骤：

(1) 为组播业务中各个目标节点的工作连接计算路由，计算中后建的连接可共享已建的连接资源；

(2) 根据工作连接所需的保护等级，为该工作连接确定要建立的保护连接或恢复连接，

其中，在为所述保护连接或恢复连接计算路由时，是在所要建立的保护连接或恢复连接不能与其对应的需要保护的工作连接使用风险共享资源的限制条件下，进行计算，并最终生成带有保护和恢复通道的组播树的生成组播树生成。

2、如权利要求 1 所述的方法，其特征在于，进一步还包括：

(3) 对经步骤 (2) 所生成的组播树进行重路由优化，获取路由优化的组播树。

3、如权利要求 1 所述的方法，其特征在于，所述步骤 (1) 包括：

(1-1) 在包含 N 条工作连接的组播业务中，为源节点至目的节点 d_1 、 $d_2 \dots d_k$ 的工作连接 W_1 、 $W_2 \dots W_k$ 计算路由，形成一个从组播业务源节点至 k 个目的节点的组播树 T_k ，其中 $k < N$ ；

(1-2) 为源节点至第 $k+1$ 个目的节点 d_{k+1} 的工作连接 W_{k+1} 计算路由时，共享使用组播数 T_k 上的资源，采用节点 d_{k+1} 至组播树 T_k 的最短路径来为工作连接 W_{k+1} 计算路由，形成 N 条工作连接的组播树 T_N ，其中 $k+1 \leq N$ 。

4、如权利要求 1 所述的方法，其特征在于，所述步骤 (2) 中的所述保护等级包括不需保护、或需要保护连接、或需要恢复连接、或同时需要保护与恢复连接。

5、如权利要求 1 所述的方法，其特征在于，所述步骤 (2) 中所述所要建立的保护连接或恢复连接不能与其对应的需要保护的工作连接使用同一风险共享资源的限制条件，是指：

如果要求工作连接和保护连接或恢复预留连接是节点无关的,则两者不能经过同一个节点;

如果要求工作连接和保护连接或恢复预留连接是链路无关的,则两者不能经过相同的链路。

6、如权利要求5所述的方法,其特征在于,所述限制条件为在基于约束条件的最短路算法中,将需要保护的工作连接设置为对应保护或恢复连接必须避开的约束路由。

7、如权利要求2或6所述的方法,其特征在于,所述步骤(3)包括:

(3-1)从全部目的节点中任意选择一个目的节点,删除通向该目的节点的工作连接、和/或保护连接、和/或恢复预留连接并释放它们所独占的资源;

(3-2)重新为步骤(3-1)中经删除操作后的所选目的节点计算工作连接路径,此计算过程中共享已存在的组播树上的资源;

(3-3)根据所述目的节点的保护等级,或不处理、或为该目的节点计算保护连接路径、和/或为该节点计算恢复预留连接路径,此计算过程中共享已存在的组播树上的资源,但需要依据所述限制条件避开对应的工作连接路径;

(3-4)比较重路由前后的组播树,如果重路由之后的组播树优于重路由之前的组播树,则保留此次重路由操作,否则,抛弃此次重路由操作,恢复该组播树到重路由之前的状态;

(3-5)在业务要求的路由计算时间内,重复步骤(3-1)至步骤(3-4)若干次直到超时为止,并保留最终重路由操作后的组播树。

8、如权利要求7所述的方法,其特征在于,步骤(3-1)中的所述独占的资源是只有该目的节点的连接使用而组播树中其它连接都没有使用的资源。

9、如权利要求2所述的方法,其特征在于,步骤(3)中进行重路由计算的算法是模拟退火算法、或遗传算法。

一种在 ASON 网络中为组播业务提供保护和恢复通道的方法

技术领域

本发明涉及通信技术领域，尤其涉及一种在 ASON 网络中为组播业务提供保护和恢复通道的方法。

背景技术

自动交换光网络（ASON: Automatic Switched Optical Network）是在传统光传送网的基础上发展出来的一种新技术。ASON 网络在不需要人为管理和控制的条件下，可以依据自身所配备的一系列路由、信令、自动发现等协议机制的运作，按用户的请求来建立符合用户需求的光信道。这一前所未有的革命性进步为光网络的发展带来了质的飞跃，同时它也推动了多种新型业务和应用的发展。

光组播业务就是近几年来伴随着 ASON 网络技术的发展而出现的一种新业务，它要求在单一的源节点和多个目的节点之间建立起合适的光通道，以实现数据的传输。一个最简单的实现光组播的方法是在源节点和目的节点之间建立多条相互独立的点对点的传输通道，但这种做法的资源利用率是很低的。一个比较好的光组播实现方案应该充分考虑到组播路径上的资源共享，以达到使用最少的网络资源来满足组播需求的目标。

提高资源利用效率只是一个好的光组播实现方案的目标之一，另一个重要的目标是提高光组播业务对网络中可能会出现的各种故障的抵御能力。由于光网络中每条光通道的数据量都比较大，单条光通道发生故障就有可能影响到很多终端用户的使用，因此往往对网络生存性有比较高的要求。提高业务对故障抵御能力的常用手段是为工作通道提供备用通道，即保护通道或者恢复预留通道，一旦业务发生故障则迅速将业务信号从工作通道上切换到备用通道上。一般来说，保护通道和恢复预留通道的区别是，保护通道和工作通道形成“双发优收”的热备份关系，而恢复预留通道（下面简称为“恢

复通道”) 在正常工作过程中是不激活的, 只有在工作通道发生故障的情况下才会将业务信号切换到恢复预留通道上去。为了提高业务的生存能力, 工作通道和保护通道或者恢复预留通道应该尽量保持资源分离, 即它们不应该使用风险共享的资源组以避免这块资源发生故障而导致业务的瘫痪。

综上所述, 光组播实现方案一方面应该尽量做到使通向不同目的节点的光通道之间共享资源以提高资源的利用效率, 另一方面又要避免同一目的节点的工作通道和保护通道或者恢复通道共享资源以提高业务抵御故障的能力。针对如何高效地为光组播业务提供保护和恢复的技术问题, 应该建立相应的解决方案。

发明内容

本发明所要解决的技术问题在于, 提出一种自动交换光网络中组播业务的保护和恢复方法, 在自动交换光网络中为组播业务提供保护和恢复通道的方案, 实现对组播业务的保护功能。

一种在 ASON 网络中为组播业务提供保护和恢复通道的方法, 包括如下步骤:

(1) 为组播业务中各个目标节点的工作连接计算路由, 计算中后建的连接可共享已建的连接资源;

(2) 根据工作连接所需的保护等级, 为该工作连接确定要建立的保护连接或恢复连接,

其中, 在为所述保护连接或恢复连接计算路由时, 是在所要建立的保护连接或恢复连接不能与其对应的需要保护的工作连接使用风险共享资源的限制条件下, 进行计算, 并最终生成带有保护和恢复通道的组播树的生成组播树生成。

所述的方法, 进一步还可以包括:

(3) 对经步骤(2)所生成的组播树进行重路由优化, 获取路由优化的组播树。

其中, 所述的步骤(1)包括:

(1-1) 在包含 N 条工作连接的组播业务中, 为源节点至目的节点 $d_1, d_2 \dots d_k$ 的工作连接 $W_1, W_2 \dots W_k$ 计算路由, 形成一个从组播业务源节点至 k 个目的节点的组播树 T_k , 其中 $k < N$;

(1-2) 为源节点至第 $k+1$ 个目的节点 d_{k+1} 的工作连接 W_{k+1} 计算路由时, 共享使用组播数 T_k 上的资源, 采用节点 d_{k+1} 至组播树 T_k 的最短路径来为工作连接 W_{k+1} 计算路由, 形成 N 条工作连接的组播树 T_N , 其中 $k+1 \leq N$ 。

所述步骤 (2) 中的所述保护等级包括不需保护、或需要保护连接、或需要恢复连接、或同时需要保护与恢复连接。

所述步骤 (2) 中所述所要建立的保护连接或恢复连接不能与其对应的需要保护的工作连接使用同一风险共享资源的限制条件, 是指:

如果要求工作连接和保护连接或恢复预留连接是节点无关的, 则两者不能经过同一个节点;

如果要求工作连接和保护连接或恢复预留连接是链路无关的, 则两者不能经过相同的链路。

所述限制条件为在基于约束条件的最短路算法中, 将需要保护的工作连接设置为对应保护或恢复连接必须避开的约束路由。

所述步骤 (3) 包括:

(3-1) 从全部目的节点中任意选择一个目的节点, 删除通向该目的节点的工作连接、和/或保护连接、和/或恢复预留连接并释放它们所独占的资源;

(3-2) 重新为步骤 (3-1) 中经删除操作后的所选目的节点计算工作连接路径, 此计算过程中共享已存在的组播树上的资源;

(3-3) 根据所述目的节点的保护等级, 或不处理、或为该目的节点计算保护连接路径、和/或为该节点计算恢复预留连接路径, 此计算过程中共享已存在的组播树上的资源, 但需要依据所述限制条件避开对应的工作连接路径;

(3-4) 比较重路由前后的组播树, 如果重路由之后的组播树优于重路由之前的组播树, 则保留此次重路由操作, 否则, 抛弃此次重路由操作, 恢

复该组播树到重路由之前的状态；

(3-5) 在业务要求的路由计算时间内，重复步骤(3-1)至步骤(3-4)若干次直到超时为止，并保留最终重路由操作后的组播树。

所述步骤(3-1)中的所述独占的资源是只有该目的节点的连接使用而组播树中其它连接都没有使用的资源。

所述步骤(3)中进行重路由计算的算法是模拟退火算法、或遗传算法。

应用本发明所述方法，既能实现通向不同的目的节点的光通道之间共享资源以提高资源的利用效率，又可以使用较少的网络资源完成对组播业务的保护功能，使得那些配置了保护通道或者恢复预留通道的组播业务可以抵抗网络中的任意单点故障，即如果网络中只有一个节点或者链路发生故障，则所有受影响的被保护的工作通道都能够切换到各自的保护或者恢复通道上。

附图说明

图1是根据本发明实施例为组播业务计算路由的流程图；

图2是本发明实施例中初始的ASON网络拓扑结构图；

图3是按照一定的顺序为图2中各条连接计算路由后获得的组播树示意图；

图4是对图3所建立的组播树结构进行重路由优化后网络示意图。

具体实施方式

下面结合具体实施例及附图对本发明进行详细描述。

本发明的核心在于为组播业务的工作连接、保护连接、恢复预留连接分配路由时所采用的路由计算方案，对这个路由计算方案的要求是既要使得通向不同的目的节点的光通道之间共享资源以提高资源的利用效率，另一方面又要避免通往同一目的节点(或者使用风险共享资源)的工作通道和保护通道或恢复预留通道共享资源以提高业务抵抗单点故障的能力。

因此，本发明中，在ASON网络中为组播业务提供保护和恢复通道时，

其路由计算过程分以下几个步骤进行:

(1) 按照任意顺序逐次为组播业务中各个目的节点的工作连接计算路由。

假设组播业务中包含 N 条工作连接, 它们分别是 W_1 、 W_2 、... 和 W_N , 按照任意顺序计算这 N 条连接的路由, 后面的连接可以共享使用前面连接的资源。假设我们已经为 W_1 、 W_2 、... 到 W_K 这 K 条连接计算了路由, 这 K 条路径形成了一个从业务的源节点到 K 个目的节点的组播树, 记为 T_K , 接下来我们需要计算从源节点到第 $K+1$ 个目的节点 d_{K+1} 的工作连接 W_{K+1} 路径。由于工作连接 W_{K+1} 可以共享使用组播树 T_K 上的资源, 因此从源节点到目的节点 d_{K+1} 的最短路径问题实际上转换为从节点 d_{K+1} 到组播树 T_K 的最短路问题。这个问题只是图论中的一个简单问题, Dijkstra 算法稍作调整即可用来处理这个问题。

(2) 按照任意顺序逐次为组播业务中需要提供保护的工作连接创建保护连接或者恢复预留连接。

组播业务中包含 N 条工作连接, 对于其中任意一条工作连接, 根据所需要的保护等级的不同有可能需要为它建立保护连接或者恢复连接。在为保护或者恢复连接计算路由时, 这些连接也可以共享使用别的连接的资源, 但多加了一个限制条件, 那就是它们和自己所保护的工作连接不能使用风险共享资源 (如果要求工作和保护是节点无关的, 则它们不能经过同一个节点; 如果要求工作和保护是链路无关的, 则它们不能经过相同的链路)。这个限制条件可以通过在 CSPF (基于约束条件的最短路算法) 中把对应的工作连接设置为必须避开的约束路由来实现。

(3) 重路由优化。

在前面两个步骤 (1)、(2) 中, 我们已经计算出了满足用户需求的从源节点到各个目的节点的组播树, 这个组播树是通过在前面两个步骤中不断的向已有组播路径中添加叶子节点来得到的, 生成的组播树的结构与步骤 (1) 和步骤 (2) 中建立工作连接、保护连接或者恢复预留连接的顺序有关, 因此计算出的组播树可能并不是最优的。

接下来需要利用重路由的方法对它来进行优化，重路由优化方法包括如下步骤：

(a) 从N个目的节点中任意选择一个节点，删除通向这个节点的所有工作连接、保护连接和恢复预留连接并且释放掉它们所独占的资源（所谓独占，意思是说只有到达此目的节点的连接使用了此资源，而组播树中其他连接都没有使用这些资源）。

(b) 重新为在上一步删除的目的节点计算工作连接路径，计算过程中可以共享已经存在的组播树上的资源。同时根据此目的节点保护等级的不同，还可能需要为它计算保护或者恢复预留路径，保护或者恢复路径可以共享已有组播树的资源但是需要避开工作连接路径。

(c) 比较重路由前后的组播树，如果重路由之后的组播树要优于重路由之前的组播树，则保留此次重路由操作。否则，抛弃此次重路由操作，恢复组播树到重路由之前的状态。

根据具体业务对路由计算时间要求的不同，重复上述操作若干次（如果此组播业务对路由计算运行时间要求不高，可以重复计算多次，甚至使用一些启发式算法，如模拟退火算法或者遗传算法，来优化生成的组播树）。

如图1所示，是本发明中为组播业务计算路由的流程图，包括如下步骤：

步骤101：按照任意顺序逐次为组播业务中各个目标节点的工作连接计算路由；

步骤102：按照任意顺序逐次为组播业务中需要提供保护的工作连接创建保护连接或者恢复预留连接；

步骤103：判断重路由优化操作是否已经执行了足够多次数，组播树是否达到最优，如果是，则转向步骤109，如果不是，则进入步骤104；

步骤104：从目的节点中任意选择一个节点，删除通向这个节点的所有工作连接、保护连接和恢复预留连接并且释放掉它们所独占的资源；

步骤105：为步骤104中删除的目的节点计算工作连接、保护连接和恢复预留连接（如果存在）的路径；

步骤106：判断经过此次重路由操作之后，组播树结构是否得到了优化，

如果是则进入步骤 107, 如果否, 则进入步骤 108;

步骤 107: 保留此次重路由操作结果, 并返回步骤 103;

步骤 108: 抛弃此次重路由操作结果, 恢复到本次重路由操作之前的组播树结构, 并返回步骤 103;

步骤 109: 如存在硬件限制, 则根据具体网络硬件设备的限制对组播树路径进行调整。关于硬件限制, 其对硬件的约束条件可以隐含在前面两步骤 101、102 中, 根据资源选取路由, 即由实际资源可用反映硬件的约束条件。

实施例 1:

假设网络拓扑是图 2 所示的一具有 6 个节点的网络, 现在需要在此网络之上建立一个组播业务, 业务的源节点是节点 1, 目的节点是节点 6 和节点 5。为了描述方便, 我们称从源节点 1 到目的节点 6 的连接为连接 1, 从源节点 1 到目的节点 5 的连接称为连接 2。假设按照用户的需求, 我们需要为连接 1 建立保护通道, 而连接 2 不需要建立保护。

按照本发明所述的方法, 为这个组播业务进行路由计算的过程如下:

(1) 按照任意顺序逐次为组播业务中各个目的节点的工作连接计算路由;

假设我们按照先目的节点 6 后目的节点 5 的顺序为组播业务建立工作连接, 如图 3 所示: 其中图 3(a)是为工作连接 1 计算路由之后的组播连接结构, 图 3(b)是为工作连接 2 计算路由之后的组播树结构。

(2) 按照任意顺序逐次为组播业务中需要提供保护的工作连接创建保护连接或者恢复预留连接;

在本实施例中, 按照保护等级要求, 只需要为工作连接 2 计算保护路径, 并且要求保护连接 2 和工作连接 2 是资源分离的, 据此计算出的连接 2 的保护通道路径, 如图 3(c)所示, 是建立保护连接后的组播树结构。

(3) 对建成的组播树进行重路由优化;

重路由优化的具体过程如图 4 所示, 包括:

(a) 首先，删除掉到达目的节点 6 的所有连接，并释放它所占用的全部资源，如图 4(a)所示。

(b) 然后，重新为由源节点 1 至目的节点 6 的工作连接 1 计算路由，所得计算结果如图 4(b)所示。

(c) 比较重路由操作前后的组播树结构，发现图 4(b)所示的组播树要优于图 3(c)的组播树，因此保留这次重路由操作。

(4) 根据具体网络硬件设备的限制对组播树路径进行调整。

图 4(b)所示的组播树结构，由于不存在硬件限制问题，因此不需要进行调整，图 4(b)就是使用本发明方案最后计算生成的组播树结构。

至此，完成了对组播业务的保护功能，使得那些配置了保护通道或者恢复预留通道的组播业务可以抵抗网络中的任意单点故障，即如果网络中只有一个节点或者链路发生故障，则所有受影响的被保护的工作通道都能够切换到各自的保护或者恢复通道上。

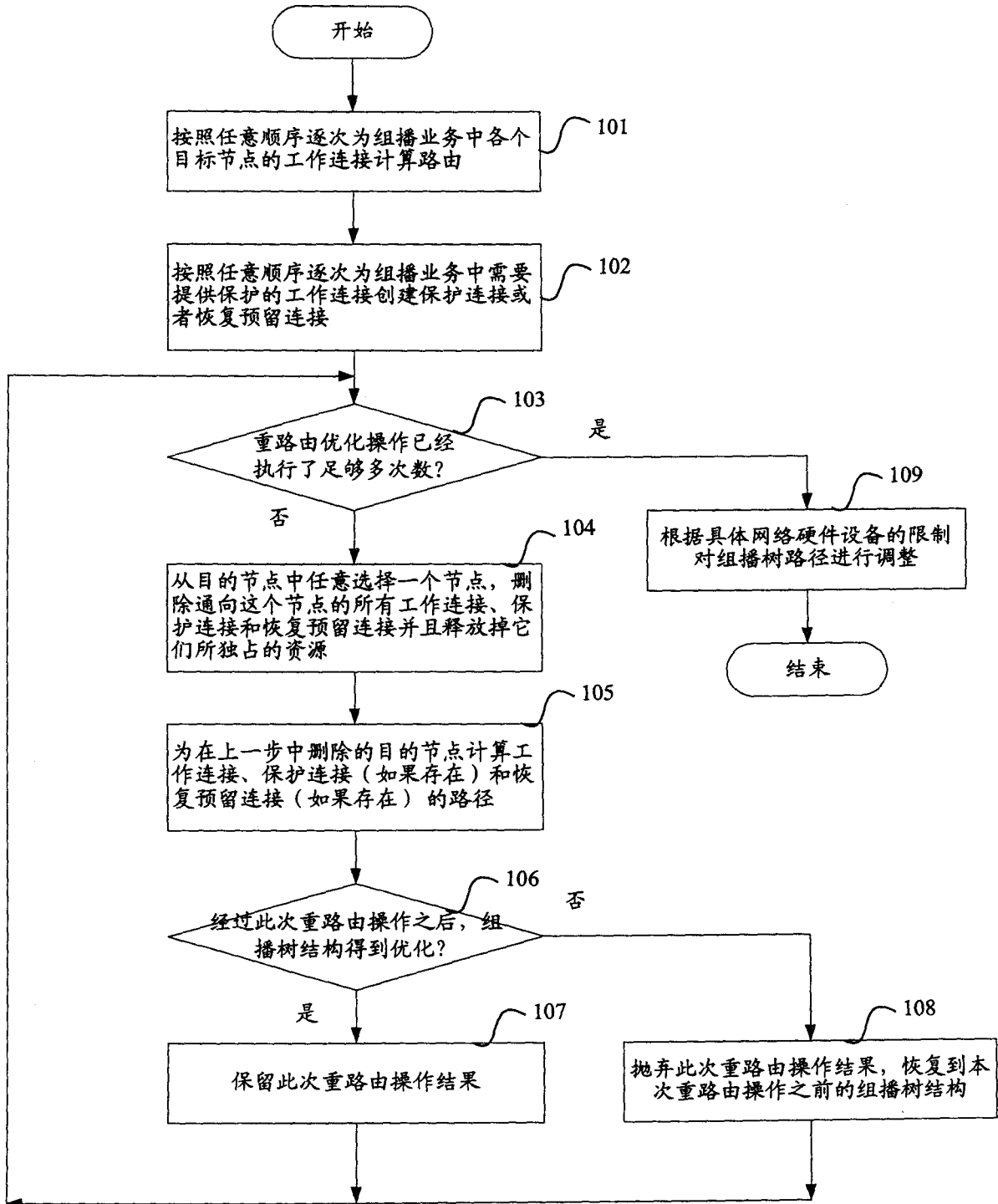


图1

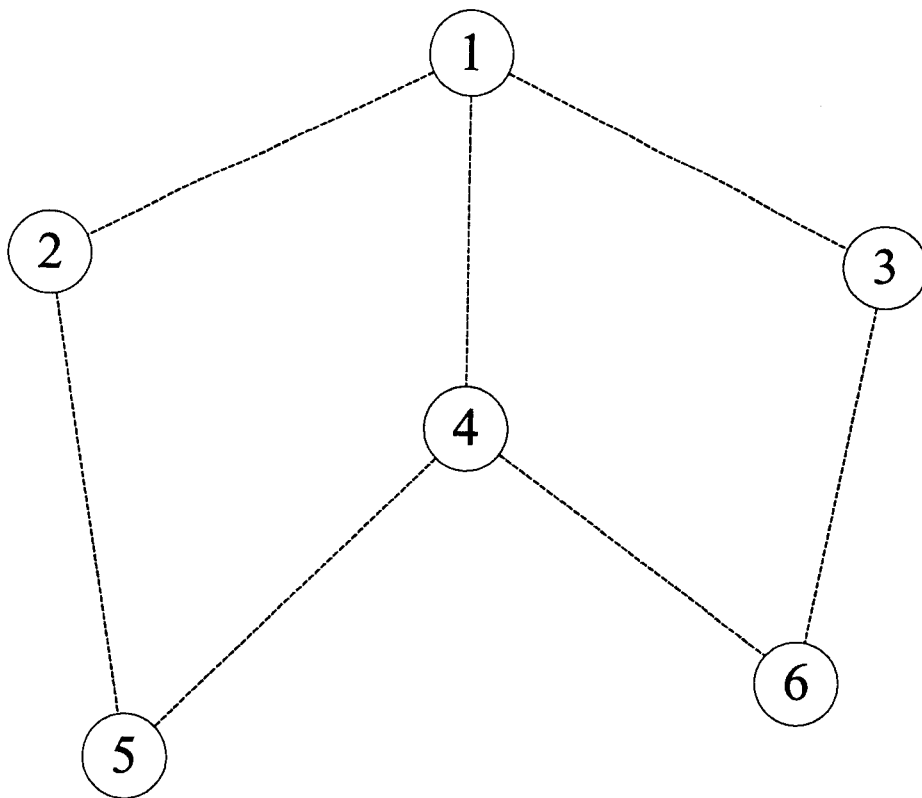


图2

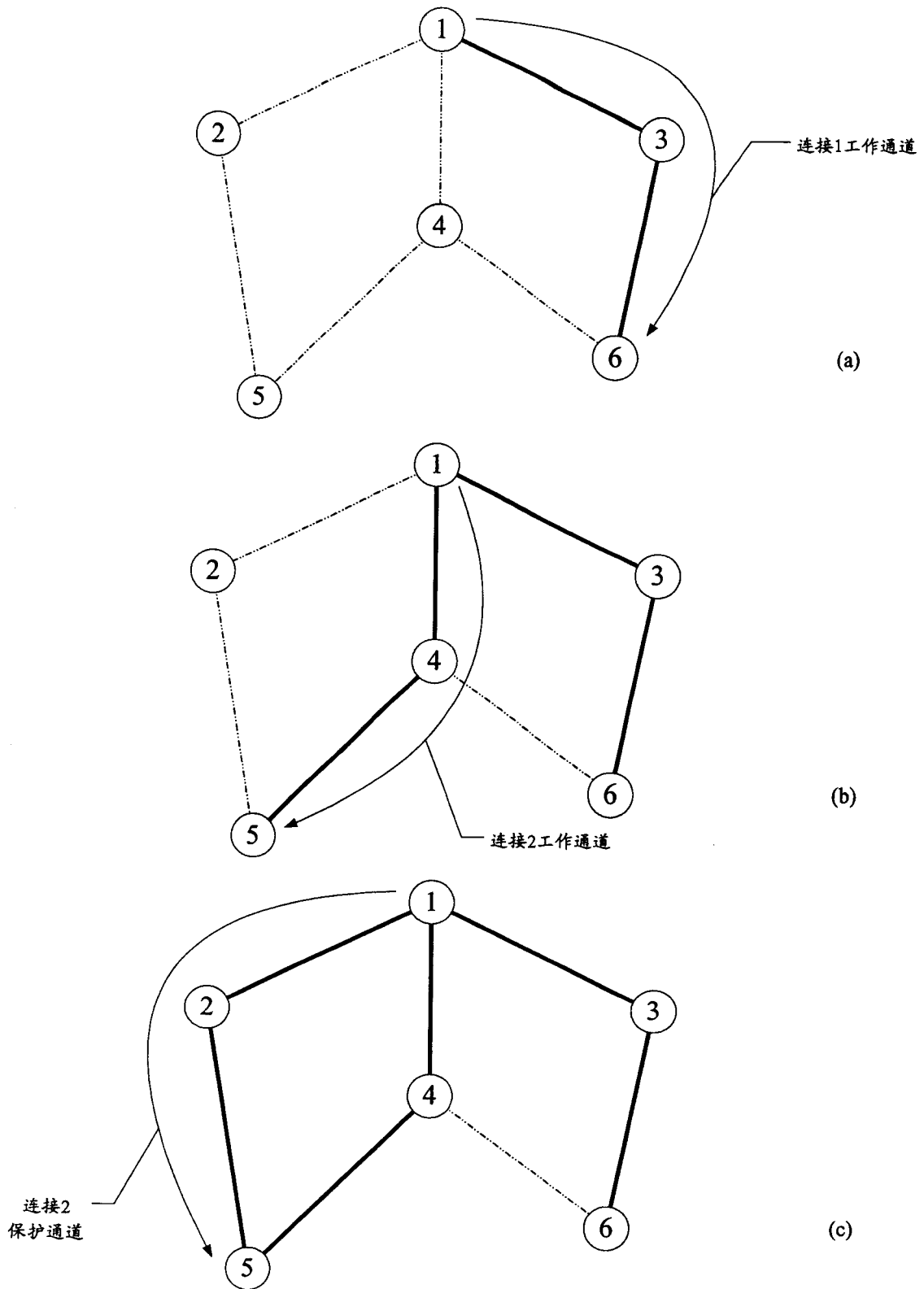


图3

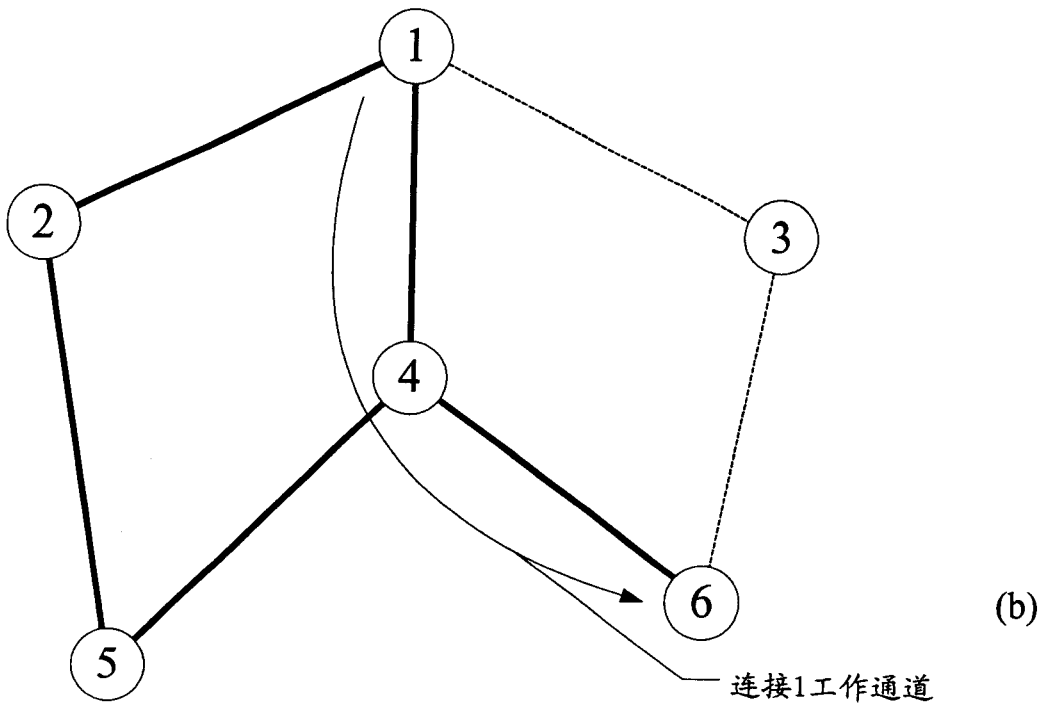
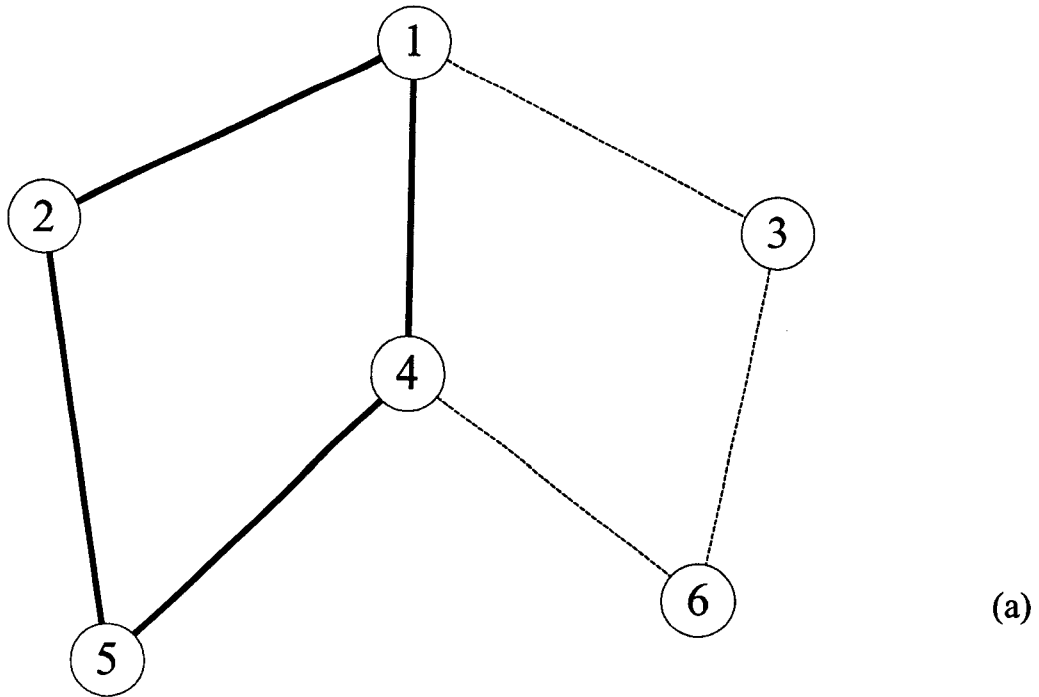


图4