



(12)发明专利

(10)授权公告号 CN 105306354 B

(45)授权公告日 2019.01.22

(21)申请号 201510309005.7

(22)申请日 2011.01.26

(65)同一申请的已公布的文献号

申请公布号 CN 105306354 A

(43)申请公布日 2016.02.03

(30)优先权数据

61/298487 2010.01.26 US

(62)分案原申请数据

201180007263.X 2011.01.26

(73)专利权人 雷·W·桑德斯

地址 美国加利福尼亚州90049,洛杉矶市卡

西亚诺路1037号

(72)发明人 雷·W·桑德斯

(74)专利代理机构 深圳鼎合诚知识产权代理有限公司 44281

代理人 彭愿洁 彭家恩

(51)Int.Cl.

H04L 12/701(2013.01)

H04L 12/709(2013.01)

H04L 29/08(2006.01)

(56)对比文件

CN 1736073 A, 2006.02.15,

US 2007067487 A1, 2007.03.22,

US 6041050 A, 2000.03.21,

审查员 谭菲菲

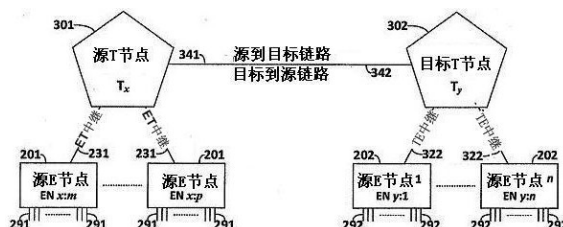
权利要求书3页 说明书34页 附图24页

(54)发明名称

一种不使用显式寻址的网络数据通信方法及系统

(57)摘要

一种不使用显式寻址的网络数据通信方法及系统,其中方法包括:形成包括入口节点和出口节点的E节点网络层;形成包括中转节点的T节点网络层;形成连接域,其中包括入口节点和出口节点之间的第一连接;给第一连接分配第一带宽;接收路径层数据;至少部分地基于对应于第一连接的路径层数据的一部分在路径帧中的位置,使用隐式寻址识别路径层数据的该部分;在E节点网络层,将路径层数据集成若干个超级路径帧;在T节点网络层,进一步将若干个超级路径帧集成若干个T节点层帧;生成T节点层帧中的小单元,其具有对应于第一连接的第一带宽的对应长度;将E节点网络层的第一网络时钟同步到T节点网络层的第二网络时钟。



1. 一种不使用显式寻址的网络数据通信方法,其特征在于,包括:
形成包括若干个入口节点和若干个出口节点的E节点网络层;
形成包括若干个中转节点的T节点网络层,所述若干个中转节点用于不使用显式寻址地将数据从所述E节点网络层的第一入口节点传送到所述E节点网络层的第一出口节点;
形成包括若干个连接的连接域,所述若干个连接包括入口节点和出口节点之间的第一连接;
给所述第一连接分配第一带宽;
从位于路径层的若干个外部设备接收路径层数据;
至少部分地基于对应于所述第一连接的路径层数据的一部分在路径帧中的位置,使用隐式寻址识别所述路径层数据的该部分;
在所述E节点网络层,将所述路径层数据集成若干个超级路径帧;
在所述T节点网络层,进一步将所述若干个超级路径帧集成若干个T节点层帧;
生成T节点层帧中的小单元,所述小单元具有对应于所述第一连接的第一带宽的对应长度;
将所述E节点网络层的第一网络时钟同步到T节点网络层的第二网络时钟。
2. 如权利要求1所述的方法,其特征在于,还包括:
利用所述第一网络时钟到所述第二网络时钟的同步,对将所述若干个超级路径帧集成所述若干个T节点层帧进行计时,从而在每个所述若干个T节点层帧中实现均匀间隔的所述小单元。
3. 如权利要求1所述的方法,其特征在于,还包括:
对于所述连接域,在所述超级路径帧内将所述路径层数据集群成均匀间隔的所述小单元。
4. 如权利要求3所述的方法,其特征在于,还包括:
改变所述若干个T节点层帧中的每一个的对应长度,其中,所改变所述若干个T节点层帧中的每一个的对应长度对应于改变所述若干个连接中的至少一个的对应带宽,所述改变所述若干个T节点层帧中的每一个的对应长度包括改变所述小单元的长度。
5. 如权利要求4所述的方法,其特征在于,所述改变所述若干个T节点层帧中的每一个的对应长度先于改变至少一个连接的对应带宽而发生。
6. 如权利要求1所述的方法,其特征在于,还包括:
在包括所述若干个连接中的每一个的对应带宽的带宽上,以所述第二网络时钟的时钟频率,发送所述若干个T节点层帧,从而实现与所述若干个连接中的每一个的对应带宽的总数对应的最小集合带宽,其中所述最小集合带宽对应于所述若干个T节点层帧的帧率乘以所述第二网络时钟。
7. 如权利要求1所述的方法,其特征在于,所述若干个超级路径帧的一个超级路径帧包括所述出口节点的地址。
8. 如权利要求1所述的方法,其特征在于,根据2的次方数帧特性,所述小单元在所述T节点层帧中基本均匀间隔。
9. 一种用于不使用显式寻址的网络数据通信方法的系统,其特征在于,包括:
一个或多个处理器;和

存储有程序的有形且非瞬时性的计算机存储器,当所述程序由所述一个或多个处理器执行时,所述程序用于不显式寻址地执行网络通信的操作,包括:

形成包括若干个入口节点和若干个出口节点的E节点网络层;

形成包括若干个中转节点的T节点网络层,所述若干个中转节点用于不使用显式寻址地将数据从所述E节点网络层的第一入口节点传送到所述E节点网络层的第一出口节点;

形成包括若干个连接的连接域,所述若干个连接包括入口节点和出口节点之间的第一连接;

给所述第一连接分配第一带宽;

从位于路径层的若干个外部设备接收路径层数据;

至少部分地基于对应于所述第一连接的路径层数据的一部分在路径帧中的位置,使用隐式寻址识别所述路径层数据的该部分;

在所述E节点网络层,将所述路径层数据集成若干个超级路径帧;

在所述T节点网络层,进一步将所述若干个超级路径帧集成若干个T节点层帧;

生成T节点层帧中的小单元,所述小单元具有对应于所述第一连接的第一带宽的对应长度;

将所述E节点网络层的第一网络时钟同步到T节点网络层的第二网络时钟。

10.如权利要求9所述的系统,其特征在于,所述一个或多个处理器还用于执行的操作包括:

利用所述第一网络时钟到所述第二网络时钟的同步,对将所述若干个超级路径帧集成所述若干个T节点层帧进行计时,从而在每个所述若干个T节点层帧中实现均匀间隔的所述小单元。

11.如权利要求9所述的系统,其特征在于,所述一个或多个处理器还用于执行的操作包括:

对于所述连接域,在所述超级路径帧内将所述路径层数据集群成均匀间隔的所述小单元。

12.如权利要求11所述的系统,其特征在于,所述一个或多个处理器还用于执行的操作包括:

改变所述若干个T节点层帧中的每一个的对应长度,其中,所改变所述若干个T节点层帧中的每一个的对应长度对应于改变所述若干个连接中的至少一个的对应带宽,所述改变所述若干个T节点层帧中的每一个的对应长度包括改变所述小单元的长度。

13.如权利要求12所述的系统,其特征在于,先于改变至少一个连接的对应带宽改变所述若干个T节点层帧中的每一个的对应长度。

14.如权利要求9所述的系统,其特征在于,所述一个或多个处理器还用于执行的操作包括:

在包括所述若干个连接中的每一个的对应带宽的带宽上,以所述第二网络时钟的时钟频率,发送所述若干个T节点层帧,从而实现与所述若干个连接中的每一个的对应带宽的总数对应的最小集合带宽,其中所述最小集合带宽对应于所述若干个T节点层帧的帧率乘以所述第二网络时钟。

15.如权利要求9所述的系统,其特征在于,所述若干个超级路径帧的一个超级路径帧

包括所述出口节点的地址。

16. 如权利要求9所述的系统,其特征在于,根据2的次方数帧特性,所述小单元在所述T节点层帧中基本均匀间隔。

一种不使用显式寻址的网络数据通信方法及系统

[0001] 分案申请说明及优先权要求

[0002] 本专利申请是申请日为2011年1月26日,申请号为CN201180007263.X,发明名称为“用于同步网络的装置和方法”的专利申请的分案申请。本专利申请要求名称为“用于同步网络的装置和方法”、申请号为61/298487、申请日为2010年1月26日的美国临时申请的优先权,该临时申请的全部内容通过引用结合于本文中。

技术领域

[0003] 本申请涉及网络和通信控制。

背景技术

[0004] 大型数据网络的高层结构已达成共识,该共识在可预见的未来不太可能改变。该结构已经由能够从一个计算机向另一个计算机转发消息和文件演化为可从全球任何源向任何目标携带任何类型数据进行传递的成熟结构。目前该结构得到一致认可并且存在不止一种实现形式(例如ATM和MPLS),需要找到比现存数据率更高的新方法和装置来低延迟、可靠地传输比特数据。本申请公开了达到这些目标的若干方法和装置,并注重简化、低成本和低功耗。

[0005] 开始于二十世纪七十年代早期,有两种采用数字技术以克服模拟电话和电报局限的途径。一种是建立注重更高质量数字化语音的数字电话网络;另一种是建立以质量为代价的强调可靠性的基于存储和转发分组交换(packet switching)的数字数据网络。分组交换因其灵活性和巨大投资以改善质量并相比于语音网络保持的灵活性而盛行。在数字语音方面的努力已逐渐过时,而分组交换途径由于基于很久以前制定的结构和实现方式而具有一些严重的局限。

[0006] 四种基本网络方法需要从根本上进行检查以重定义相比较于现有网络环境的未来网络能力,这些方法是:

[0007] 1. 数据封装方法

[0008] 2. 数据交换方法

[0009] 3. 分组缓存方法(Packet buffering methods)

[0010] 4. 流量路由方法(Traffic routing methods)。

[0011] 第一个方面,数据封装,描述了将数据分割成置于共享容器中可从一点传输到另一点的单元的方法。数据源可为计算机、电话、照相机、麦克风、传感器,或只要其为数字形式的其他设备。

[0012] 在数字语音网络中,数据封装是很受限制的。这些数字语音网络基于数字时分复用体系(digital time division multiplex hierarchy)中的“八位位组”(octets)。如所实现的,这些网络具有操作数据流的有限能力,这已被现存电话公司以语音为中心的网络所证实。以数据为中心的网络从语音网络中分离演化而来。这些以数据为中心的网络通常将数据封装成包(packets)。分组复用过程(Packet multiplexing)向每个源数据片段添加

信息,识别源和目标位置加上其他信息。添加的信息可包括服务类型、数据格式细节、错误探测和修正,以及诸如网络控制的其他目的。换言之,封装涉及添加“报头”(headers),“选项”(options),“错误控制”(error control),以及其他参数。数据封装扩展至作为分组协议标准方法一部分的协议“层”(layers)。每个层具有名称(例如,链路,网络,传输,等等)。这就要求附加的报头和其他信息。在网络中包含协议层通常会造成长包(longer packets)。因此,这些层经常使用超出传输数据需要的大量带宽开销。

[0013] 在基于分组(packet-based)的网络中,使用报头的数据封装是需要的;但这种方法带来了几个困难。一个是由所需要的包开销(packet overhead)带来的带宽需求。另一个是来自诸如数字化语音等数据率相对较低的源的比特数据填充包缓存(packet buffer)时所需要的延迟。封装声音的过程需要数十毫秒以避免发送短语音包所需要的更大的报头带宽。另外,路由器内需要包缓存来平滑可带来延迟变化和抖动(jitter)的通信流量。另一个困难是,即使包头(packet header)中只有一个比特错误都可导致错误的分发和/或包的丢弃。其后果是服务的中断以及数据丢失,数据丢失可导致延迟及在路由器中的大量处理以重新转发被丢弃的包。按照导致现今网络工作方式的思路进行推理可以引出这样的问题,“将数据封装进包是个很好的想法,但是否有一种途径,能够以小到象一个比特位的短脉冲转发封在包内的数据而不依靠在每个交换节点读取每个包的报头、标签(label)或标记(tag)呢?”读取标记比读取包头快很多,但可做改进。本申请公开了使用短时分帧(short time-division frames)的简单电路交换以克服这些限制的方法。

[0014] 第二个方面,交换方法,在语音网络中是颇为简单的,但是其封装方法限制了在数据网络中的使用。在数据网络中,使用异步交换方法(asynchronous switching methods)从一个目标到另一个目标转发包对于通过统计时分复用(只要基本过供应带宽存在)保证信息传递是个突破。这些方法保证网络中从源到目标在传输设施故障或在危险操作条件下的传输。它们还保证比现存通过静态时分复用更有效率的对可用带宽的使用。但是,四十年前常见的情形现今并不常见。将统计复用和静态时分复用进行比较是“证明”分组超过电路交换好处的流行方式。这忽视了若正确设计时分复用可极端动态的前景。这些设计将在本申请中呈现。

[0015] 异步交换解决了工程化用于语音服务的同步网络的限制。不幸的是,该结论现在已变成电路交换对于当前和将来的网络不可用的传统工业知识的僵化判断。该感知是从电话工业界的该技术缺乏灵活性而得来的,而并非从本申请公开的电路交换的现实可能性得来的。

[0016] 过去,时分复用在几个方面受到限制。一个是将高数据率连接以整数形式分割为若干个低数据率连接的自然限制。另一个是为电话建立的电路连接在电话的持续期间保持连接状态;而现存没有用于处理突发数据的将电路分段(circuit epoch)限定到毫秒、微秒或纳秒的机制。其三,现存没有迅速改变分配给连接的带宽的方法,对于多媒体和视频应用而言这是达到有效使用可用带宽的强制要求。

[0017] 作为克服诸如电话网络等“一次性”系统(one-off system)的限制的解决方案的一部分,将数据放置于“包”中出现了。数据包对于可预见的未来将一直是一项伟大的发明。互联网协议真正成为了所有用户通信的通用语(lingua franca)。在这种情况下,使用口号“互联网无所不在(IP everywhere)”来说明网络边界是一个好想法。尽管它是否该应用于

未来的网络内部是非常值得怀疑的。本申请的目的是提供克服现有实践缺陷的用于以IP为中心的互联网和其他网络的下一代内部网络。

[0018] 用户数据到达的随意性(例如,随机的)并不必然导致需要随机网络解决方案。克服具有扩展覆盖的异步分组网络的随机特性的复杂性就是需要另一个解决方案的充分证据。这引出问题,“当现实可为简单而有效时,为什么通过诸如“电路仿真”的想法来使用更高的网络复杂性呢?”现在网络可低成本地使用现今技术的巨大潜力,尤其是网络计时、存储,以及计算机处理。由如下问题出发现代技术可克服异步网络的缺点:“有这么多的快速发展的技术,我们难道不能建立低成本地克服现今复杂性又提高可扩展性的同步节点网络吗?”。本申请提供方法以减少网络复杂性并提高交换性能。

[0019] 现有网络的第三个方面是将包缓存置于当今网络之中。当首次发明互联网时(如美国国防部中的ARPANET),服务中的最小计算机是微机。不存在具有大型处理能力的商用微型计算机。商用输入/输出设备是没有有效处理器和储存能力的终端。当时用于处理“突发”数据的最好的可能方法是在微机中接收所有到来的数据,那时,当这些数据到来时没有足够的通信带宽去发送它们。最初的方案,正如消息交换存储和转发处理器,将数据存储在先进先出(FIFO)缓存中。若缓存太小不能容纳所有进来的数据流,有选择地丢弃包成为选择用于解决拥塞问题的方法。对于这个结果,成本是首要驱动参数。计算机数据存储是非常昂贵的,当计算机数据存储不很昂贵时,包缓存也开始增长。

[0020] 内部网络缓存加剧了现有网络的随机性,而且大大增加了网络复杂性、抖动、延迟、丢包率以及成本。其周边效应包括需要流量整形(traffic shaping)和管辖消费者对带宽的使用。内部网络缓存往往是糟糕服务的肇事者。当流量强度高时,存在超过路由器或外向连接可处理的流量时它们因拥塞而受损害。作为补救,丢包时需要数据源重发被丢弃的数据包。在流量强度是拥塞的首要原因时这只会加剧拥塞。今天,大型数据存储单元一半导体以及硬盘驱动器一很廉价了。它们可存在于输入端口,不再丢弃包。即使这样,原来导致拥塞的方法仍然在现有的网络结构中盛行。当然,很难说网络内缓存增强了服务质量或性能;它们仅仅导致不可预见性能的随机“最佳效果”。这些缓冲问题近来被命名为“缓存膨胀”(buffer bloat),并且认为是网络延迟和某些高时延的原因。问题:“我们仍然使用同样的方法,因为‘那是我们总是这么做的方式’?或存在需要将缓存放置在网络的数据传输端口内的操作和成本原因吗?”。现有网络的不足和延迟可通过本申请提供的方法改进,通过仅接受在入口的能够确保通过网络传输的包来避免必须丢弃包。

[0021] 分组和电路交换网络的第四个方面是它们注重连接的动态路由。分组交换网络注重通过边界网关(border gateways)在主要的平面网络中进行“自治系统”的互连。电路交换(语音)网络注重大(4类和5类)交换,最初采用五层体系,随着时间的推移减少到仅仅两层。网络结构需要使用连接的动态路由。没有简单的备选方案去克服互连上亿(现在上升到数百亿)的可连接实体的所谓 n 平方问题。[如果 n 是可连接实体的数目,对于每个实体必有 $n-1$ 个可用连接允许其连接到所有其他实体。需要的连接数目是 $n \times (n-1)$,当 n 值很大时,接近于 n 平方,这就是网络路由问题的一个著名的源头]。这提出了这个问题,“难道没有可能通过大量分散的网络控制以对每个路径提供多个当前路由将 n 平方化敌为友?”。

[0022] 本申请的总体背景包括已授予本申请发明人的涉及同步适应基础设施网络(SAIN)的美国专利5,502,723和6,041,050。本申请通过引用将这些专利整体结合于此。这

些专利描述了具有在传输链路中快速建立通信信道的独特能力的装置和复用方法。与其他已知的能力不同,该方法能快速设定连接和释放,以及快速改变连接带宽。

[0023] 业界需要在底层交换和发送数据的技术,该技术注重在上述四个方面以及网络的其他方面克服现有技术的局限。

发明内容

[0024] 在一个实施例中,提供了从网络入口节点的网络路由的方法。该方法包括从若干个外部设备接收网络通信,每个网络通信指向目标节点。该方法还可包括将网络通信集成若干个路径,每个路径包括关联单个目标节点并包括若干个路径帧的网络通信的一部分,路径帧与单个目标节点同步并包括若干个由路径数据组成的路径小单元(path cellets)。该方法还可包括进一步将网络通信集成若干个超级路径(superpath),每个超级路径包括关联单个中间节点并包括若干个超级路径帧的网络通信的一部分,超级路径帧与中间节点同步并包括若干个由超级路径数据组成的超级路径小单元(superpath cellets)。该方法还可包括传输该若干个超级路径到传输节点。该方法还可包括可动态调整大小的路径小单元和超级路径小单元。该方法还可包括具有不同尺寸大小的路径小单元和超级路径小单元。该方法还可包括单独或同时大小为一个比特的路径小单元和超级路径小单元。该方法可包括经过若干个传输路线传输到中间节点的若干个超级路径。该方法可包括通过传输表示小单元大小的控制矢量的小单元,以及小单元大小可动态设定的帧。

[0025] 在一个实施例中,提供了在入口节点处网络数据路由的方法,包括从外部设备接收网络通信,该网络通信指定出口节点。该方法可通过传递控制矢量到出口节点来和出口节点建立成对连接,该连接包括带宽。该方法还可包括与出口节点保持成对连接,该保持包括根据接收的网络通信调整带宽。该方法还可包括从外部设备将网络通信集合到集合数据流中,该集合数据流包括隐式寻址帧,该隐式寻址帧同步到出口节点。该方法包括传递到出口节点的控制矢量,该控制矢量包括表示分解该隐式寻址帧的数据,以致于由成对连接在出口节点执行集合数据流的分解。该方法可包括配对连接可以带宽为零。该方法可包括网络通信包含包,每个包包括寻址报头,并且其中集合步骤包括从网络通信中剥离该寻址报头;且其中控制矢量包括在出口节点重新组成寻址报头的信息。该方法包括集合数据流是若干个集合数据流中的一个,每个集合数据流关联出口节点,并且该方法还包括将若干个数据流集合到中间数据流的步骤,该中间数据流关联中间数据节点,其中该中间数据节点是关联若干个数据流的出口节点的父节点,并且建立和保持与中间数据节点的成对连接,该与中间数据节点的成对连接包括表示分解中间数据流的信息。

[0026] 在一个实施例中,网络中转节点包括若干个链路,每个链路对应于若干个网络入口节点的一个网络入口节点,该若干个链路中的每个转发原数据流到网络中转节点,该源数据流包括若干个初始超级路径的集合,每个初始超级路径关联若干个目标中转节点中的一个目标中转节点,并且每个初始超级路径包括从网络入口节点到若干个目标出口节点的路径集合。该网络中转节点还可包括若干个源数据流分解交换,该分解交换从若干个到网络入口节点链路中的一个链路接收源数据流,每个源数据流分解交换用于从每个源数据流中分解若干个初始超级路径。该网络中转节点还可包括若干个交叉连接(crossconnect)交换,每个交叉连接交换对应于若干个目标中转节点中的唯一目标中转节点,每个交叉连接

用于接收关联对应于交叉连接交换的目标中转节点的初始超级路径,分解初始超级路径以恢复从网络入口节点到目标出口节点的路径,并集合从网络入口节点到目标出口节点的路径以形成目标超级路径,每个目标超级路径包括从若干个入口节点到关联唯一目标中转节点的单个目标节点。该网络中转节点可包括若干个转发节点集合交换,每个转发节点集合交换用于集合关联一个目标中转节点的若干个目标超级路径以形成中转数据流。该网络中转节点可包括用于转发每个中转数据流到若干个目标中转节点的若干个外向链路(outbound link)。网络中转节点可具有接收来自于若干个入口节点的控制矢量的若干个链路,该控制矢量包括表明分解信息的数据。网络中转节点可具有进一步包括表明来自于入口节点的带宽调整请求信息的控制矢量。网络中转节点可向若干个网络入口节点提供同步信息以将中转节点和若干个网络入口节点进行同步。网络中转节点通过若干个外向链路中的不止一个传输若干个中转数据流中的至少一个。网络中转节点可通过同步、隐式寻址变换算法执行集合和分解。

附图说明

- [0027] 图1显示根据一个实施例的底层网络;
- [0028] 图2显示根据一个实施例的分层核心网络层(hierarchical core network tiers)的图;
- [0029] 图3a和图3b显示现有技术的蜂窝时隙复用器(cell slot multiplexer)图;
- [0030] 图3c显示现有技术的蜂窝时隙选择器(cell slot selector)图
- [0031] 图4显示根据一个实施例的导致等距分布时域(equally spaced Time Domain)小单元的连接域(Connection Domain);
- [0032] 图5显示用于非2的次方数(non-power-of-two)长度帧示例的时域结果;
- [0033] 图6显示根据一个实施例的用于每帧非2的次方数目的小单元的时域结果;
- [0034] 图7显示支持E节点到E节点路径的两层网络结构;
- [0035] 图8显示具有20个T节点和40个双工中继(duplex trunks)的城域网(Metropolitan Network);
- [0036] 图9a和图9b显示根据一个实施例的具有帧时钟发生器(Frame Clock Generator)的交换栈选择器(Switch Stack Selector)的图;
- [0037] 图10a显示根据一个实施例的用于改变交换时钟率(switch clock rates)和帧大小的装置;
- [0038] 图10b显示根据一个实施例的用于改变交换时钟流量率(switch clock flow rate)的流程图;
- [0039] 图11显示根据一个实施例的用于改变带宽寄存器的装置;
- [0040] 图12显示根据一个实施例的改变连接域带宽范围的装置;
- [0041] 图13显示根据一个实施例的基本集合/分解交换对;
- [0042] 图14显示集合/分解交换对的一个实施例;
- [0043] 图15显示根据一个实施例定义设置新连接流程的流程图;
- [0044] 图16a、16b、16c显示根据实施例的E节点和其T节点父亲的各种配置;
- [0045] 图17显示根据一个实施例的连接域中的源端(source-end)集合方法;

- [0046] 图18显示根据一个实施例的连接域中的目标端(destination-end)分解方法;
- [0047] 图19显示源E节点集合/分解路径交换的实施例;
- [0048] 图20显示目标E节点集合/分解路径交换的实施例;
- [0049] 图21显示根据一个实施例的连接源E节点和目标E节点的1一个路径层交换对(Path Level Switch Pair)、2个第2层对(Level 2 Pairs)以及3个第3层对(Level 3 Pairs)的系统图;
- [0050] 图22a显示根据一个实施例的集合连接到目标E节点的所有路径的源E节点集合交换;
- [0051] 图22b显示根据一个实施例的分解连接到源E节点的所有路径的目标E节点分解交换;
- [0052] 图23显示交叉交换的一个实施例,该交换导致一个父T节点(T06)的每个子E节点连接到第二T节点(T11)的所有子E节点;
- [0053] 图24显示3级源T节点示例,该示例使用交叉连接交换初始化重定向499个路径中的25个2级集合中的每个,使其从1对多到多对1;
- [0054] 图25显示目标T节点11的示例,该示例使用交叉连接完成从20个T节点到T节点11的25个子E节点的多对1映射;
- [0055] 图26a、26b、26c和26d显示集合层的示例;
- [0056] 图27a、27b、27c和27d显示根据实施例的具有附随时域的各种帧和子帧连接域。

具体实施方式

[0057] 参考附图,具体实施方式公开了实现本系统和本申请方法的几个示例性的优选实施例。描述于此的设备和方法相关的本领域具有普通技能的人将理解或懂得某些实施例的特征。为了简明和易读,这样的“本领域具有普通技能的人”是指“熟练技工”(skilled artisan)。

[0058] 熟练技工,根据本公开,将懂得描述于此的某些组件可方便地使用计算机软件、硬件、固件(firmware)或软件、硬件和固件的组合实现。虽然网络节点将典型地在硬件或固件中实现控制元件,但任何可由使用硬件实现的控制逻辑可通过未描述于此的使用硬件、软件或固件的各种组合加以实现。例如,通用计算机上的固件或软件可完全实现这样的控制。

[0059] 熟练技工,根据本公开,可分割或组合描述于此的模块。例如,根据本公开,熟练技工将懂得单个组件可提供网络中多个组件的功能。相反地,任何一个组件可分割成多个组件。

[0060] 描述于此的前述实施例或这些实施例的其他变形可由熟练技工在不背离本发明的情况下完成。根据这个理解,因此描述的实施例是示例性的,并且本发明不限于描述的实施例,某些实施例结合参考附图描述于下。

[0061] 同步适应性基础结构(SAIN)的顶层概述

[0062] 同步适应性基础结构(SAIN)是一种数字网络技术,其能在不到一个毫秒的时间内设立和拆开电路连接。该技术是本申请的主题。该技术如图1所示使得用户数据能从路由数据传输中亮线分离(bright line separation)。其使用同步交换节点,其可将现有实践中使用的服务指标(Service metrics)的质量减少到单个参数一时延。

[0063] 在SAIN中转节点处转发的延迟可为没有大抖动和延迟变化的少许纳秒。SAIN方法还可大幅度减少系统功率消耗,因为该技术使用短如一个比特的数据片段并将时分交换连接到半静态内存印象图(memory maps)。在网络内不存在用于路由的数据报头,并且没有产生抖动的包缓存。所有的包缓存仅存在于连接用户终端设备的入口和出口节点处。

[0064] 虽然公开于此的该技术可用于多个网络体系和结构中,但本申请着重于如图2底部两层所示的两层结构,就是:入口/出口节点层,即,E节点层20,以及中转节点层,即,T节点层30。图2顶部的第三层显示交换节点层,即可提供与其他两层SAIN网络互联的X节点层40。

[0065] E节点是SAIN网络到用户设备的接口层节点(interface level node)。一组E节点可连接到执行到目标T节点的分级路由选择(hierarchical routing)的中转节点("T-Node")。E节点包含到用户世界的参数化用户接口(PUIs)和本申请公开的到SAIN传输世界的转换。每个E节点交换可设定和管理到网络中所有其他E节点路径中的连接。其通过发送数据到父T节点和从父T节点接收集合数据来完成,这个父T节点可担任源T节点和目标T节点。每个源T节点转发来自于子源E节点的连接集合,这些子源E节点分离于指定到用于向T节点的子背转发的每个T节点的超级路径。

[0066] 父源T节点将其子源E节点从单源E节点(one-source-E节点)到所有目标E节点(all-destination-E-Nodes)的外观重集合成为一个目标E节点从所有源E节点的外观。它转发其重集合到网络中适当的每个T节点。如下详述,该过程包括如下步骤:

[0067] 1. 源E节点集合引入流量成为指定到其他所有(目标)E节点的路径;

[0068] 2. 该源E节点集合其路径成为指定到目标T节点的超级路径,该超级路径是每个目标T节点的子E节点路径的集合;

[0069] 3. 该源E节点集合超级路径成为更高级别的超级路径,其能转发源E节点产生的到其父源T节点的所有流量;

[0070] 4. 该源T节点重排1对多E节点源结构成为多对1目标结构;

[0071] 5. 该源T节点发送目标结构到每个目标T节点;

[0072] 6. 目标T节点分解该结构并发送所有源E节点流量到每个目标E节点;

[0073] 7. 目标E节点分解来自于源E节点的该流量成为用于传递流量到用户的路径。该过程是每个源E节点集合方法的分解逆方法。

[0074] 每个目标T节点转发每个源集合到子E节点目标。其结果是网络使用预建立路由来去除对动态逐跳(hop-by-hop)路由的需求。SAIN不需要替代现存网络,单可通过与现存网络互联克服现存的限制。另外,其提供低成本和在绿地和升级应用中要求的低功耗,特别是那些涉及光纤传输的情形。

[0075] 现有技术的作用

[0076] 图3a、图3b和图3c显示颁布给本发明人的美国专利5,502,723和6,041,050中的现有技术。这些专利中呈现的复用器/解复用器(demultiplexer)技术能快速建立连接,并且一旦连接存在,还能快速改变其带宽或释放其带宽。本申请可使用其他未被所引用专利覆盖的其他技术,只要该技术具有相同或相似的功能。名称:本申请中的“交换栈选择器”(Switch Stack Selector)替代了显示的现有技术图中的名称“蜂窝时隙选择器”(Cell Slot Selector)。本申请的说明书中未使用这些图中的数字。

[0077] SAIN大体结构

[0078] 图1显示SAIN网络的顶层组件。该网络的主要特征在于作为本申请主要议题的SAIN底层网络100。就是将用户数据与其传输分离亮线之中的网络部分。在亮线之外是用户主机、终端、服务器或在入口或网络末端出口处的附属于用户接口连接290的网络101。

[0079] 底层网络100网络的每端的末端部分包括连接到协议翻译器214的在线参数化用户接口210。协议翻译器214将任何用户协议,包括具体地,以太网,转换成为序列比特流,其中引入的包头可被小到1或2个字节的压缩报头的连接识别符所取代。

[0080] 在底层网络100中,有取名为小单元的数据单元。一个小单元可为一个数据片段,该数据片段长度依赖于其环境为数个比特。对于给定的数据链路其为固定大小的数据元(data element),但可依不同的链路变化。一个小单元可小到1个比特,或大到需要转发高带宽集合的程度。

[0081] 基于分组(Packet-based)的寻址命名为显式寻址。显示寻址表现为基于分组的数据封装。包头包括源和目标地址,其形式取决于具体的协议,如以太网和互联网(IP)协议。

[0082] 小单元可存在于周期分时复用帧内。这些帧可取决于网络中转发数据的类型在期间从纳秒到秒变化。帧是小单元的集合,小单元的数目规定了帧的大小。对于给定的连接,其小单元的位置确定了其身份(identification)。

[0083] 该从其他连接中识别某个连接的方法命名为隐式寻址。隐式寻址能通过其在时分或空分帧中的位置对诸如小单元的数据源进行寻址。例如,时分帧的第三小单元也许属于从点A到点B的连接,而第四小单元可属于从点C到点D的连接。

[0084] 隐式寻址可减少使用包头的显示寻址方法所要求的带宽数量,而且特别适合于可变速时分交换应用。该方法与使用包头识别包数据归属连接相比很具鲁棒性(robust)。隐式寻址是世代相传的电话网络中使用的基本寻址方法。

[0085] 电话网络和SAIN网络差别的主要观点在于SAIN网络在网络的不同部分使用不同长度小单元的能力。SAIN复用算法可对于存在于SAIN底层网络100以外的数据协议提供传输方法。

[0086] 差别的第二种主要观点在于显式寻址依赖于将包无误地发送到其正确的目标。通过隐式寻址,设定连接的错误可能性非常小。更为重要的是,一旦设立,在数据初相持续时间内没有所要求的进一步的寻址消息,该持续时间可从亚微秒到年变化。任何可能发生的数据错误与设定连接本身无关。这就是电话网络具有高质量服务如此可靠的原因。SAIN网络有相同的可靠性。一旦连接设定,为了在包流(packet flow)持续时间发送包到连接于SAIN交换的某个位置的已分配FIFO缓存,只在网络边沿读取包头。若消费者想要专线服务(private line service),连接一旦设定,可存在于消费者付款的任长时间。这不需要特殊的操纵。

[0087] 电话途径的主要缺点来源于用于设定连接的方法,对于数据应用来说,时间长得可笑。在SAIN网络中,包括设定时间的连接时间可为一个毫秒或更少,在多数情况下不浪费带宽,连接可使用单行消息在几个微秒或更少的时间内从虚拟状态或睡眠状态转换到真实状态,该单行消息通常为封装在控制矢量中的一个字节或更小的长度。虚拟状态不要求带宽,而且睡眠状态使用少数量的保活(keep-alive)带宽。虚拟和睡眠状态将在下面进一步描述。控制矢量可为由小单元构成的隐式寻址消息,其中每个是应用于通信流程某些方面

的消息。

[0088] 两个参数确定在SAIN网络中要求的小单元和帧大小,它们是:

[0089] 1. 中继(trunk)或链路数据率(每秒比特);以及

[0090] 2. 传输用户连接要求的带宽最小数量。

[0091] 为本申请之目的,中继线路是诸如光纤连接的物理对象、承载数据流量跨越网络的有线或无线连接。链路和中继通常承载若干个诸如隐式或显式定义的时/空分对象的连接。

[0092] 例如,1G赫兹(GHz)链路数据率的链路可支持期数据率小于每秒1G比特带有1比特小单元的连接。帧尺寸中小单元的数目取决于要求带宽的最小数量。帧周期是每帧小单元的数目与链路数据率的乘积。要求带宽的最小数量,叫做量子数据率(Quantum Data Rate)(QDR),等于小单元大小除以帧的周期。帧周期是每帧小单元数目除以链路数据率。例如,若帧周期为1微秒且小单元大小为1比特,QDR为每秒1,000,000比特。对于链路数据率超过1Gbps的,小单元可更大一些。例如,一个8比特小单元可封装其集合数据率为8Gbps或更小的1比特小单元的集合。对于1微秒帧,QDR可为每秒8,000,000比特。

[0093] 本申请的实施例将由用户连接数据集合或这些集合的集合组成的流量分割成小单元帧。它们通过如下来使用设定连接和其带宽的方法:1)对于每个连接和其集合规定每帧的小单元数目,以及2)提供将每个数据小单元置入帧内已分配物理时间和/或空间位置的时钟机制。术语“连接”(connection)是用于连接集合与用户级连接的通用术语。

[0094] 本申请中公开的方法使用同步基础网络技术,就是“SAIN”方法。公开于此的SAIN方法对于为任何目的实现数字通信网络是有用的。一个目标是建立能与现存网络互联和使用现存网络组件的网络。第二个目标是满足现有和未来挑战的新一代网络而奠定基础。实施例注重那些能在总体上使未来网络受益并有效增强现有网络的方法。

[0095] 为使该方法和装置易于说明和理解,附图和讨论中使用的这些示例一般是对于城域网和尤其对于城域以太网而言的。这样做并不限制该技术在其他情境中任何方式的使用。本申请通常以硬件术语来描述装置。正如本领域技术人员所知,以硬件描述的组件也可在软件中实现,并且软件版本可推出相同的结论。

[0096] 下面是该技术的某些基本方面,以集合或个体的形式使用文字来定义其方法:

[0097] 1. 两点连接控制可减少逐跳连接路由的需求。建立连接和动态控制其带宽仅发生在SAIN底层网络100亮线以内的源和目标点。这种网络控制从用户数据协议中分离;

[0098] 2. 该网络控制平面从用户端物理上和逻辑上都不可及,因此增强了网络安全性;

[0099] 3. 连接可基于单工基准设定,双工连接包括两个单工连接;

[0100] 4. SAIN网络可存在于两层或多层分层网络拓扑中,因此能使得网络控制大规模分布;

[0101] 5. 同步网络节点到共同时钟可减少大部分的复杂性和异步时钟备选方案的随机性;

[0102] 6. 交换原理可使用描述于引用发明人专利中的物理基于电路的复用机制。这克服过去电路交换中的限制并对于低和高带宽都拥有更高的伸缩性和低确定性时延(deterministic latency);

[0103] 7. 使用具有路由备选方案的大量选择的半静态路由替代动态逐跳路由进一步简

化了网络。该途径导致确定性操作参数,包括动态连接带宽;

[0104] 8. 因为同步节点,时延可成为SAIN网络中确定服务品质(QoS)所要求的唯一衡量标准。在SAIN网络内部,包缓存拥塞不需存在以至于丢包率作为QoS参数也就没有意义了;

[0105] 9. 该基础结构和部署参数使得抖动和延迟变化足够小到可忽略;

[0106] 10. 除了灾难的情形,网络的控制机制保证了接受进网络的所有数据的发送;

[0107] 11. 在随机网络中,要求流量整形和用户管理。总的来说,二十世纪七十年代的随机网络是那个世代基于消息、文件、转发的流量市场的合适选择。现今的语音、视频以及多媒体市场显著地是流量流,即基于电路。将随机网络变形为协议覆盖的基于电路的网络在现今的网络中是个不小的工作。将基于电路的底层置于已存在的之下则更为简单。使用更少的电功率使得资本和运营成本更加低廉。

[0108] 现有互联网要求相对少数目的较大‘一个尺寸适合所有’(One size fits all)的边缘路由器。SAIN网络结构变形为大量大规模分布式迷你边缘路由器(mini-edge-routers)。每个迷你路由器注重于本地用户语言、社会和商业需求,以及城域网内与外界的包容性互联互通。本地用户协议中的所有用户数据进存在于SAIN底层网络100的入口和出口边缘。内部,网络使用OSI第2层和其上的协议以及OSI模型的物理传输层1而存在于主机、终端、服务器或网络101入口和出口连接之间。换言之,其存在于其自身SAIN底层1.5。该定义并不排除使用仿真SAIN底层存在于其上的物理层1的协议。

[0109] 主机、终端、服务器或网络101,使用任何形式的数字访问协议,通过用户接口连接290连接到SAIN网络很象传统网络。SAIN网络的主要目的是向用户提供支持其现有需求而不要求对现有用户应用进行修改的网络。另外的目的是使得服务提供者使用现存的网络部署克服现有网络在伸缩性、性能和成本上的缺陷。

[0110] SAIN底层网络100的顶层原理强调相比于现存网络的一个主要好处。SAIN网络将用户数据转换为符合底层1.5到处使用的简单转发协议(simple forwarding protocol)的比特流。

[0111] 该简单转发协议的主要目的是从源到目标端点鲁棒地和确定性地透明转发用户数据比特。该方法在网络的交换节点之间,以减少由现有网络的随机性造成的大部分复杂性和服务品质问题的方式,使用同步时钟。该时钟机制可着重于同步各个节点时钟。这可包括同步所有节点到基于现存网络同步技术的通用协调时间(Coordinated Universal Time)(UTC)。

[0112] 具有同步时钟的SAIN节点使用确定性方法以在网络入口端接收数据用于发送之前克服“突发”数据。置于数据进入SAIN底层网络100之前的包缓存保证了不丢包地发送。包缓存不依赖于SAIN底层1.5的内部。在传统网络中于路由器内设置缓存是服务品质复杂性和性能差的主要原因。SAIN方法减少了出现在传统核心网和接入网的数据突发性。

[0113] 参数化用户接口210是柔性数据接口,其可1)对于商品数据类型是通用的,且2)对于特殊数据类型是专用的(application-specific)。PUI80是可替代的和可升级的,以满足改变用户或网络提供者的需求。

[0114] SAIN网络可使用现有网络的要素。例如,参数化用户接口210从用户输入数据中以提取详细信息以确定网络中预期的出口目标。还可确定流量归属的服务类。与传统路由网络不同,网络通过对于每个流量类型应用更多或更少带宽区分流量的优先次序。可调整带

宽以满足对于突发流量和随时间变化的数据率的严格延迟指标。没有过供应, SAIN途径可防止网络拥塞以及丢弃带宽已不足却强制数据重传输的包。

[0115] 集合的第一级减少了大量路由器复杂性以及基于包的方法所要求的处理能力。重要的接口将着重于诸如由IEEE、ITU、ANSI和城域以太网论坛(Metro Ethernet Forum)(MEF)等组织规定的城域网(MAN)以太网标准。该关注没有建议限制这里实施例所应用的各类网络的范围。

[0116] 到SAIN网络的接口包括词条“参数化的”是有重要原因的。既然SAIN网络使用广泛应用于所有网络接入协议的内部转发数据协议, 可有参数化用户接口(PUI)210的多种变化以适应外界。每个连接入口PUI211/出口PUI212支持用于具体用户应用的相互兼容协议。除此之外, 没有技术限制。接口的参数化特性允许新用户通过软件从用户、其组织(例如虚拟专用网络)(Virtual Private Networks)或如升级一样从网络服务提供商下载来访问加入到入口PUIs211和出口PUIs212的协议。

[0117] 参数化用户接口210方法的另一个优势在于处理多种流量类型的分布式本质。大量高度分布式的小处理器替代了复杂的“所有事情对所有人”(all things to all people)的大边缘节点路由器。在相对简单元件的大型网络中分散处理能力是一种有效方法以相对低成本地产生庞大的处理能力。

[0118] 入口PUI211使用协议翻译器214功能以在源端点处将用户数据协议封装进SAIN网络转发协议。在目标端点, 协议翻译器214将SAIN内部网络协议改回用户友好的形式。E节点包含若干个定义于上的通过入口和出口PUIs212的用户连接。

[0119] 本申请的公开在于示于图2的分层结构的上下文中。该图将SAIN网络分割成层, 每层执行具体定义好的功能。网络节点可存在于平体系结构中(flat architectural structure), 但其可能丧失路由简单性和其他的好处。如图2所示的SAIN网络层三个顶层的下两个为建立开始于城域网的大型网络提供坚实的基础。就是, 网络在一组连接到网络的用户、独立外在源和现存核心网之间交换数据。SAIN网络可通过使用顶层交换节点, 即, 如图2所示的X节点层40与两个下层的结合, 扩展到更大的集成结构。

[0120] 变换算法的基本原理

[0121] 图4、5和6概述了SAIN变换技术的基础方面。参考专利包括本申请复用方面的更多详细内容。

[0122] 为本申请之目的, 一帧数据是周期性的、排序的、小单元的时/空集合, 其中每个小单元包括规定数目的数据比特。在给定的帧内, 小单元具有同样数目的比特。每个小单元绑定到具体的连接(或它的集合作为一个新连接)。换言之, 每个小单元是来自于短或长序列数据流的片段。为了在单个帧内传递若干个数据流, 来自于该若干个连接的小单元在帧内混合。SAIN变换算法将来自于给定连接的小单元通过时/空分割均匀等距排布。每个小单元表示带宽的一个量, 等于小单元中的比特数目除以帧周期。换言之, 小单元表示量子数据率(QDR), 等于小单元中的比特数目乘以周期帧率。

[0123] 为简短之目的, “时/空”的使用变成了“时间”。换言之, 象“时/空分复用”的短语变成了“时分复用”。如果本申请中没有具体指出相反, 词语“时间”在某些方面涉及隐含“时间”和“空间”作为其基础的复用。

[0124] SAIN变换算法包括在两个域中定义小单元的一个帧, 这两个域分别是如图所示的

连接域150和时间域160。如图4(a)所示的连接域定义单个连接为邻近范围的小单元一例如,图中的四个排列在一起的“A's”。连接带宽为4倍的QDR。

[0125] 第二域是如图4(b)所示的时间域160,其具有与连接域150帧同样数目的小单元。每个小单元在第二域中的定位如下确定:

[0126] 1. 分配二进制数字,起始于0,到连接域150的每个后续小单元位置;以及

[0127] 2. 向后读取二进制数以确定每个小单元在时间域160中的相应位置。

[0128] 图4(b)示出图4(a)通过时间域160帧传播的4个小单元。连接域150中的每个小单元与时间域160的匹配小单元一对一对应。[注意:域存在仅定义1)连接的数据率(连接域)以及2)时/空帧(时间域)内连接的小单元位置。封装在小单元中的数据仅存在与物理帧中,其中SAIN变换的时间域指向帧内的小单元的连接位置。]

[0129] 例如,从连接域150开始,左边第一小单元是0(即二进制符号0000)。明显地,时间域160中的匹配小单元也是0000(即十进制符号0)。连接域150向右的下一个小单元是0001,在时间域160中的匹配小单元是1000(即十进制符号8)。连接域150接下来的小单元0010和0011在时间域160中的匹配小单元是0100和2100,(即4和12)。逆向也为真;物理时间域160中的每个小单元指向连接域150中的一个小单元。物理小单元仅存在与时间域160中。连接域150中的小单元指向时间域160中小单元的物理位置。

[0130] 合起来,两个域定义了用于复用系统的时间域内存印象图。时间域160帧中的每个位置表示对应于表示每个小单元归属连接域150中的小单元的物理时间(或空间)小单元。

[0131] 使用该算法建立帧的好处在于小单元位置通过时分帧可几乎均匀分布,因此对于任何给定连接减少交换时延。

[0132] 2的次方数帧和段(segment)长度

[0133] 时间域160可通过将帧长度除以整数可分割成段。这里F是帧长度(即小单元数目)且n为整数,若n是F的整数约数则每段正好包含 $\text{INT}[F/n]$ 个小单元,或者若n不是整数约数,将是 $\text{INT}[F/n]$ 和 $\text{INT}[F/n]+1$ 的组合。在SAIN交换内通过时间域160帧将时分帧分割成段开发分布式小单元定位。

[0134] 获得等距分布小单元的时分帧

[0135] 时分交换工作基于以下的基本原理:在重排序(或其他操作)以传输到外向链路上之前,存储来自于一个或多个源的数据帧或数据的子帧段。分割帧是减少交换时延的方法。有多种方法将帧分割成段,但用2的次方数分割是重要的一种。图4显示2的次方数(PoT)长度帧的基本案例。该图显示了PoT长度的帧和PoT长度的连接。另外,对于图4(a)和图4(c),连接在帧内起始于“PoT边界”(PoT boundary)。换言之,连接域150的第一小单元起始于边界为“0”或帧中小单元数目的PoT整数约数。在图4(b)和图(d)中,其差别是连接域150和时间域160帧内的小单元的位置。在两种情况中,时间域160中的测来床上是等距的。

[0136] 该等距特性并不限于PoT约数。任何整数只要其为帧长度的因数则可用。例如,具有20个小单元的帧可分割为每个具有4个小单元的5个段。置于每段中的单个小单元定义了带宽等于五倍帧量子数据率的连接。

[0137] 非等距分布2的次方数长度连接

[0138] 图4(e)和4(f)显示连接域150最左边小单元的结果,其在连接域150中不起始于

“0”或帧长度的PoT因数边界。在图4(f)中,小单元不再等距。但是,小单元仍然存在于帧的PoT段内,每段是帧长度除以4。如图4e所示,连接域150中四小单元邻近范围的任何位置导致每个PoT段有一个并且仅有一个小单元归属于连接。此外,4小单元范围的每个最左边位置导致小单元位置的不同模式。这种事实,对于一般应用而言,在交换内和交换之间使用SAIN变换方法具有重要的结果。在标题为“用于连接串关节点(Tandem Nodes)的实施例”公开了这些结果。

[0139] 将非2的次方数长度帧除以一个2的次方数

[0140] 将一帧除以一个2的次方数产生一个2的次方数分段帧(segmented frame),即PoT段帧。PoT分割不依赖于2的次方数的总帧长度;其可用于任意长度的帧。图5(a)显示在一个13小单元帧中起始于小单元位置“8”的四小单元连接。[注意数字13是一个质数,即其整数约数为1和13。因此,并不是所有的段有相同数目的小单元。]在图5(b)的时间域160中,第一PoT段有4个小单元时隙且在其他每段中有3个。

[0141] 在图5(c)连接域150中,4个小单元连接“A”开始于位置“6”,其不能被帧中PoT段的数目4整除。另外,图5(c)显示图5(c)的连接域150以及图5(d)的时间域160的完整16个小单元虚拟帧。阴影中的小单元识别未被真实帧包括其中的虚拟小单元。换言之,它们是0长度小单元位置。图5(d)时间域160的结果是4个小单元,每个小单元仍将包含在除以4的PoT段中。图5(e)是真实帧结果,其中虚拟小单元设置成0长度。因其在图5(a)和图5(c)中的起始时间不同,图5(e)中的小单元几乎不象5(b)的小单元那样等距。请注意在图5(a)和图5(d)中,位置“8”和“9”的小单元包含四个“A’s”中的两个,并且它们出现在图5(b)和图5(e)的位置“1”和位置“8”。本质上,图5(a)中的位置“9”和“10”的小单元已移到小单元位置“6”和“7”。即便如此,事实还保持这样,在连接域中保持在四邻近位置的4小单元导致每个小单元具有自己的PoT段。这是使用在SAIN交换中算法的另一个重要方面,其赞成使用SAIN帧的PoT分割。

[0142] 图5(d)使用包括于此的文字说明来达到说明之目的。除了其真实帧长度13小单元之外,其还具有虚拟帧长度。对于真实帧长度不是2的次方数长度的所有情况,虚拟帧长度是大于真实帧长度的最小的2的次方数。对于非2的次方数帧长度通常之情况,获得时间域160依赖将连接域150帧长度分割成2个相等的段。第一段的长度等于真实时间域160帧。PoT虚拟帧长度的剩余部分形成第二段。系统将对应于连接域150第二段的时间域160的每个小单元视为不存在。图5(e)(加上图5(c)虚拟连接域150中右边三小单元位置)例示了算法的这种特性。图5(c)和5(d)的连接域和时间域中对应的索引低于和高于如图5(e)所示的时间域160版本。

[0143] 处理非2的次方数连接

[0144] 图6显示连接长度既不是2的次方数也不是2的次方数整数倍的情况。图6显示各种帧配置的五小单元连接。图6(a)和6(b)显示连接域150中具有起始于小单元位置“0”的五小单元连接的16小单元帧。在时间域160的图6(b)中,三个PoT段有单个小单元而一个PoT段有两个小单元。如图6(c)所示的五小单元连接起始于图6(a)中的小单元位置“1”而不是“0”。第三PoT段现在包含如图6(d)所示的两个小单元而不是如图6(b)所示的第一PoT段。

[0145] 图6(e)和6(f)显示出改变帧长度并不影响结果,就确定包含两个小单元位置的PoT段而言。图4、5和6的示例对于理解本申请的实施例是有用的。

[0146] 网络中路由E节点到E节点路径的概述

[0147] SAIN网络的一个实施例可使用一个网络层的若干个E节点,其连接到下一个更高网络层中的若干个T节点。每个E节点对于网络中其他每个E节点而言可作为源和目标节点。每个源E节点到目标E节点的连接是路径。E节点内的路径集合嵌入从E节点到每个其他E节点的每个路径。该集合连接到父T节点以用于处理和转发到网络中每个T节点。每个目标E节点分解该路径集合,其源来自于两层网络的所有其他。

[0148] 为说明和实施之目的,本申请假设每个E节点连接到下一个更高层的单个T节点。扩展到多个连接可从两个途径实施,一个是使E节点隶属于一个或更多个父辈。另一个是分割父T节点为若干个子T节点,此为生存性和安全原因而付出。

[0149] 在图7中,每个E节点200连接到父T节点300。数据可从E节点200向父T节点300以及从父T节点300向子E节点200流动。每个E节点200通过T节点300的下一个更高层网络连接到网络中的其他所有E节点200。每个T节点300可通过直接物理连接或通过中转节点连接,连接到网络中其他所有T节点300。中转节点可为其他的T节点300。在每个成对节点连接中,数据可沿两个方向的每一个流动。每个源E节点与若干个目标E节点配对。每个源T节点(源E节点隶属的)与若干个目标T节点配对。在点对点连接中,一个源E节点连接连接到单个目标E节点。在多点连接中,一个源E节点可连接到若干个其他E节点。在T节点之间,点对点和多点连接操作相似。

[0150] 网络实施例可包括不同于E节点200和T节点300的节点,而且可包括直接连接的E节点200。

[0151] T节点300互联的一个实施例是如图8所示的网状网络,其作为描述SAIN网络的功能和实施方式的示范网络。双工TT链路340使T节点彼此连接。网状网络是在网络设计中一般技术人员公知的多个可能配置中的一个。不管配置或其大小如何,本申请的方法和装置可适用并可调整到很大尺寸的网络。网络往往是集线器簇(clusters of hubs),如2002年由Perseus Publishing出版、作者为Albert-Laszlo Barabasi的《Linked, the New Science of Networks》中详述的那样。经济可决定何时建造多个城域网并互联它们是划算的。

[0152] 图8用于说明本申请公开的示范网络的基本原理。图中所示有20个T节点,假设每个T节点是25个子E节点的父辈,以至于网络包含500个E节点,每个E节点可彼此通信。

[0153] 在SAIN网络中,一个路径是从一个E节点200到另一个E节点200的单工连接。双工用户连接包括两个路径,每个传播方向上一个。

[0154] SAIN网络中T节点300的任务是提供超级路径,其为E节点200到E节点200路径的集合。它们的互联也是在单工基础上设定的。这些超级路径可使用双工控制矢量控制,该双工控制矢量包含嵌入于隐式寻址帧内的小单元形式的消息。可能有其他用于超级路径的控制方法。

[0155] 路径的任务是在源E节点200处集合用户连接并传输该集合而不是单个的连接到目标E节点200。每个E节点200的互联对(interconnected pair)可作为描述于下的源节点和目标节点。

[0156] 源E节点是每个路径的控制节点。换言之,源E节点200使用路径的预定路由,其具有足够的带宽一支持达到从源到目标E节点200的端对端(end-to-end)用户流量。当用户流

量强度变化了,E节点200分配更多或更少带宽给路径,并且伴随地,可包括可用带宽的调整以支持网络带宽不足时的多种流量。下面的实施例详述涉及完成这些任务的装置和方法。

[0157] 网络的主要需求在于能将接受用户数据的所有节点互连到所有能传输用户数据的节点。使用如图2所示的三层体系是一种减少网络复杂性而相伴增加伸缩性和性能的重要途径。概念上而言,E节点层20主要是数据集合层且T节点层30是路由层。[X节点层40是T节点的X节点城域网域和E节点之间的转发层(forwarding tier)]

[0158] 在城域网中,E节点200集合所有引入的用户数据到若干个路径内。每个路径是进入E节点200以传输到另一个E节点200的所有数据的集合。E节点200集合其路径成超级路径,其数目等于网络中T节点300的数目。于是集合这些超级路径成更高级别的超级路径,其包含可传输到所有其他E节点200的所有用户数据。源E节点200转发此超级路径到其父T节点300。该父T节点300于是将每个中间超级路径路由到合适的目标T节点300。在示范网络中,结果是25个源E节点200连接到25个隶属于20个T节点300的每一个的目标E节点200,有一个例外。[一个源E节点200不需要连接到该源E节点200自身以获得数据。其可在测试环回安排中设定少数量的带宽以验证其双方向到其父T节点300连接的完整性。]

[0159] 在多链接T节点300的配置中,若干个可能路由存在,每个可描述路径的超级路径集合。这些路由的表格可包含重要参数,这些参数使得网络动态选择优化网络性能的路由。例如,每个预定路由是无环路的,具有熟知的端到端时延。该表格还可包括对于每个路由的可用带宽,由系统周期性地更新。

[0160] 引用标号的方法

[0161] 接下来,一般转发元件(即,那些发送或接收数据的)按下述方式分配图示参考标号:1)结尾是“0”,或2)单个数字或两位数字。每个元件的子类型保持其首位或首两位数字并当主题涉及源端功能时加“1”。子类型加“2”以用于目标端功能。例如,E节点200表示发送和接收数据的一般E节点。源E节点201表示E节点200的发送端功能,以及目标E节点202表示其接收端功能。这样做的原因是区别开网络的发送和接收功能并因此简化下面的公开。

[0162] 下面的公开首先描述网络中单独子系统的实施例,接着再作为整体来公开系统的实施例。

[0163] SAIN交换栈选择器的实施例

[0164] 图9a显示第[0017]段引用的现有技术专利中的图3a交换栈选择器120的解释。另外,该图包含现有技术公开中未包括的帧时钟发生器121。其使公开于此的SAIN交换网络中快速动态交换方法和装置成为可能。图9a包含一些来自于现有技术图的更新引用标号,其与本申请的其他图兼容。

[0165] SAIN交换的一个实施例包括如图9a所示的交换栈选择器120。交换栈选择器120操作的概述如下:

[0166] 交换栈选择器120实现了描述于上面自第[0063]段开始的“SAIN变换算法的基本原理”的SAIN变换。引用专利详述了适用于如图9a所示装置的方法。该装置包含小单元计数器133,其取得SAIN帧中每个位置的新状态。该计数器133开始于帧的全0状态。其具有从0到最小整数2的次方数N-1的范围,其中N是对于一给定长度帧F的虚拟帧长度(即,在段及其后,将非2的次方数长度帧除以2的次方数),换言之,

[0167] $V = 2^n$, 其中 (1)

[0168] $n = 1 + \text{INT}(\log_2(F-1))$ 。

[0169] 帧时钟发生器121在帧期间发出帧时钟130脉冲,包括设定系统时钟124和帧时钟130为0的帧复位123脉冲。每个帧时钟130脉冲使得计数器增量“1”。小单元计数器133对于环境计数包括虚拟帧状态。如描述于段及其后的,环境可包括帧中空的小单元。

[0170] 如下面所示,图9a中的小单元计数器133是若干个标柱为CC/CBR空连接134 CC/CBR、备用连接(Spare Connection)135、以及CC/CBR操作连接136的连接比较器/连接带宽寄存器(CS/CBRs)。图9b显示CC/CBR操作连接136分割成其两个部分。它们是连接比较器141和连接带宽寄存器142。连接带宽寄存器142部分规定了连接域150的虚拟帧中分配给连接的小单元范围内的小单元的起始位置。

[0171] 分配给连接m的小单元数目等于存储于连接带宽寄存器连接m+1处的数目减去存储于连接带宽寄存器连接m处的数目。若两个数相等,则m连接在连接域150中没有小单元。换言之,其表示虚拟连接。[一个虚拟连接是具有0分配带宽的连接,虚拟连接是可成为数据支撑的物理连接占位符。]这是SAIN网络内的独特和重要特性。一个连接可存在于虚拟状态,即使当调用、路径或超级路径均没有分配带宽。这是在SAIN网络内使用隐式寻址的重要益处。

[0172] 请注意小单元计数器133的最小重要比特(least significant bit)出现在图9a的左边。一个CC/CBR空连接134、CC/CBR备用连接135和CC/CBR操作连接136具有在图中最右边位置的其最小重要比特。这种安排产生了SAIN复用算法。

[0173] 当小单元计数器133中的数是:

[0174] 1. 大于或等于连接m处连接带宽寄存器142中的数,以及

[0175] 2. 小于连接m+1处连接带宽寄存器142中的数,

[0176] 系统将图9a中的附属选择器线路138置为“on”状态。所有其他选择器线路在“off”状态。每个选择器线路138确定SAIN交换内发送或接收的当前小单元的连接域150位置。若在连接m处连接带宽寄存器142的内容和在m+1处连接带宽寄存器142的内容相等,则任何选择器线路138都不为“on”状态。

[0177] CC/CBR空连接134确定是否小单元计数器133的“1”最小重要比特预报一个虚拟小单元作为其下一个状态。真实小单元存在于编号为(0, 1, 2... F-1)的帧位置。连接域位置也有相同的编号。区别在于连接域中虚拟帧小单元不存在于位置(F, F + 1... N-1)。仅当连接域150的值大于虚拟帧长度N/2-1时,所有小单元计数器133的取值对应于不存在的虚拟连接。若不其然,虚拟帧长度的虚拟PoT值将更低。如图5(c)所示,不存在的(即0长度)连接域小单元位置出现在帧尾部的邻近位置。在时间域,图5(d)显示其出现在遍及虚拟帧中具有0长度连接域位置的1对1位置的位置。

[0178] 另外,基于2的次方数长度段(即,PoT段)交换的一个实施例可使用SAIN变换算法的特性去指定所有PoT段边界。小于2的次方数的任意长度F的帧,可包含等于小于F的最大2的次方数的PoT段的最大数目。换言之,PoT段的最大可能数目是N/2,帧的虚拟帧长度除以2。当应用该变换所发到帧的连接域时,PoT段边界开始于时间域中对应于连接域中最初N/2个小单元的N/2个小单元。若帧长度自身是2的次方数长,真实帧长度和虚拟帧长度是相同的,即N = F。既然这样,N取代上面的N/2。

[0179] 每PoT段一个小单元的组规定了最大基本数据率,其是帧支持的最大PoT数据率。

最大基本数据率的任何因数可有助于SAIN网络。基本数据率(包括最大基本数据率本身)的任何整数倍也是可能的。例如,若帧长度为6小单元,虚拟帧长度N是8。N/2是4,在帧中有两个0长度小单元位置。在时间域中,一个0长度位置发生于两个PoT段中的每个。最大基本数据率是每帧周期两个小单元,以及连接的最大数据率是每帧周期6个小单元,即,每PoT段3个小单元。

[0180] CC/CBR空连接134包含对应于连接域150帧中第一不存在小单元的连接域150数。CC/CBR空连接134能确定现有小单元计数器133值增加“1”是否会导致连接域150不存在小单元的位置。这通过反转连接到CC/CBR空连接134的最大重要比特(most significant bit)小单元计数器133中的最小重要比特来产生。

[0181] 这等同于当其当前值结尾为“0”时增加小单元计数器133“1”,该值表示连接域150虚拟帧的前半部分的连接域地址。存储于帧中的该值当该帧小于虚拟帧长度时为F。若F=N,即,若实际帧长度为2的次方数长度,CC/CBR空连接134为空。

[0182] 帧时钟发生器的实施例

[0183] SAIN网络在每个网络节点安置了高速系统时钟。该若干个节点时钟可直接或间接与共同时钟源同步。E节点中的时钟可同步到它们的父T节点300,并且每个T节点300可使用诸如IEEE标准1588或包括美国专利2,986,723的其他方法的标准时钟技术直接或间接连接到共同时钟源。

[0184] 图10a显示帧时钟发生器121的框图。其目的是产生若干个帧时钟130脉冲,等于对于具有帧大小为F的小单元位置的交换栈选择器120而言每帧小单元的数目。每个脉冲对应于交换栈选择器120的时间域160帧内的一个位置提前(an advance of one position)。

[0185] 时钟发生器121使用三种输入信号。一种是系统产生的帧复位123信号以表示帧的开始。另一种是高速形同时钟124信号。第三种是帧大小增量(Frame Size Increment) (FSI) 122,其使得从来自于高速系统时钟124的帧时钟130获得,以至于:

[0186]
$$FSI = f_s \cdot p_F / F \quad (2)$$

[0187] 其中, f_s =以兆赫兹为单位的高速系统时钟率,以及 p_F =以微秒为单位的帧周期,以产生帧时钟130和正交时钟(Quadrature Clock) 131脉冲。

[0188] 具有微处理器的网络控制器存储如图10a所示的实体;图10b显示发生于每个帧循环(frame cycle)的操作顺序。初始地,当系统启动或重新启动时,帧中有新数目的小单元,系统从Start开始并执行图10b中的操作(1) 601。控制器使用公式(2)计算当前帧大小的FSI、高速系统时钟率以及想要的帧周期。下一个步骤是将FSI除以2并存储结果与增量寄存器(Increment Register) 125。这样做的原因在于CC/CBR空连接134机制要求用于正确操作的两个穿插时钟信号(interspersed clocking signals)(如段及其后所描述的)。提供这些要求信号的简单方法是产生帧时钟130和正交时钟131。

[0189] 当帧复位123在网络中发生时,操作(2) 602启动。该信号保持了一组时钟,并因此在SAIN网络节点内同步帧起始时间。该操作设置系统时钟计数器129和比较器127内的触发器(Flipflop)为“0”。该触发器辨别加法器寄存器(Adder Register)中的值是表示帧时钟130还是正交时钟131脉冲。在所描述的操作中,触发器值“0”表示正交时钟值,其导致接下来说明的后来比较值的循环。

[0190] 操作(3) 603开始从系统时钟124给系统时钟计数器129增加1。操作(4) 604探测系

统时钟计数器129的溢出状态。探测溢出的目的是保证系统已保持与帧复位123的同步。

[0191] 操作(5)605确定系统时钟计数器129是否大于加法器寄存器126中的值,如果不大于,其回到操作(3)603。如果为真,系统将转到操作(6)606,其中会发生两件事。[注意:既然值0.5首次存在于加法器寄存器126中,计数的第一系统时钟129脉冲大于存储在帧复位时间的0.5]第一个将隶属于比较器127的触发器从“0”改变为“1”,或从“1”改变为“0”。在帧的开始处,触发器设置为“0”状态。这导致其状态从“0”变为“1”。操作606还导致加法器寄存器126增加FSI/2存储于2增量寄存器125中。下一个时间系统时钟计数器129大于加法器寄存器126,触发器状态设置为“0”。

[0192] 操作(7)607确定触发器的状态并发送帧时钟130(操作(8)608)或正交时钟131(操作(9)609)上的脉冲。操作(10)610增量寄存器触发器(Increment Register Trigger)125a给加法器寄存器126增加FSI/2并触发操作(3)603。

[0193] 为实施例的数字示例,设 $F=3,856$ 小单元, $p_f = 0.125$ 微秒(msec),并且 $f_s = 10^6$ kHz (1 GHz)。FSI的小数部分需要具有足够的二进制位置以保证帧计数准确等于F个小单元。该数可通过下式计算:

$$[0194] \quad FSI = \text{INT} (\text{TotN} / F \times 2^{\text{Exp}}) / 2^{\text{Exp}} \quad \text{其中} \quad (3)$$

[0195] $\text{TotN} = p_f \times f_s$ = 帧中高速时钟脉冲的总数,以及

[0196] $\text{Exp} = \text{INT} (\text{LOG}_2 (F - 1) + 2)$ 。

[0197] 使用这些公式,

$$[0198] \quad FSI = \text{INT} (10^6 \times 0.125 / 3,856 \times 2^{13}) / 2^{13} = 32.4169921875。$$

[0199] 请注意当如图10b所示的帧大小增量122要求给Exp值增加1时,使用FSI/2。

[0200] 一个备选实施例将两状态触发器替代为单稳态触发器(未示),其中来自于触发器输出脉冲的尾部产生正交时钟131。既然这样,当增量寄存器125值和Exp值未增加1时,操作601未执行(即,FSI未除以2)。

[0201] 连接比较器/连接带宽寄存器(CC/CBR)级(stage)的实施例

[0202] 图11显示将一般分解交换(Generic Disaggregation Switch)502内的连接比较器/连接带宽寄存器(CC/CBR)级与器配对的一般集合交换501同步。(“一般集合交换”和“一般分解交换”意味着该术语使用的上下文可适用于任何交换级别)。这里的“连接带宽寄存器栈”的缩略语是CBR栈。对于“一般集合交换/一般分解交换对”变为一般A/D交换对。所有SAIN交换操作一般A/D交换对503如从数据修饰(data grooming)中分离的交换过程。这样的修饰主要发生在E节点中,其中一般集合交换501和一般分解交换502正常存在。当流量带宽改变(更高或更低)进入一般集合交换501时,它激活方法以改变其自身CBR栈553并同步改变其配对的一般分解交换502的CBR栈553。该方法使用两个标柱为CBR栈A553a和CBR栈B553b的CBR栈。在操作中,在每个交换对中,每个交换中的一个CBR栈553管理当球流量的带宽,而另一个栈准备变成下一个活动版本。源一般集合交换501转发控制矢量消息到其配对一般分解交换502,指定将开始应用新带宽分配的帧数。

[0203] 在交换节点控制器560的交换操作之后,两个CBR栈交换角色。若CBR栈A553a在操作,交换节点控制器560激活标柱“A”的元件,如图中深线所示。

[0204] 若交换为一般集合交换501,来自于连接源或接收器门(Sink Gates)550源端的源的流量小单元传递数据小单元以填充外向复用流中的小单元位置。若交换为一般分解交换

502,内向复用流中的小单元位置的数据小单元则传向目标接收器(sink)。

[0205] 比较器/连接带宽寄存器级的一个实施例是图12的着重点。其显示图11的元件,这些元件具有能在帧内快速改变分配给连接的带宽的配置。在要求带宽分配或重分配的情形,SAIN交换可使用这种方法。

[0206] 如图12所示,连接带宽寄存器具有5个部分标签。其中的两个是连接带宽寄存器142a和连接带宽寄存器142b;第三个是流量管理处理器(Traffic Management Processor)563,第四个是C栈加法器寄存器556,第五个是比较器寄存器555。连接带宽寄存器142a和连接带宽寄存器142b包含该数目的小单元,因此就是,分配给交换栈选择器120的带宽。取决于CBR栈A553a还是CBR栈553b在图11的交换内在运行。交换节点控制器560确定是CBR栈553a还是CBR553b激活。交换节点控制器560激活转接线(Switchover Line)A561a,从而激活图11和图12中标柱“A”的元件。通过激活转接线B561b,交换节点控制器560激活所有“B”的项目。

[0207] 如图9a所示,系统帧复位123信号在交换栈选择器120中发起帧。CBR栈A553a和CBR栈553b。集合/分解(AD)交换对的每个包含为内存印象图的连接带宽寄存器栈。一般集合交换501作为其自身内存印象图和其配对一般分解交换502中内存印象图的主导(master)。假设一般分解交换502中的连接带宽寄存器栈A502a的内存印象图与一般集合交换501中的连接带宽寄存器栈A502a的内存印象图恰好匹配。下面列出的动作列表或与其类似的可保持一般A/D交换对503的两个连接带宽寄存器栈同步。

[0208] 1. 在两个交换中,在同步过程开始时,保证每个交换中的非运行连接带宽寄存器栈与交换中的运行连接带宽寄存器栈匹配;

[0209] 2. 1)数据到达路径集合交换511或2)任何类型集合交换的路径或超级路径带宽发生了改变;

[0210] 3. 假设当球集合链路带宽对于支持该改变是充足的。[若不是,公开于此的其他方法可克服该缺陷]

[0211] 4. 一般集合交换501设定非运行连接带宽寄存器栈中的新带宽并导致控制矢量发送到一般分解交换502;

[0212] 5. 当两个交换开始使用新更新的连接带宽寄存器栈时,使用枚举帧(enumerated frames),从一般集合交换501向一般分解交换502发送控制矢量消息指示帧数;

[0213] 6. 周期性地确保一般A/D交换对503中的连接带宽寄存器栈保持同步。

[0214] 改变路径和超级路径带宽的实施例

[0215] 改变网络中分配给帧的带宽与改变帧内单独连接的带宽同时发生。改变SAIN交换栈选择器120中的帧带宽是改变帧内小单元数目的两状态过程。第一个状态涉及将帧内若干个连接压缩成起始于地址“0”的连接域150小单元的邻近范围;第二个状态涉及向连接域150/时间域160帧中增加或从中取走小单元。步骤发生的顺序取决于每帧小单元的数目是增加还是减少。

[0216] * 当增加帧大小时,增加其大小先于增加帧内连接的带宽。

[0217] * 当减少帧大小时,减少帧内连接带宽到适合更小尺寸帧的程度先于减少帧大小。

[0218] 图11显示实现带有单个比较器栈551和双连接带宽寄存器(CBR)栈553a的交换栈

选择器120。在给定的交换栈选择器120内，一个连接带宽寄存器，例如，CBR栈A553a在运行。另一个CBR栈B553b正准备在连接域150/时间域160连接分配中改变。CBR栈553和伴随比较器栈551支持同样数量可为任意大的连接。

[0219] 若集合交换节点控制器560发现给定帧内要求的带宽数量必须增加时，第一步将当前连接压缩成邻近连接域150范围。使用本申请实施例的方法自动引起连接的压缩。下一步是向帧增加邻近范围以支持要求的额外带宽。集合交换节点控制器560通过向帧增加CC/CBR备用连接135来实现之。

[0220] CC/CBR备用连接135不具有到数据源或接收器的链路。其不要求计算备用带宽中所要求的小单元数目，因为在交换栈选择器120中设置的CC/CBR空连接134值自动设置小单元的数目。虽然空信道小单元的数目不要求计算和存储，但是帧中所有小单元范围的总数，包括备用带宽必须等于帧大小F。

[0221] 实现SAIN网络的关键部分在于连接带宽的管理。SAIN网络的重要目标是确保网络接受流量的传输。换言之，改变传统分组网络范式从“接受所有到达的流量，以及丢弃那些无法传输的”到“接受只有确保传输的”和“仅在灾难或某种编程情形下丢弃流量”。

[0222] 在传统网络中，该目标仅在大量过供应时可达到。结果，增加需要的新网络容量以跟上需求就非常昂贵。另外，丢弃包却要求网络转发失败的重新传输而增加了流量强度。

[0223] 幸运的是，SAIN结构能通过遍布于网络的服务品质类实现在接受进入网络的数据前报告带宽可用性的简单子系统。对于网络最坏可能发生的情况是对于不那么重要的流量类告诉用户“所有的连接忙”。等待转发的数据将可存储用于后面的传输而不要求用户重获许可。该结果可达到避免大量过供应的较高性能。另外，SAIN结构能在流量进入网络的转发部分之前进行流量的动态重路由，因此优化了安装带宽的利用。

[0224] 增加路径和超级路径带宽方法的实施例

[0225] 在SAIN网络内分配带宽是很动态的过程。在路径集合层是最动态的，因为该层最靠近用户流量的突发。当流量负载移动时，更高集合层(Higher-aggregation-level)流量发生改变，但这些移动是不够动态的。在路径层的个体流量突发仅代表更高集合层一个处的总流量的仅仅小的部分。

[0226] 当新高带宽流连接出现在源E节点201的用户源数据端口191，可用带宽的迅速扩展必须快速发生。图15的流程图显示容纳新连接所采取的步骤。

[0227] SAIN集合/分解节点对交换类型概要

[0228] 在SAIN网络中，所有交换存在于一个集合交换和一个(或多个)分解交换对中。该结果在于所有连接控制和它们的分配带宽要求仅在每个节点对之间通信。图13显示一般集合交换501，一般分解交换502，以及互联元件(Interconnecting Elements)599。图中的箭头表示数据传播以及控制信息从一般集合交换501到一般分解交换502的方向。响应信息从一般分解交换502向一般集合交换501出现。互联元件599可包含下一节公开的各种集合层的一般集合交换501和一般分解交换501之间的其他成对交换。

[0229] 本申请公开的SAIN网络中有四种类型的集合交换和匹配的分解交换。所有交换的一部分是交换栈选择器120。SAIN网络的交换子系统包含若干个独立于其网络应用的实体类型。它们是1)集合交换，2)分解交换，以及3)在作为操作对的两个交换之间的双工控制矢量。控制矢量是集合交换和其配对分解交换之间的私有消息承载双向对话。

[0230] 一般及四个子类型集合交换是：

[0231] 1. 一般集合交换501；

[0232] 2. 路径集合交换511a(即,第一层集合交换)；

[0233] 3. 第二层L2集合交换521；

[0234] 4. 第三层L3集合交换531；

[0235] 一般即三个子类型分解交换是：

[0236] 1. 一般分解交换502；

[0237] 2. 路径分解交换512(即,第一层分解交换)

[0238] 3. 第二层L2分解交换522;以及

[0239] 4. 第三层L3分解交换532。

[0240] 除了这些称呼之外,为了简便,集合/分解节点对变成了下面：

[0241] 1. 一般A/D交换对503；

[0242] 2. 路径集合交换511/路径分解交换512对变成路径A/D对513；

[0243] 3. 第二层集合交换521/第二层分解交换522变成L2 A/D对523；

[0244] 4. 第三层集合交换531/第二层分解交换532变成L3 A/D对533。

[0245] 除了列在上面的交换类型,存在一个额外结构以完成SAIN网络关键目标。这就是交叉连接交换540,使用其可互联同层的交换。在某些实施例中,通过集合来自于根据目标E节点的子E节点的流量使用交叉连接交换540在源T节点处互联第二层交换。在其他实施例中,交叉连接交换540可被用于目标T节点处而不是源节点处。

[0246] 一般集合/分解交换对的实施例

[0247] 不象电话网络,现代通信网络必须应对遍布网络的流量强度的快速改变。公众交换电话网络(Public Switched Telephone Network) (PSTN) 只有效处理一种流量—语音。语音呼叫,一旦建立,在一段时间内保持连接,大约几分钟。现代网络不这样工作。就服务品质而言,语音流量是现今流量中的关键部分,但就流量强度而言仅仅是是一小部分。总流量强度在相对短的时间内变化范围宽。另外,使用静默探测(silence detection),其中数据只在某人正在讲话时传送,是现今基于分组语音网络的一部分。不幸的是,一般需要的包几乎大到静默探测不如基于电路的网络有用。在SAIN网络中,静默探测的概念可通过虚拟连接实现。虚拟连接可维持到目标节点的连接并通过使用控制矢量用于控制以替代全包头来激活少许比特的传输。

[0248] 语音流量已成为通信网络中总流量的很小部分。即便如此,在传输面向动作的视频中有推论,改变可用带宽以满足总是变化的需求是重要的。

[0249] 使用网络内的包缓存来集合突发流量可舒缓流量回转,但仅仅在某种程度上。仅将包缓存置于网络边缘并使用1)虚拟连接以及2)快速改变分配给流量集合带宽的动态带宽管理可提供重大网络改进。这样做而不求助于大量过供应是SAIN范式的一个主要优势。本申请的本节显示这样做的基本方法和装置。

[0250] 图13显示遍及SAIN网络的基础结构。一般AID交换对503可应用由互联元件599使用本申请方法提供的带宽,只要该连接元件599可提供足够的带宽以支持一般AID交换对503目标。

[0251] 如图14所示,每个一般集合交换501具有一个帧时钟发生器121,并有三个连接到

它的输入：一个用于帧大小增量122，一个用于系统时钟124，以及一个用于帧复位123。在一般集合交换501和一般分解交换502中都可用的帧复位123，产生系统时钟124和帧复位123输入。该两个参数使用工业上可用的计时和帧同步技术得自网络系统。该参数可在SAIN网络中大量时间中保持静态，就是对于每层集合的稳定帧率。网络可使用现有可行的方法以达到该目的。

[0252] 有多种方法提供每个SAIN交换节点的稳定时钟。目标是确保其互相同步而成为自给式网络(self-contained network)。更大的途径是利用现有方法将节点同步到诸如通用协调时间(Coordinated Universal Time) (UTC) 的共同全局时钟源。IEEE标准1588已证明能在少许纳秒内达到同步。

[0253] 除了同步节点时钟以外，有必要知道所有链路离开和进入节点的帧开始的地方。一种简单方法可在每个T节点中使用同步时钟作为所有隶属到那的E节点的参考时钟。对于源E节点，每个产生的帧可比向外T节点帧的开始时间稍微提前一点到达其父T节点。

[0254] 对于起源于若干个T节点的集合数据，有两种通用方法提供隶属E节点的同步。一个是缓冲引入的小单元流量使得来自于所有远T节点的帧时间对齐以克服链路的传播延迟差异。该方法的缺点是增加了附近T节点的延迟。

[0255] 第二种方法使用上面概述的计时方法，其中所有E节点帧的父T节点帧是时间对齐的。达到这个结果而不将有害时延注入网络的简单方法是测量E节点到T节点的延迟，假设往返延迟是单程延迟的两倍。每个源E节点201提前发送其帧足够远以确保其父源T节点301以时间对齐的方式接收它们。

[0256] 使用示范网络作为示例，有19个远T节点发送数据到其他所有T节点。每个T节点将从所有其E节点接收的数据集合为第二层帧。在目标第二层到路径层的接口处，有20个帧起始时间，来自于每个T节点。将帧对于在源T节点301处的每个源L2集合721对齐不能确保所有有目标L2分解722是时间对齐的。不能保证从一个T节点到另一个的距离是相同的。

[0257] 帧的起始时间仅在E节点对内具有重要性。如本申请后面所公开的，没有必要保持帧起始时间顺着中转节点的路由原封不动。需要处理潜在源E节点201流量的QDR和小单元大小确定用于路径A/D对513所要求的帧周期。这个要求不存在于源E节点201和目标E节点202之间的过渡链路中。测量出的几微秒的帧周期通常涵盖了在路径A/D对513层处的需求。该需求不存在于SAIN网络内除集合的第一路径层以外的部分。将过渡帧分割成很小的段可导致纳秒级或更小周期，这可引起很小的延迟且没有抖动和有意义的延迟变化。

[0258] 帧大小增量122是频繁改变的参数。它是本申请达到SAIN自适应目的方法的重要部分。参数改变的频率与链路的集合层成反比。换言之，帧大小增量122在路径集合层改变最频繁，而在第3层最不频繁。若涉及的交换要求更多的带宽以处理其流量负载，用总量测量的在给定层的帧大小一定增加。高时钟率和相对大的小单元的组合支持该增加。

[0259] SAIN网络的另一个方面是要求每个一般分解交换502内连接带宽寄存器142(见图9b)的内容总是与其配对的一般集合交换501同步。一般集合交换501控制包含于其域内连接带宽的分配。一般分解交换502内的带宽分配从其主一般集合交换501。

[0260] 每个路由内分配给连接的带宽是一般A/D交换对503连接带宽寄存器142内的一组位置。每个位置通过存储每连接域帧小单元的数目表示连接的带宽。这些位置可对于具有几乎常数平均带宽基于流的流量(flow-based traffic)在扩展周期内保持在合适位置。这

些流量包括但不限于语音,流媒体,某类视频,以及web站点嵌入的剪辑。

[0261] 在两个E节点之间的路径A/D对513对内,连接带宽寄存器142位置可为三种状态中的一种。它们是真实状态(即,运行状态),睡眠状态,或虚拟状态。真实状态承载用户流量以及必要的管理和控制平面流量。睡眠状态是可快速转变为真实状态的状态。其可包括,例如发送足够的控制流量到和从睡眠状态交换终端以确保快速真实状态的恢复。路由的虚拟状态在连接带宽寄存器142内具有位置,但分配带宽为0。在暂时无大量流量存在的基眠状态,有足够的控制带宽以传输信息确保数据连接的耐久性(viability)。

[0262] 对于真实状态的适当操作,每个交换对必须具有充足的分配带宽一嵌入呈现给它的连接。这些连接集合成为更高集合层处交换对的连接。当在任意层一般A/D交换对503对内设定一个新连接时,如图15所示的步骤适用。图中的步骤(1)621显示新连接的达成。在步骤(2)622询问的第一个问题是将承载该连接的集合的状态是否是虚拟状态。如果是,于是步骤(3)623一定发生,其中若该连接在睡眠状态或真实状态,则信息可用。总是存在用于在这两个状态之一集合的数据包括步骤(4)624的回答。若步骤(4)624所提问题的回答是no,采取步骤(5)625。步骤(6)626在步骤(4)和(5)625完成后发生。交换节点控制器560软件控制公开于此的流程。

[0263] 图15对于仅和下面讨论的路径A/D对513有关,而不是更高的集合层。SAIN网络可以确定的方式支持各种各样的服务类和服务品质。为说明之目的,讨论将着重于三个服务类(CoS),其中可存在多个服务品质类型。很多其他的方法也可以。最初的讨论包括三个CoS's加上共享超额信息率类(Excess Information Rate class)。

[0264] 图15起始于分析引入连接以确定其CoS类型。步骤(2)632确定引入连接是否是基于时间的。换言之,该连接是时延敏感的吗?对于诸如语音呼叫或视频会议的交互式服务,回答是“yes”。对于某县人不直接参与的交易处理可能更加时延敏感。实际上,在某些金融交易,微秒或甚至纳秒都重要。在人类互动中,竞争性在线游戏要求短传输延迟以满足质量需求。其通常大大短于语音通话可允许的双向延迟,其要求包括传播延迟在内的300或更少的毫秒。(通过地理上静止卫星的双向通话差不多需要500毫秒,该时间已发现是很不令人满意。不幸的是,在某些蜂窝电话系统中的延迟因不可避免的逐跳包缓冲延迟接近或超过这个值。)

[0265] 步骤(3)633流量限制更少一些,它可能仍要求在给定周期的平均带宽,但既没有互动也没有亚秒的UTC确定性(certainty)。换言之,其具有弹性特性。例如,其可为实时播放的记录材料,但可忍受少许延迟起始时间以填充缓存到因间歇带宽饥饿(intermittent bandwidth starvation)而发生的服务中断将不发生的点。大多数广播流量和多媒体流量属于此类。

[0266] 下表包含每类服务所要求的可能时延的估计。这些或更多类型的服务品质目标可包括在服务提供者和其客户之间的服务层协定(Service Level Agreements) (SLAs)中。该表仅仅是一个示例以显示仅有服务类型之间的优先权并不是很好的方式来表明服务类。优先并不能代表用户可以指定什么和观察什么。

[0267] 表1

服务	MEF服务类	不包括传播的符合要求的时延	注释
----	--------	---------------	----

常数比特率 (CBR)	A	10微秒—50毫秒	当交付的带宽没有使用,只要在分配的时间内,其交付的值可重建立,该带宽就是可重分配的。
Web网站搜索	B	双向 (RT) 250毫秒	服务在鼠标点击链接或输入URL地址后按下“Enter”后呈现。
音频/视频广播或实时播放	B	双向 (RT) 500毫秒	在点播或广播连接中点击或使用电视遥控器。
点播流量	B	1毫秒到2小时	当源可用时
E-mail	C	1.0—60秒	点击“发送/接收”
消息	C	双向 (RT) 2秒	按下“enter”键
VOIP	A	双向 (RT) 200毫秒	停止对话/开始侦听
控制矢量	A	1.0微秒—5毫秒	时延确定带宽效率
控制包	A	1—10毫秒	时延确定带宽效率

[0268] 显示SAIN节点物理连接的图

[0269] 参见图16,从源E节点201到目标E节点202的一条路径可使用两个或三个串联链路(tandem links)。涉及两个链路,如图16b所示,其中T节点300既作为源T节点301又作为目标T节点302。图16a显示具有分离源和目标E节点和T节点的基础SAIN网络的整体配置。具体而言,其显示通过连接到目标T节点302的源T节点301连接到若干个目标E节点202的若干个源E节点201。该图显示标注为 T_x 的源T节点301具有到标注为 T_y 的目标T节点302的连接。在SAIN网络中,每个T节点可以单工和双工方式连接到其他所有T节点。标称地,从一个T节点向另一个转发数据是单工过程。伴随地,双工控制消息使用控制矢量在源和目标节点之间传播。在示范网络中,每个T节点转发数据到如图16a所示通过网络连接的其他19个T节点。一个T节点还从一个子E节点内连到如图16b所示隶属于同一个父辈的所有其他E节点。

[0270] 源T节点301到目标T节点302的链路可包括如图24和图25所示并在下面描述的多个中转节点。该中转节点透明地从源T节点301向目标T节点302转发数据。一个T节点300除了其其他功能之外可为一个中转节点。SAIN复用变换产生集合连接可使用小尺寸的小单元。基于PoT的数据率和基于除了2以外基数(radices)的数据率可导致少的交换节点时延。

[0271] 图16b显示作为连接到E节点201和目标E节点202的回程源T节点301和目标T节点302的T节点300。请注意,图16a和图16b中的源E节点201具有隶属于源E节点201的用户源数据端口291,并且具有隶属于目标E节点202的用户目标数据端口292。另外,源E节点201使用标注为ET-Trunk 231的E节点-T节点链路隶属于其父源T节点301。目标T节点302使用标注为TE-Trunk 322的T节点-E节点链路隶属于其子目标E节点202。

[0272] 显示SAIN节点逻辑连接的图

[0273] 图17显示从E节点200到T节点300的逻辑链路既可作为连接到源T节点301的源E节点201也可作为连接到目标E节点202的目标T节点302。如本申请后面所详细公开的那样,图中所示的三层集合可发生于源E节点201内。该集合可达到两个目的。一个是集合所有用户输入数据成集合路径和超级路径,这样可以单个集合发送其到源E节点201的父源T节点301。第二个目的是提供集合以致于父源T节点301可以某种方式操作它,转发数据到目标T节点302用于向它们的子目标E节点202传递。

[0274] 第一逻辑链路可将若干个用户连接集合成如图17(a)所示的源路径集合711。该图显示源路径集合711可包括在如图17(b)和17(c)所示的连接域中更高的集合中。

[0275] 如图17 (b) 所示的示例中,若干个源路径集合711中的每个成为源E节点201内产生的源路径集合711的源L2集合721超级路径中的连接域段。该源L2集合721包括所有的源路径集合711,其转发到父辈为源L2集合721的目标T节点302的目标E节点202。图17 (b) 详述了这种方法。源L2集合721的每段包含源E节点201的地址, E_m 连同目标E节点202的地址, $E_m:E_{15}$ $E_m:E_2 \dots E_m:E_{25}$ 。缩写标记显示后面跟一冒号的 E_m 后面是目标E节点地址。该图示出25个目标E节点202和每个父图标T节点302的实施例。对于目标T节点302而言任何数目的目标E节点是可能的。

[0276] 如图17 (c) 所示的源L3集合731已相似的方式处理源L2集合721。既然这样,该若干个源L2集合721包含所有从源E节点201到所有目标T节点302的连接。每段源L3集合731包含源T节点301的地址, T_x 连同示范网络的20个目标T节点302的地址。在本实施例中,这些地址是: $T_x:T_1, T_x:T_2 \dots T_x:T_y \dots T_x:T_{20}$ 。

[0277] 每个源T节点301转发从其子源E节点201接收的每个源L2集合721。每个源T节点301修改从其子源T节点301接收的原L2集合721成为目标L2分解722。该修改将从一个源E节点到多个目标E节点集合改变为从多个源E节点的一个目标集合。下面公开的交叉连接交换可执行这种修改。每个源T节点301发送已修改的目标L2集合721,其被每个目标T节点302当作目标L2分解722。这是通过如图16a所示的源—目标TT-link341/目标—源TT-link342对发送的复用集合 (multiplexed aggregations)。每个目标T节点302将已修改的目标L2分解722集合成为目标L3分解732,对于每个子目标E节点202有一个。图18显示镜像该集合过程的分解过程。

[0278] 路径集合和分解(第一层)交换对的实施例

[0279] 使用具有总数为500个E节点200和20个T节点300的示范网络为例,图19显示具有一个源E节点201和499个路径集合交换511的E节点200。该交换511连接到网络中的每个目标E节点202,除了E节点自身中的目标E节点202。该499个路径集合交换511连接到第2层集合交换521并接着连接到第3层集合交换531。第三层集合交换531连接到如图16示出的源E节点201的父源T节点301。

[0280] 在图20中,目标E节点202呈现出源E节点201的镜像。但是,在目标E节点202中的路径分解交换512和源E节点201的路径集合交换511之间并没有1对1的对应。在本实施例中,在目标E节点202中仅有一个路径分解交换512连接到源E节点201中的路径集合交换511。所有其他路径集合交换511存在于其他的源E节点201中。这意味着必须要有将源E节点201连接到目标E节点202的另一种方式。其他的方式是下面公开的通过交叉连接交换540的方式。

[0281] 设定和维持用户连接的实施例

[0282] 在SAIN网络形成时,其包含若干个E节点200。每个E节点200能执行源E节点201和目标E节点202的功能。E节点200内的每个源E节点201可连接到网络中每个目标E节点202,除了E节点200内的目标E节点202。同样地,E节点200内的每个目标E节点202可与网络中每个源E节点202连接,除了E节点200内的源E节点201。T节点301根据下面公开的方法配置。

[0283] 当运行时,如图19所示,主机、终端、服务器或网络101通过若干个用户源数据端口291中的一个发送数据。该数据端口连接到图19中所示的若干个入口PUI211内的一个入口PUI。

[0284] 使用入口参数化用户接口 (PUI) 210可产生对于入口E节点控制器221的下述输入:

- [0285] 1. 目标E节点地址,
- [0286] 2. 流量类型,诸如
- [0287] a. 单重播送(Unicast),或
- [0288] b. 多重播送(Multicast),或
- [0289] c. 广播,和/或
- [0290] d. 以太网(MAC地址),或
- [0291] e. 其他定义的地址类型
- [0292] 3. E节点或以网桥的端口号
- [0293] 4. 服务的分配类或时延类。

[0294] 入口PUI211a首先搜索其地址缓存215和连接ID(CID)缓存218以确定引入的包是否与存在于(最近)过去的一个匹配,匹配上述的那些项目。如果匹配,PUI发送该包到以前选择的源分配FIFO缓冲区243,并向入口E节点控制器221发送宣布该情况的警示信号。入口E节点控制器221于是确定分配的带宽是否满足上面列表中的第4项的类目标。若可用带宽的数量允许系统满足第4项的目标,入口E节点控制器221没有动作。若分配带宽多于所需要的,即,若对于清空缓存是足够多的,入口E节点控制器221可减少用于该连接的带宽。减少带宽可通过减少该连接的存储于交换栈选择器120中该位置的小单元时隙数目来执行。

[0295] 对于不存在与地址缓存216中的新连接,网络使用传统以太网、域名系统(DNS)、和/或路由器方法去发现目标E节点202的连接地址。系统内MAC地址和关联的E节点和端口地址的列表可使使用该方法可行。包括大型数据库将互联网URL或其他标准与E节点地址匹配的其他方法是可行的,或可推理得到。入口PUI211a将具有目标E节点202地址的其连接信息发送到入口E节点控制器221,从源FIFO缓冲区池241中指定源分配FIFO缓冲区243到引入的连接。其伴随地分配该源分配FIFO缓冲区243到选择的路径集合交换511内CBR栈553as的一个位置。该源分配FIFO缓冲区243对于CBR栈553是数据源。FIFO总线40将小单元从源分配FIFO缓冲区243转发到交换511,其将小单元集合成为源路径集合711复用数据流。

[0296] 集合用户路径连接

[0297] E节点200内的路径集合交换511将用户连接集合成为源路径集合711。每个路径起始于源E节点201路径集合交换511中并终止于目标E节点202路径分解交换512中。图17(a)显示发源于源E节点201的这样的源路径集合711。该路径作为如图18(a)所示的目标E节点202内的目标路径分解712终止。只要网络在更高层集合内向其目标E节点202转发路径,其为源路径集合711。当其通过更低的分解层时,其变为目标路径分解712。换言之,在源E节点201中的每个路径集合交换511和目标E节点202中的路径分解交换512之间有1对1连接。该目标E节点202可为如源E节点201那样的E节点200,在这种情况下,不需要连接。

[0298] 图20显示在一个目标E节点202处的逆向操作。输入是目标路径712,其中每个起始于位于不同E节点200中的路径集合交换511。每个目标路径分解712连接到路径分解交换512,其改变存储于目标FIFO缓冲区池242的小单元中的复用数据流目标路径分解712。从接收FIFO缓冲区池(Sink FIFO Buffer Pool)62,存储的小单元转发到合适的出口PUI212以及连接到主机、终端、服务器或网络101的端口。这些操作可由出口E节点控制器222执行。

[0299] 参数化用户接口(PUI)210执行两种功能:1)作为入口PUI211,以及2)作为出口PUI212。与这两个功能关联的是地址缓存216和连接识别符缓存218。地址缓存216存储关于

源和目标PUI210连接的当前地址信息。当一个新地址信息呈现于入口PUI211之内时,其存储于源节点201壳(housing)和入口PUI211内。除PUI210以外连接中涉及的除PUI210以外的信息于是就可得到了。与该连接涉及的目标E节点202共享这些信息。也可将这些信息存储于整个城域网和之外可用的数据库中。

[0300] 连接识别符缓存218可存储出现于后续包中不做修改的包头信息。与该信息关联的是连接识别符(CID),表示该信息的小数目的比特。当包进入要求一个新CID的入口PUI211时,入口E节点控制器221和本系统中其他处理器可提供一个CID。该入口E节点控制器221根据需要将具有相关信息的新CID发送到一个或多个E节点控制器。该操作相似于IETF鲁棒报头压缩(IETF Robust Header Compression) (ROHC) RTFs,因其详细的设计而可用作为标准。

[0301] CID还可在适当的地方成为网络宽度的数据库的一部分。这些缓存减少与数据包一起发送报头信息。控制矢量隐式包含寻址消息段,其可代替使用在其他网络中的传统控制包。这种方法提供确定性的控制、消息时延,并节省源和目标E节点200和T节点210之间的带宽。

[0302] 图21有两个目的。一个是接合要求所有来自于源E节点201的路径终止于不同E节点200中的目标E节点202的连接。另一个目标是接合要求所有来自于不同E节点200中的源E节点201的路径终止于单个目标E节点202的连接。

[0303] 为本段之目的,引入下面描述的用于E节点200和T节点210的寻址符。这个寻址信息用于参考。对于这些寻址符,假设每个E节点200具有其父T节点301域内分配的6比特数。假设每个T节点301可支持上到64个E节点200,独特的12比特E节点200地址是 $64 \times (\text{T节点地址}) + (\text{E节点地址})$ 。开始小的网络可调整为包含64个T节点210,其中每个可调整为64个E节点200,从而有4,096个E节点200。通过这种方法,对于一个E节点,其父T节点是

$$[0304] \quad = \text{INT} (\text{E节点地址}/64) 。 \quad (4)$$

[0305] T节点域内6比特节点地址是

$$[0306] \quad = (\text{E节点地址}) \text{ MOD } 64 \quad (5) 。$$

[0307] 本申请中图21和其他图对那些有用的T节点和E节点使用这种编号方案。请注意使用6比特编号仅在本申请中用于示例之目的。其他寻址标准在进一步的实现中是可行的。

[0308] 图21显示连接到标注为T06的其父源T节点301的源E节点201。其还显示终止于其父辈是标注为T11的目标T节点302的所有目标E节点202。对于作为实施例的示范网络,在T06子网络中有25个源E节点201,并且在T11子网络中有25个目标E节点202。该图示出每源T节点301的示范网络的12,475 (25×499) 个路径的625 (25×25) 个互联,因此有249,500 ($20 \times 12,475$) 个可能路径。每个源E节点201通过ET中继231连接到其源T节点301父辈。本实施例中,每个源E节点201具有在E0384到E0408范围内的标签。

[0309] 连接的标注为T11的目标T节点302已配属标注为E0704到E0728的目标E节点202。在源T节点301 T06之间的该TT链路连接到命名为源到目标(Source-to-Destination)TT链路341或目标从源(Destination-from-Source)TT链路342,其取决于图25所示的交叉连接交换540的位置。若交叉连接交换540在源T节点301内,该链路是源到目标TT链路341,若其位于目标T节点302内,则该链路则为目标从源TT链路342。

[0310] 示于图21中的三个集合层具有下面的特性:

[0311] 1. 路径310层(第1层)处理用户数据复杂性定位地将路径集合交换511连接到路径分解交换512。该复杂性不与两个端交换之间的底层比特的传输混合;

[0312] 2. 第2层超级路径320层具有三个目的;

[0313] a. 一个是互联若干个路径310层,即T节点300对的子层连接;

[0314] b. 另一个目的是将若干个子E节点200连接传递到交叉连接交换540以将单个源E节点201到所有其他目标E节点的连接重定位成许多个源E节点201到单个目标E节点202;

[0315] c. 再一个目的是提供两个单工连接中每一个中调整路径集合710带宽的方法;

[0316] 3. 第3层超级路径330层具有SAIN网络内基于其位置的若干个目的。总的来说,它们在功能上集合不同的若干个L2集合720。具体而言,其他某些目的如图21所示:

[0317] a. 将来自于源E节点201的所有源L2集合721集合到父源T节点301,在图中标注为L3A1=>L3D1,

[0318] b. 将目标L2分解722集合到目标E节点202,在图中标注为L3A3=>L3D3;

[0319] c. 将所有来自于源T节点302的源L2集合721集合到目标T节点302,在图中标注为L3A2=>L3D2。

[0320] 上面3c中L3A2=>L3D2连接的角色既可为两个T节点300的直接链路,也可,更可能(在具有超过最小数目节点的网络中)是通过串联中转节点传递连接。既然这样,就有若干个L3D2=>L3A2的转发。换言之,每个中转节点包含连接到一个出口L3集合交换531的入口L3分解交换532。

[0321] 在图23中,请注意每个源E节点201包含从源E节点201到不同目标E节点202的路径。另外,每个目标E节点202包含来自于不同源E节点201的路径。

[0322] 当对新连接带宽可用时,是一般集合交换501中集合交换节点控制器560的责任来以确定性方式应用该带宽。对于新用户连接,该集合交换节点控制器560使用从参数化用户接口(PUI)210收集的关于用于路径层集合的连接类型的信息。另外,才路径层,网络中可有若干个服务类。图15b示出这种实施例的方法。

[0323] 路径帧同步

[0324] 影响用户连接的帧同步在SAIN网络的两个地方发生—在路径集合交换511处以及路径分解交换512处。可得到几种方法,一种是发送相似于在基于分组系统中使用的诸如以太网的帧前导序列(frame preamble)。这种方法要求相当大的开销处理。

[0325] 另一个实施例要求源T节点301的子源E节点201将其时钟与源T节点301同步。该目的是为了确源E节点201中的每个路径集合交换511的帧起始时间到达成为分配小单元空间,其在源T节点301和其目标T节点302拍档之间的主帧(master frame)中。以这种方式,帧同步的责任可属于在其他方法中的L3 A/D对533。

[0326] 第2层集合和分解交换对的实施例

[0327] 对于示范网络,每个源E节点201包含20个第2层集合交换521,对于每个目标T节点302有一个。每个第2层集合交换521集合所有起源于源E节点201并终止于一个目标T节点212的所有路径。在示范网络中,19个第2层集合交换521将25个路径集合到目标E节点3202;对于第20个第2层集合交换521,回程到其父辈也是源E节点201父辈的24个目标E节点202。

[0328] 图17(b)显示数据集合内用于第2层源超级路径721的上下文。图18(b)显示数据集

合内第2层目标超级路径722的上下文。

[0329] 图22a显示集合流程的一个实施例,在图中顶部的第2层集合交换521中,若干个源路径集合711终止于目标E节点202,其父辈也是目标T节点302T00。每个路径集合交换511的输出终止于互联小单元缓冲区250。这既可是具有小数目级的FIFO缓冲区,也可为单级缓冲区(single stage buffer)。标称地,只要路径帧段对于源路径集合711存在,其就是单级缓冲区。由第2层集合交换521产生的第2层源超级路径721通过互联小单元缓冲区250连接到第3层集合交换531。若L2帧段对于第2层源超级路径721存在,单级缓冲区通常就足够了。

[0330] 如图22a和图22b所示,若干个源路径集合711的每个可成为在目标E节点202处目标路径分解712。图22b显示通过第3层分解交换532和第2层分解交换来恢复目标路径集合712(从源节点来看,源路径集合711)的分解。这样地,源路径集合711连接到目标E节点202中的路径分解交换512。每个路径分解交换512传递包含在目标路径分解712中的小单元到目标E节点202中的用户目标数据端口292,以逆向方式到由上面所描述的路径集合交换511执行网络输入的集合。

[0331] 第3层集合和分解交换对的实施例

[0332] 每个源E节点201将其第2层源超级路径721流量集成第3层源超级路径731。一个源超级路径731包含所有起源于源E节点201到如图17(c)所示网络中的所有目标E节点202的所有路径。在示范网络实施例中,第3层源超级路径731集合来自于20个第2层集合交换521的20个第2层源超级路径521。如下面所讨论的,其第3层目标超级路径732同伴出现在源E节点201的源T节点301父辈内。其变换成为如图18(c)所示的以单个目标E节点202为中心的形式。

[0333] 在示范网络实施例中,该第3层源超级路径731包含来自于每个源E节点201的所有499源路径711并发送到如图22a所示的ET—中继231中的其父源T节点301。除了用户流量之外,ET中继231还可携带控制信号。

[0334] 源T节点中交叉连接交换的实施例

[0335] 在每个源T节点301内,对于网络中的每个T节点可有一个交叉连接交换540。每个交叉连接交换540专用于转发流量到网络中一个目标T节点302,其在示范网络中,将流量从1对多转换为多对1。就是说,源T节点301接收的来自于每个源E节点201的流量可直接到目标T节点302的许多子目标E节点202。该交叉连接交换540转换流量,以致于许多源E节点201根据单个目标E节点202集合在一起。图24显示嵌入于源T节点301中的这样一个交叉连接交换540。

[0336] 源T节点301包含其他多个物体之间的若干个第3层分解交换532,如图24所示。在示范网络中,有25个第3层分解交换532,一个用于连接到父源T节点301的每个源E节点201。每个第3层分解交换532在一个源E节点201中与一个第3层集合交换531配对。ET—中继231连接每个对从源E节点201(包含第3层集合交换531)到其父源T节点301(包含第3层分解交换532)。

[0337] 每个配对的目标L3分解732分解示范网络中的该25个第2层目标L2分解722。每个第2层目标超级路径722包含去往连接到一个目标T节点212的目标E节点202的路径流量。该父源T节点301将每个第2层目标超级路径722连接到父源T节点301中的交叉连接交换540,其转发流量到正确的目标T节点302。

[0338] 源E节点201中的路径集合交换511将所有节点路径封装到网络中所有其他目标E节点202。但是,每个目标E节点202中的一般分解交换502接收来自于网络中所有其他源E节点201的路径。交叉连接交换540可用于重组织这个路径流量以完成这个目标。

[0339] 图24显示示范网络中具有20个第2层目标分解722的25个第3层分解交换532的每个的描绘、每个第2层目标L2分解722连接到互联小单元缓冲区250,其依次地,连接到父源T节点301中的一个交叉连接交换540。

[0340] 图24显示用于父源T节点301T06和目标T节点302T11的交叉连接交换540。该交叉连接交换540包含25个第2层分解交换522,示范网络中的每个源E节点201有一个。每个交换的输入是来自于如图24所示的25个第3层分解交换532的输出。

[0341] 来自于第2层分解交换522的输出连接到交叉连接交换540中的一组第3层集合交换521的输入侧。第2层集合交换521的复用输出连接到如图24所示的第3层L3集合交换531。这些流程均在每个源T节点301内发生。

[0342] 图25显示SAIN交换之间的交叉连接的不同类型。这样的话,该交叉连接仅存在于第2层。对于示范网络,目标T节点302中20个L3分解交换532的每个与存在于20个源T节点301的每个中的L3集合交换531配对。既然假设有来自于20个源T节点301的每个的25个第2层连接,总共有500个源L2集合721信号,其从20个L3分解交换532连接到25个L3集合交换531,该集合交换531的每个连接到25个目标E节点202。

[0343] 用于连接串联轴节点的实施例

[0344] 图13显示一般A/D交换对503的基本排列。互联元件599可分离诸如路径A/D对513的低层交换对。所有大型网络必须以某种方式集合数据以相对互联低层入口和出口节点。换言之,不象传统分组网络,使用诸如多协议标签交换 (Multiprotocol Label Switching) (MPLS) 的协议的数据集合通过给包附上标签来提供数据集合,该标签可由串联轴节点路由器转发。这些路由器必然是非常高速且昂贵的。在SAIN网络中,在串联轴节点中数据集合交换发生在电路层,其中交换连接的很不贵的电路可处理成百上千个用户基于分组连接的单个集合。

[0345] 图13的一个重要方面暗示,在诸如一般A/D交换对503对的低层的成对交换连接不要求在互联元件599内包的复制。互联元件599中所要求的是在更高集合层处一个或多个交换对的串联连接。每个这样的集合不要求显式转发包。每个集合仅要求简单的电路交换。

[0346] 在串联轴节点中,设定一般分解交换502连接以接收来自于引入连接的数据集合。换言之,串联轴节点提供与上行 (upstream) 一般集合交换501配对的一般分解交换502。这样的交换配对可为来自于上行节点的T节点到T节点L3集合交换531连接。该串行节点包含一个或多个L3集合交换531,其从L3分解交换532向一个或多个下行链路节点转发数据集合。

[0347] 转发到多于一个下行链路节点发生于多重广播连接中。用于控制这样的流程的机制可使用控制矢量,该控制矢量由其他T节点管理,其中每个这样的T节点具有到涉及在其中的T节点连接的控制矢量。

[0348] 用于建立低时延高容量网络的实施例

[0349] 经典时分交换网络传递一帧信息到后续节点,其中帧模式从一个网络链路到另一个保持同样的方式。例如,在公众交换电话网络内就是这样的。在SAIN网络中,在每个串联轴链路内不需要复制帧模式。这打开了途径以在中转节点处最小化交换时延以及产生诸如在

多个光学中继之间分割高速数据的其他优势。

[0350] 图26a、图26b、图26c和图26d显示演示用于在SAIN网络中最小化时延的方法的四个例子。每个示例显示为本申请中假设的示范网络而定义的三个集合层次的每个的参数。虽然本申请中描述了三个集合层次并参考图26a—d,但是使用公开方法的其他集合层次也是可能的。这样,在所有SAIN网络实施例中,集合层次的数目不必为3个。如图26a—图26d所示是对于交换对连接的独立参数和依赖参数。

[0351] 第一参数是系统时钟率和其逆,时钟周期。如图26a和图26b所示,时钟率可对三个集合层的每个相同。图26b和图26c在每个集合层具有不同的时钟率。

[0352] 在每个示例底部的路径集合交换511处有三个参数:交换的帧率,小单元大小,以及帧段的数目。显示的其他参数得自于这三个参数加上系统时钟率或集合层时钟率。

[0353] 图26a和图26b都使用每秒125,000帧(即125 Kfps)的帧率,其帧周期为8,000纳秒(即8微秒)。小单元大小为1比特。包括其他的参数,结果是量子数据率(QDR)为125Kbps。拼成一帧的时钟周期数目(Nbr Clock Periods),示于Min右手边的栏中。集合带宽=帧周期 \times 系统时钟率。换言之,时钟脉冲的数目= $8 \times 10^{-6} \times 1.5625 \times 10^9 = 12,500$ 。帧段可为规定最小集合带宽的时钟周期数目的整数约数。12,500有很多整数约数。对于这个示例,每帧的帧段数目选择为50。

[0354] 该每帧50个子帧的每一个具有250个时钟脉冲,并且该子帧周期是160纳秒。另外,若每个子帧填充以单个小单元,总带宽等于 $50 \times 125,000 = 6,250,000$ bps,即6.25 Mbps。这现今成为用于路径A/D对513的最小集合带宽。其对QDR不做任何事,其中连接可为125Kbps的任意整数乘积。其仅影响附加容量的带宽增量,即6.25Mbps的任意整数倍。对于这组参数的最大集合数据率是1.5625 Gbps,其对于1比特小单元是系统时钟率。

[0355] 在一个L2集合交换521处,其帧周期现在是来自于路径集合层的子帧周期。在本示例中,在第2层为160纳秒,L2帧率与路径层的最小集合带宽相同。在层2的结果是:

[0356] L2小单元大小选择为16,导致 $L2QDR = 16 \times 6.25 = 100$ Mbps。

[0357] L2最小集合带宽=帧段 \times QDR= $50 \times 100 = 5$ Gbps。

[0358] 最大集合带宽=L2小单元大小 \times L2时钟率= $16 \times 1.5625 = 25$ Gbps。

[0359] L2子帧周期=L2帧周期/12帧段= $160/50 = 3.2$ 纳秒。

[0360] 在L3层的帧周期为从第2层输入的3.2纳秒。选择的小单元大小为64比特并且L3帧段的数目为1。对于这个值,L3子帧周期仍然为3.2纳秒,与L2子帧周期以及最大集合率100Gbps相同。图26a显示其他第3层的结果。图26b显示对于1.28Gbps系统时钟率以及125Kbps帧率的结果。其结果与图26a相似。

[0361] 两个图的这些结果看上去有趣而且有用,但其没有充分利用SAIN算法的2的次方数特性的优势,其可导致帧内均匀分布的小单元。图26c和图26d显示使用该算法充分发挥其优势的结果。

[0362] 图26c和图26d显示具有1比特小单元和64Kbps数据率的路径集合层交换。两个图中帧段的大小设定为128,导致向外的子帧周期在L2交换的输出处为122.070纳秒,以及在L2交换的输出处为0.954纳秒。在L3交换中,帧段均设定为1,这可导致L3帧段与L2的相同。换言之,除了L3小单元大小这一个以外的所有输入参数在两个示例中相同。在图26c中,L3小单元大小为8比特;在图26d中,其为32比特。这显示参数可导致各种结果。实际上,下一

节,使用2的次方数长度子帧以最小化时延的实施例显示方法,这些方法通过简单策略,不需要仅使用诸如图26c和图26d中显示用于帧段的那些参数的2的次方数参数。

[0363] 仅随L3小单元大小改变而改变的依赖参数是L3最小集合带宽和L3最大集合带宽。如图26c所示,最小和最大集合带宽分别为8.389Gbps和33.554Gbps。在图26d中,它们是33.554 Gbps和134.218 Gbps。在两个示例中,可用最大集合带宽与最小集合带宽的比为4。换言之,除了最小和最大集合带宽外,当每小单元的比特数目改变时所有依赖参数保持不变。

[0364] 一般技术人员可通过建立交换元件实现公开于此的SAIN交换,该交换元件改变1)当流量负载改变时通过自动操作,或2)通过操作者从管理控制站进行改变。例如,来自于源或到接收器的小单元的大小可设定为备选表格中的一个。

[0365] 公开的方法和装置具有重要的副作用,能实现可调整到极端高数据率的交换。既然该系统可使用2的次方数相关的小单元大小优势。例如,交换可通过3225Gbps光纤波长使用引用于此的业界发展水平的同步方法发送32比特小单元的每个比特。合并的总数据率是800 Gbps。增加小单元尺寸到64比特和64光纤波长,结果是每秒1.6太比特(terabit)(Tbps)中继。本实施例中,不需要处理包的分离和重组。如总是在SAIN网络中那样,包仅存在于入口和出口端口。随着相干光中继(coherent optical trunks)的出现,将导致使用单个波长的每秒几百千兆比特的数目,单个光纤内的数据量显著增加。

[0366] 即使公开于此的大部分假设在路径和第2层集合层,小单元大小可为1bit,其他小单元大小也是可以的。随着半导体更高时钟率的出现,这将是一个重要途径以发挥SAIN复用算法用于序列数据源和接收器的优势。但是,许多源和接收器起始于多比特字(multi-bit words),具体是8、16、32和64比特宽。序列化这些字是许多上下文中普通的操作方法,并且SAIN网络可为其中之一。本方法的优势在于就随机网络内的简单性而言容易理解。但是,该流程也存在弱点。

[0367] 即使初相小,仍然有与该流程关联的拆卸和重组装时间。在这些情况下,本申请的SAIN方法可导致网络内端到端时延的极端最小化。在诸如半导体的应用中,其中距离很小且光传输最终减少要求的功率,该方法可找到应用。在更长距离的应用中,应用SAIN方法可导致在几个皮秒中测量端到端时延。这可导致三角化物理位置的能力以有效竞争而不依靠GPS和其他基于卫星的方法,这些方法具有内在的可靠性和生存性问题。影响精度的仅有问题可由温度和随时间的地球空间变化引起。这是一个可以追溯到很多年的良好的研究领域。

[0368] 当网络(或子网络)实例化后,可以如图26a所示类似的初始最小集合带宽设置来设定节点交换。这样做使得网络(或子网络)内的每个集合交换511、第2层集合交换521和第3层集合交换531供应足够的备用带宽以确保平滑的启动操作。

[0369] 用于连接成对交换的实施例

[0370] 在将一个SAIN交换连接到下行链路交换中有两种不同的情况。在一种中,一个交换是一般集合交换501,其连接到成对一般分解交换502。在另一种情况中,交换既可为一般集合交换501,也可为一一般分解交换502,其中其下行链路交换是在最后一节中描述的过渡(串行)交换。

[0371] 第[0063]段的SAIN变换算法的基本原理详述了图4、图5、和图6用于成对一般A/D

交换对503的内存印象图。本节描述的结果对于路径A/D对513是非常重要的。

[0372] 内存印象图可在成对交换中彼此匹配。另外,该系统可同步路径分解交换512帧的起始时间在接收到路径集合交换511起始时间后立刻开始。换言之,同步过程补偿了源发送时间和目标E节点到达时间之间的所有网络传播的延迟。唯一的限制是从路径集合交换511接收的每个小单元的小单元边界发生了,以致于在到达小单元中接收数据的及时发生以将其置于伴随向外的小单元位置上。

[0373] 如下面所描述的,描述于SAIN变换算法的基本原理中的信息可使得一般技术人员确保本实施例发生的适时性。就是说,若连接要求不是PoT乘积的多个小单元,只要连接开始于两个相关交换栈中的相同连接域小单元位置就没有问题。

[0374] 1. 交换可通过把边界内的小单元当作一个单位的交换来使用PoT边界。每个PoT段将PoT边界当作来自于一般集合交换501并返回到一般分解交换502处的原先边界的交换下行流(downstream)中的子帧边界。

[0375] 2. 只要每PoT段p是相同的,这样的交换除单个小单元缓冲区外不加入时延。换言之,只要交换的数据率是基本数据率的整数倍,基本数据率定义为每PoT段一个小单元。

[0376] 3. 若每PoT段小单元的数目不相同,二PoT段(two-PoT-segment)FIFO缓冲区要求p+1个小单元以保证每个向外的小单元时隙具有一个引入的小单元。

[0377] 4. 结果是只要基本数据率保持不变即使当整数倍改变时,端到端时延保持常数。

[0378] 图27显示连接域内四个连接,涉及长度不是2的次方数(其中长度为29小单元)的帧。有3个非2的次方数长度的连接;两个连接中每个包含5个小单元,标注为“A”和“B”;以及连接“D”具有3个小单元。另外,一个连接是2的次方数长度;连接“C”具有4个小单元。这些4个连接示出方法,SAIN网络可使用该方法导致伴随时间域中对于其传输数据的低和确定性时延。下面的结果从所做的假设中获得:

[0379] 1. 帧(或子帧)内偶数个小单元使得(子)帧的一半小单元存在于每一半(子)帧中;

[0380] a. 证据:给小单元编号,从0到帧长度减1,一半小单元是偶数而另一半是奇数。颠倒小单元数目的二进制表达的顺序导致一半小单元起始于“0”并且另一半起始于“1”;

[0381] 2. (子)帧中奇数个小单元导致一半帧比另一半帧多包含一个额外小单元。

[0382] 图27显示对于四个连接中每个的均匀分布输入信号所要求的前置时间(lead-time)的数量。四个中仅仅一个显示输入信号“C”信号可正好在SAIN时分帧开始之前发生。这样是因为“C”信号运行于基本数据率,其仍可正好在时分帧开始之前发生,但只要整数乘数为2的次方数。换言之,若整数倍不是2的次方数,则要求FIFO缓冲区。缓冲区的大小由小于整数倍的最大2的次方数确定。在周期间隔中到达的由这个2的次方数倍数确定的所有小单元被缓冲。例如,假如基本数据率是每帧4个小单元,并且引入数据率是每帧24小单元。小于24的最大2的次方数是8。因此,FIFO缓冲区必须拥有3个小单元以保证不存在下溢。

[0383] 使用2的次方数长度子帧以最小化时延的实施例

[0384] 本节适合于SAIN网络中所有成对的交换,但对成对串行节点交换尤其重要。以正确的方式使用2的次方数子帧除了仅依靠子帧的长度小之外可最小化时延。

[0385] 图27显示SAIN连接域/时间域帧以示出SAIN算法的备选用法。这些例子的主要目的是公开导致最小化SAIN网络中端到端时延的方法和装置。

[0386] 图27 (a) 显示支持5个不同PoT段长的62小单元帧,该5个不同PoT段长为32、16、8、4和2小单元。该PoT段长在图中是降序的。结果是时间域中的一组连接,其中小单元等距分布几乎没有抖动。[既然数62不是2的次方,从2个小单元等距分布的状态将有一个小单元的位移]该示例在真实世界中并不典型,因为SAIN网络的链路数据率在城域网和其他核心网内通常为高。如图27b和图27c所示,对于来自于第2层集合交换521的路径AID对513,帧中可用小单元的最大数目,对于每秒125帧和128帧的例子,巧合地都为每帧10,240 小单元。

[0387] 等距小单元的主要优势在于中转节点交换不需要等待整个PoT段确定应该到达的所有小单元已经到达。

[0388] 对于连接而言,小单元的等距分布依赖于两个参数,它们是:

[0389] 1. 每PoT段小单元的数目是一个2的次方数;以及

[0390] 2. 每帧PoT的数目是连接域的虚拟帧长度中连接的起始位置的约数。

[0391] 为进一步说明,假定PoT段成为更小的PoT段以用于分配低速流量。图27 (b)、图27 (c) 和图27 (d) 示出主要的问题。

[0392] 结果是:

[0393] 1. 可能的抖动最多在一个链路数据率位置(加上一个链路时钟周期),以及

[0394] 2. 任何固定时延,虽然小,是确定性的和可预判的;网络控制器可通过端到端时延测量来解决;

[0395] 3. E节点端口的位置可用厘米和分米测量,其中存在高速链路。测量的精度可超过GPS的精度。

[0396] 图27 (b) 是有47个小单元的帧大小。连接“A”具有32小单元PoT段连同8小单元、4小单元、2小单元和1小单元PoT段。以长度的降序排列,每段提供近等距分布(near-equal spacing)的小单元。其原因在于每个子帧的起始位置开始于边界需求,其在上面的段的列表的第2点中公开。

[0397] 图27 (c) 显示将32小单元PoT段进一步分割成更小的PoT段,这些段每个分别有18、8、4和2个小单元加上两个1小单元段。这仍然是降序排列的PoT段大小使得等距再次产生。其进一步显示显著的无序。

[0398] 图27 (d) 显示当PoT段以长度降序排列时可发生的事情。有一点抖动存在。例如,总是在图27 (c) 中16个PoT段的每一个的最左边位置上的连接“A”现在表现出其PoT段从左到右的抖动。

[0399] 明显地,不是所有连接在真实世界中都将是2的次方数长度。在路径AID对513对连接的情况下,这不是一个大问题。要求FIFO缓冲区将引入的包匹配到具有向外连接的源E节点201是一次性发生的事情。对于具有超过每秒千兆比特数据率的传输链路具有非常大的最大基本数据率,其可为大的2的次方数。它们子分成2的更低次方可保证公开于此的低时延中转节点操作。



图1

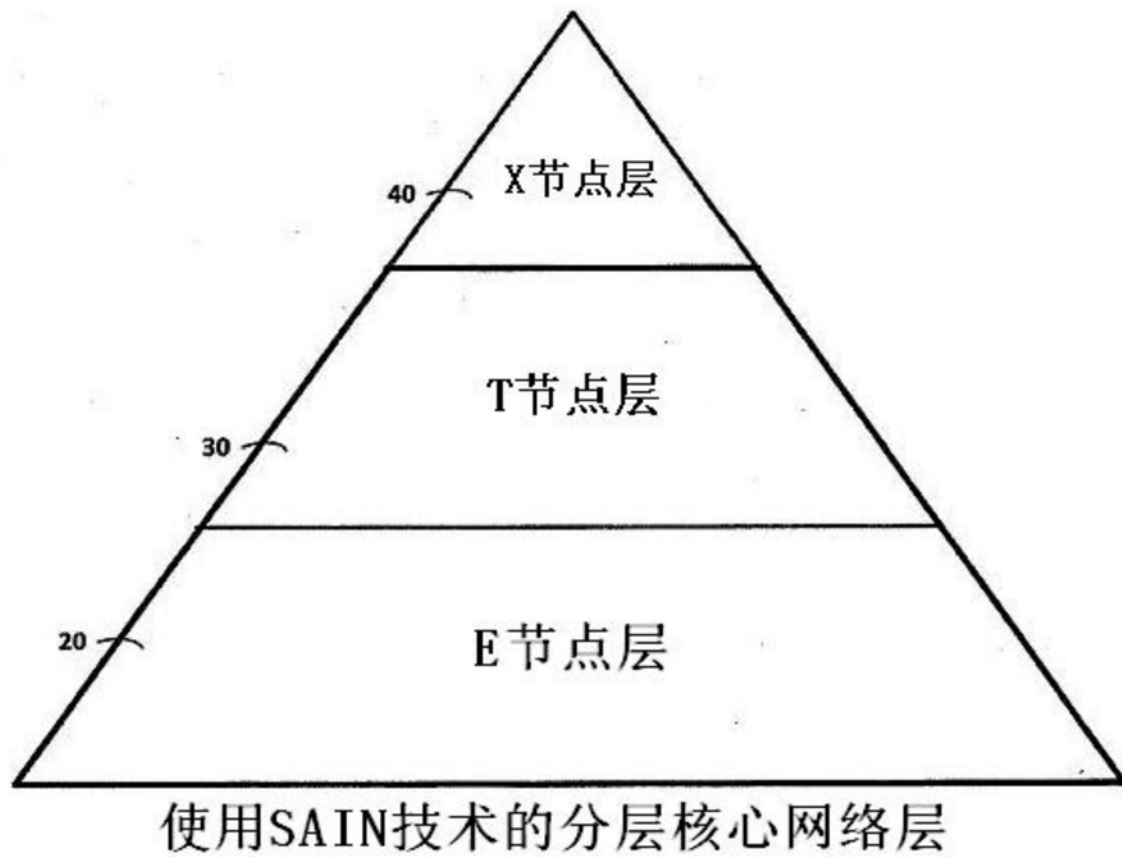


图2

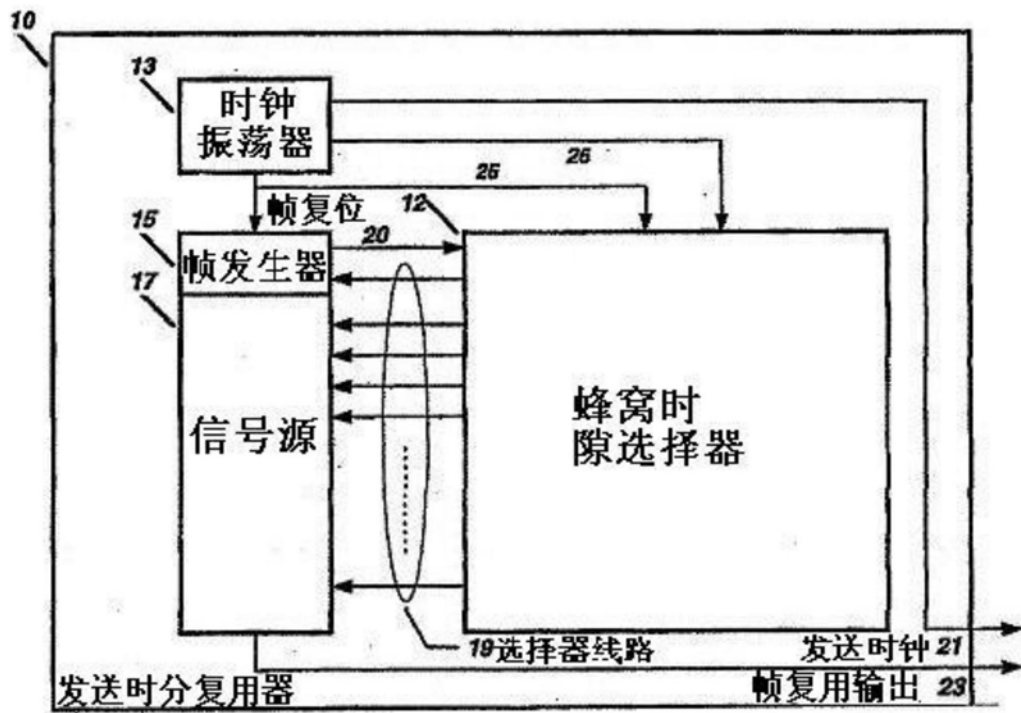
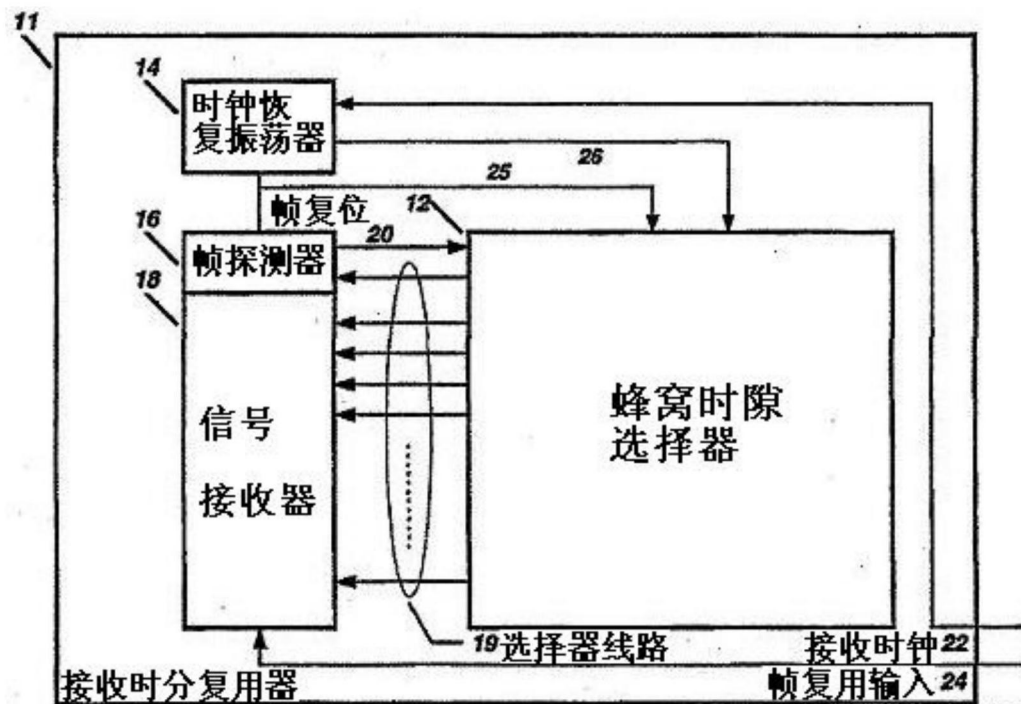
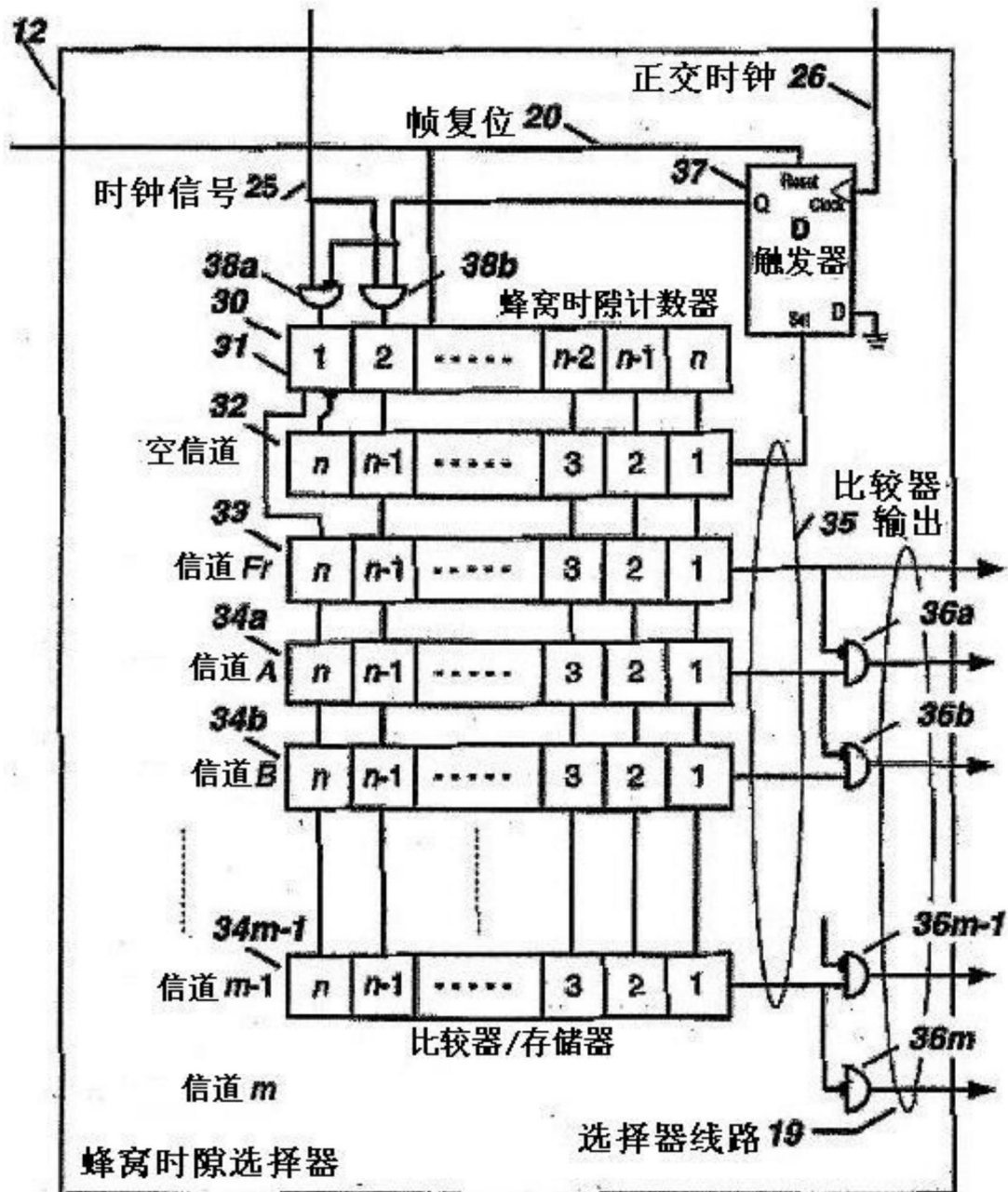


图3a



蜂窝时隙复用器 现有技术

图3b



蜂窝时隙选择器

现有技术

图3c

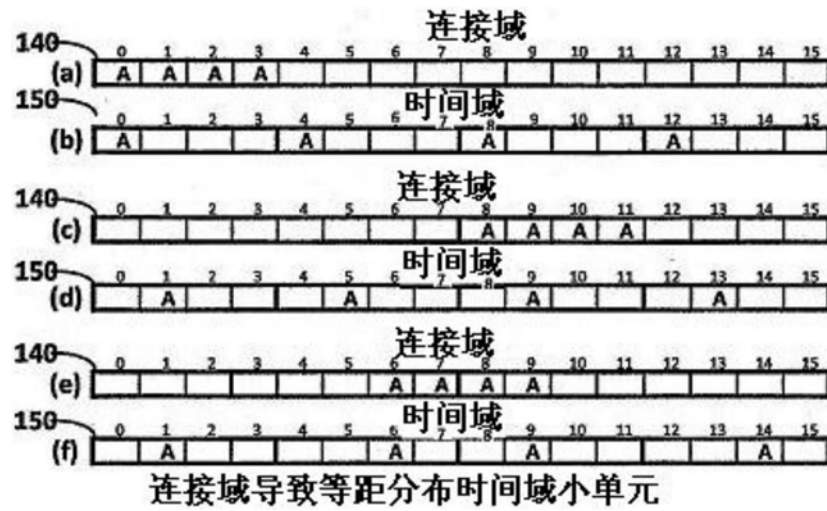


图4

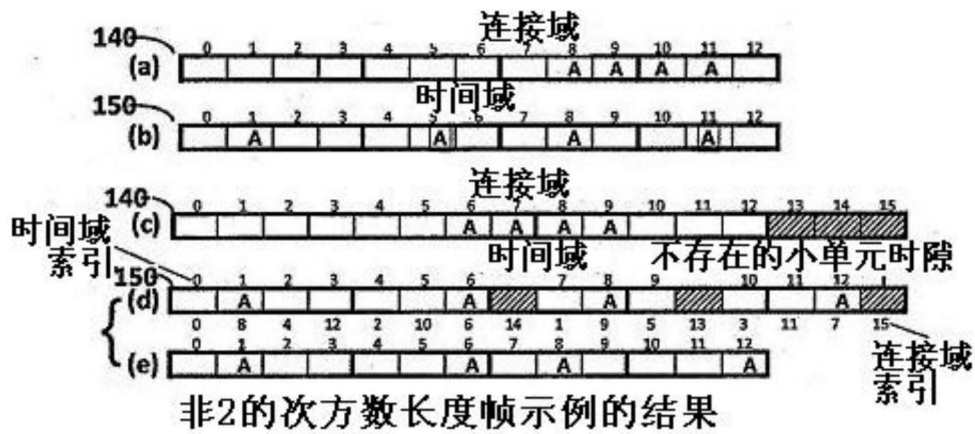


图5

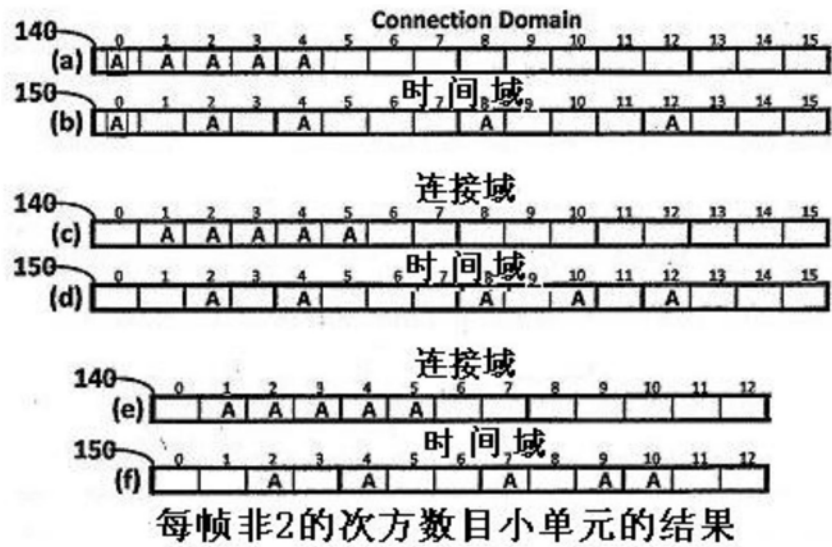


图6

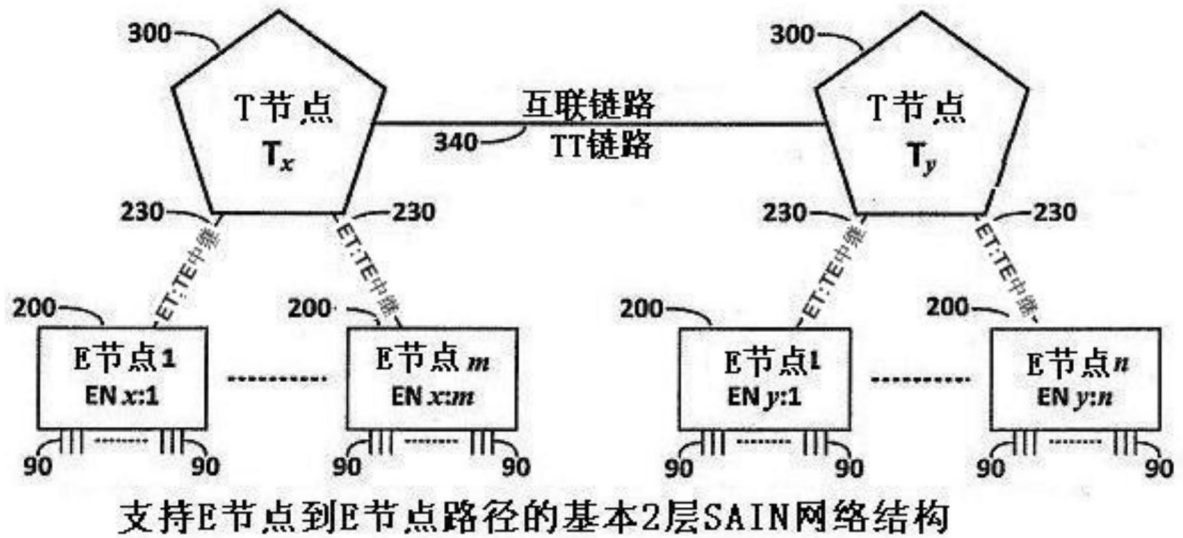
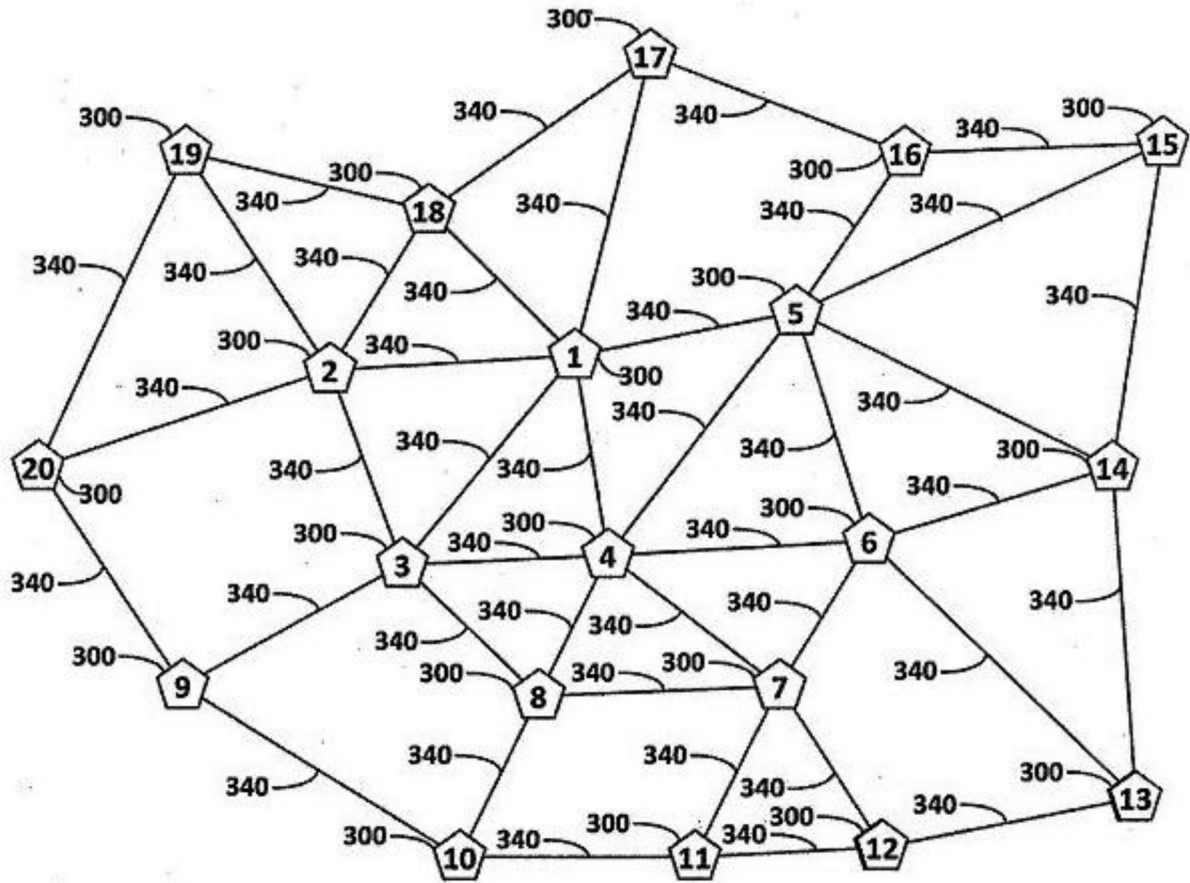


图7



具有20个T节点和40个双工中继的城域网示例

图8

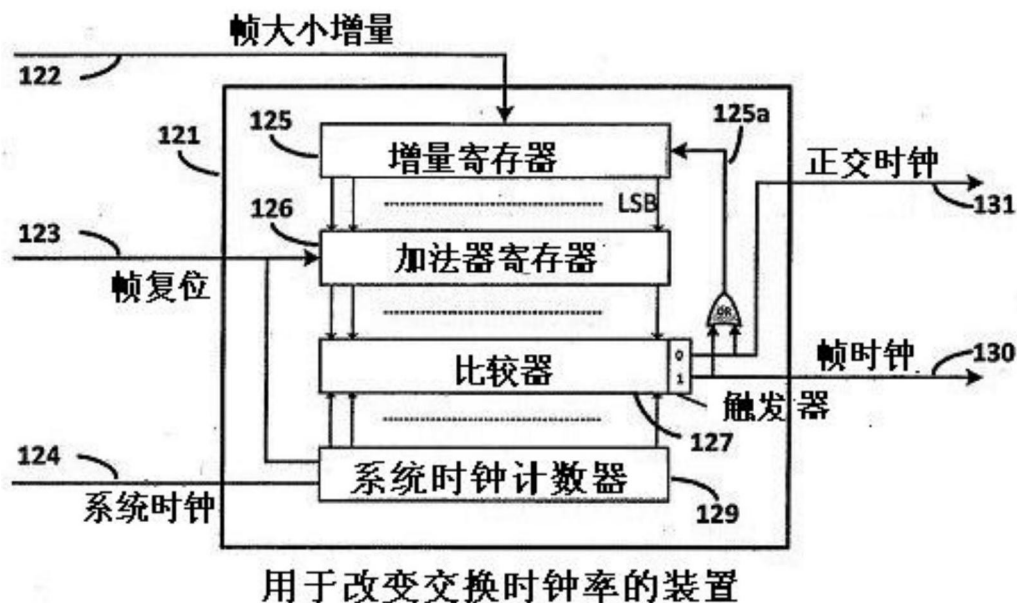


图10a

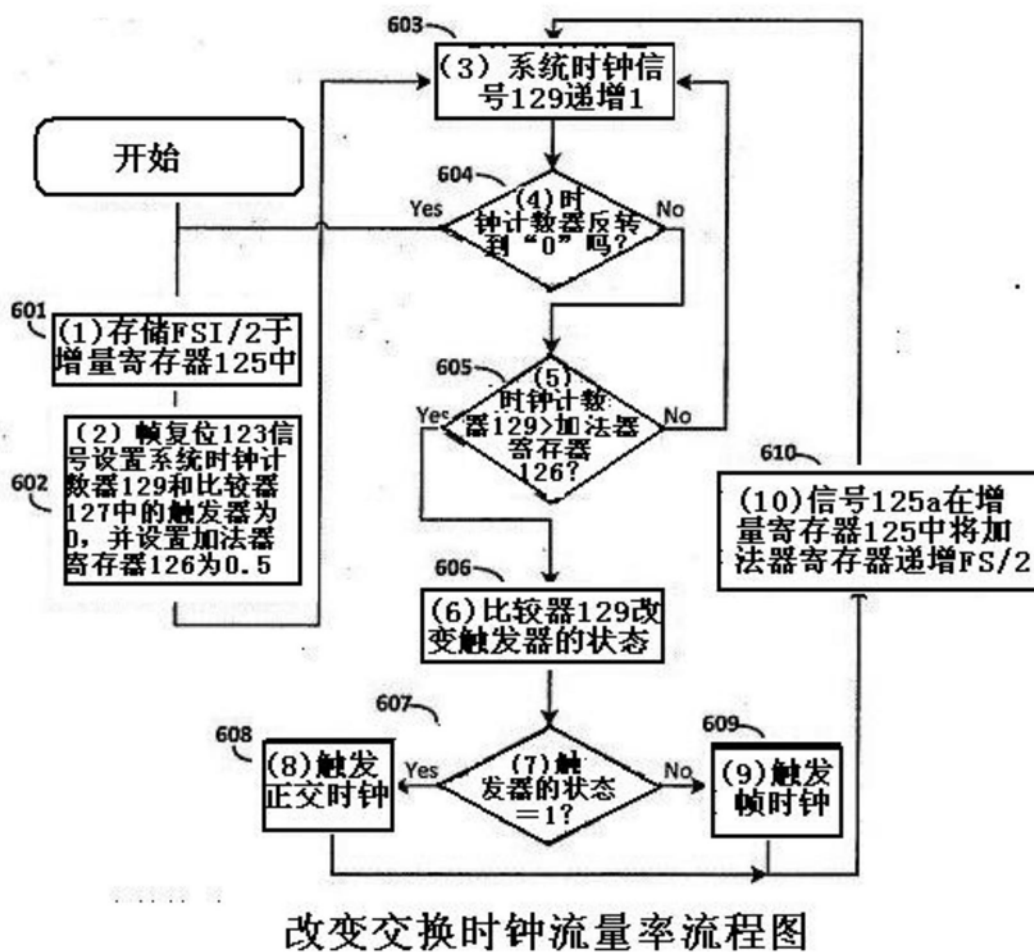
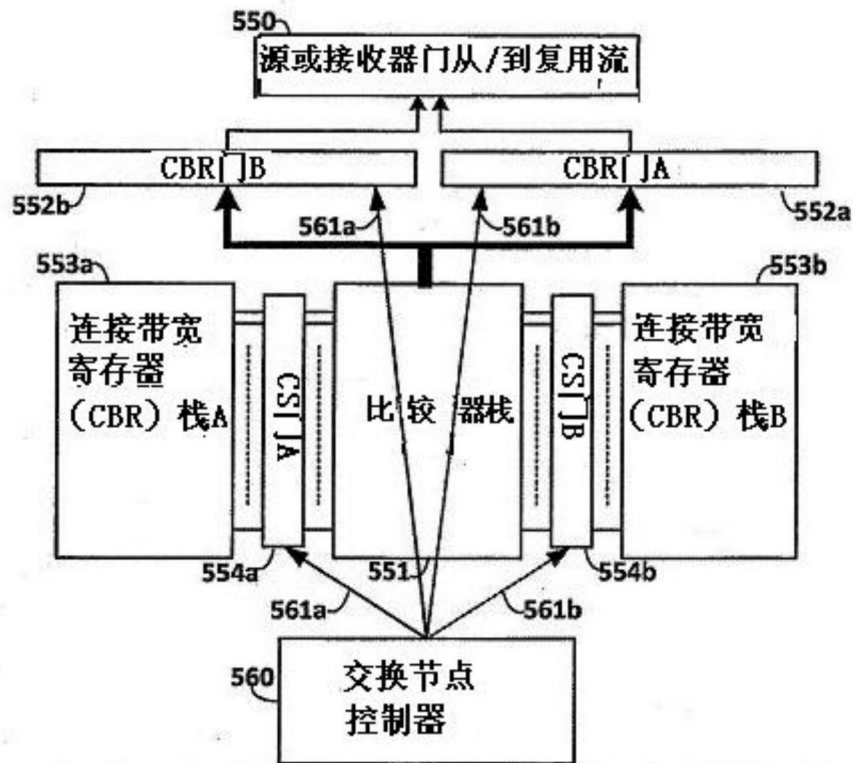
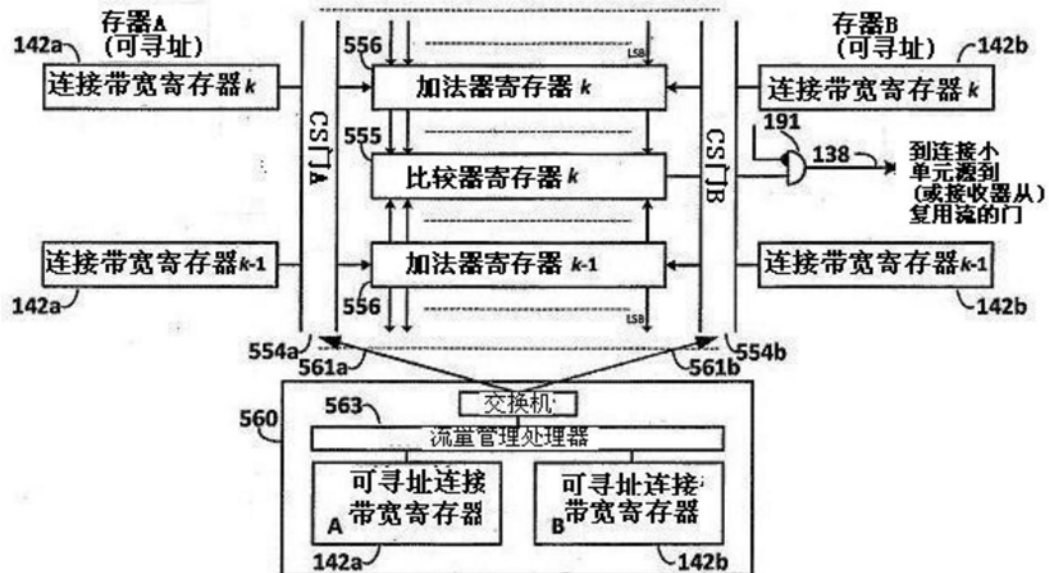


图10b



用于改变连接比较器的装置：带宽寄存器级

图11



用于改变连接域连接带宽范围的装置

图12

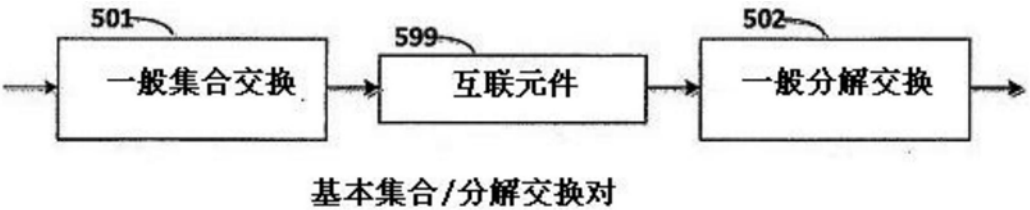


图13

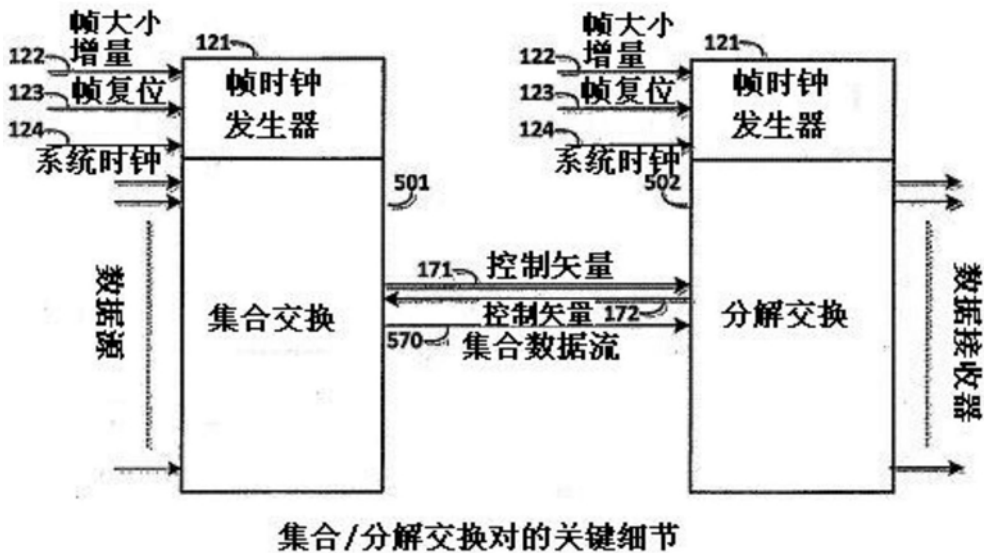
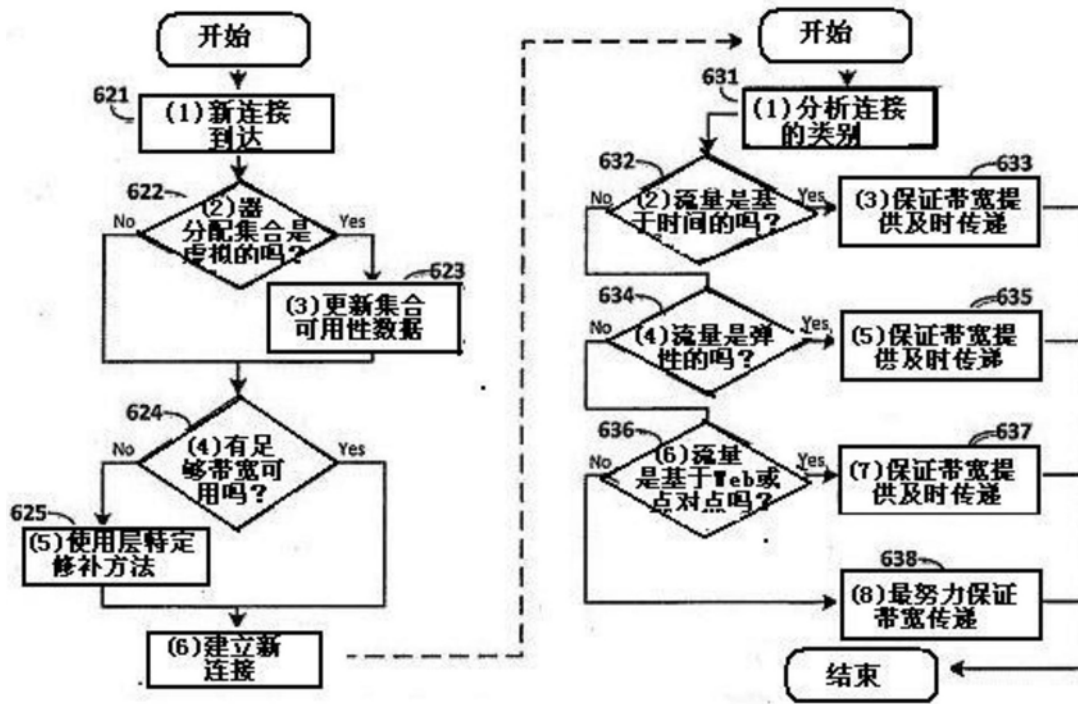


图14



定义设定新连接流程的流程图

图15

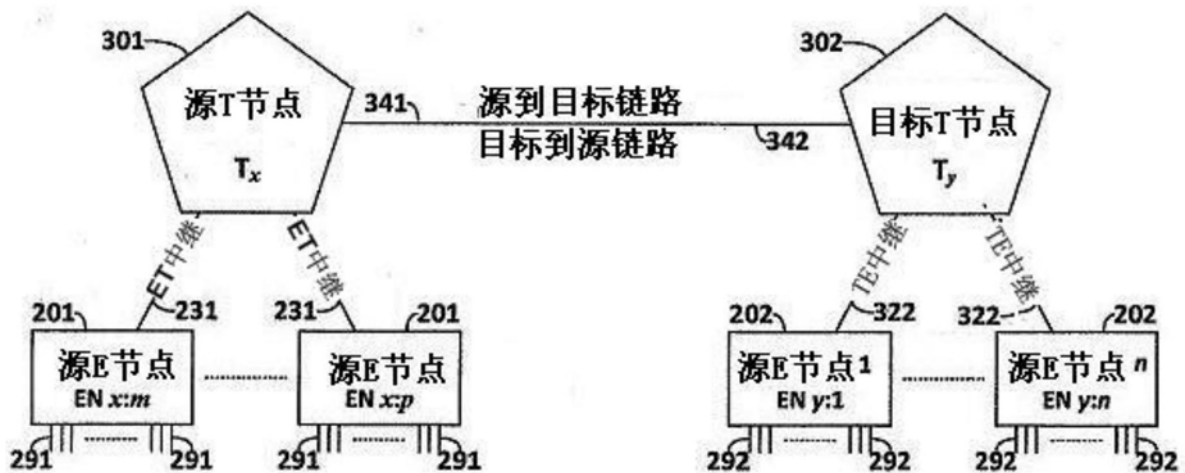


图16a

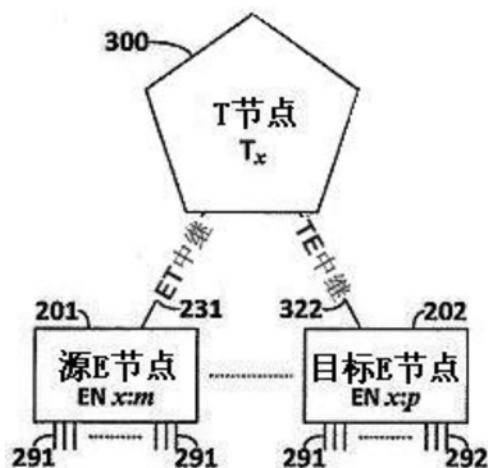
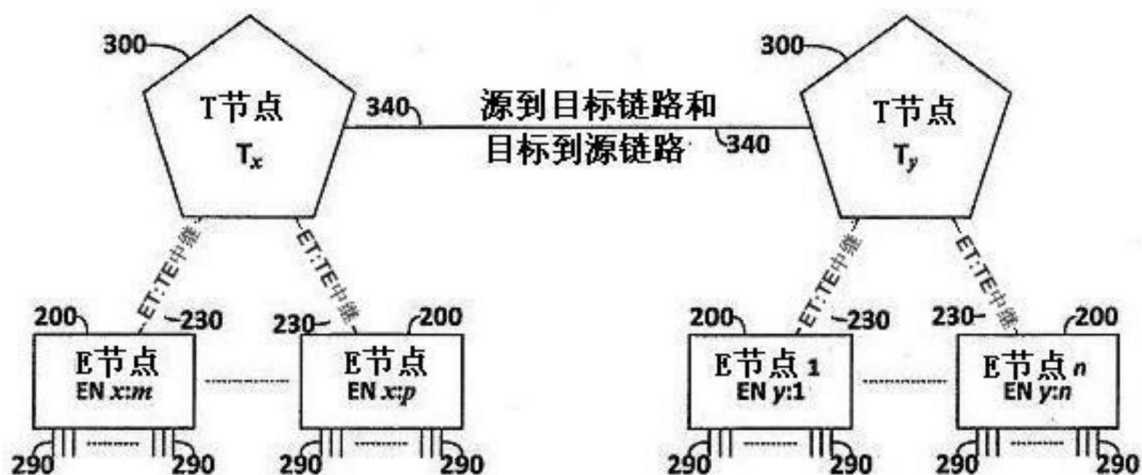


图16b



E节点和其父T节点的各种配置

图16c



连接域：源端集合方法

图17

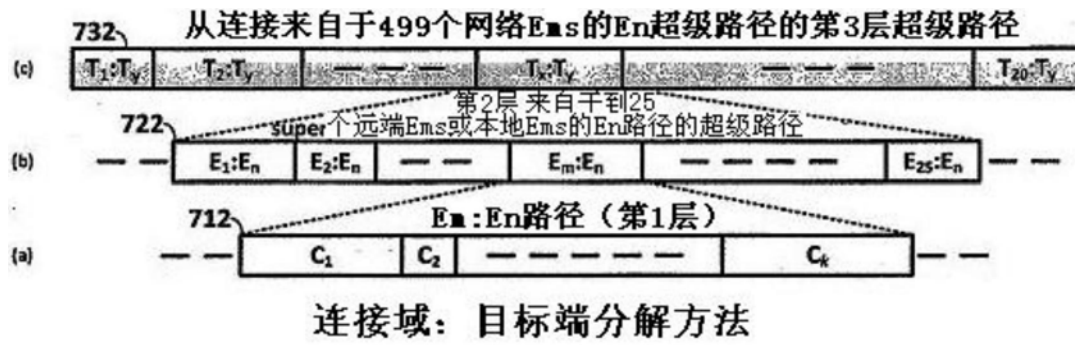


图18

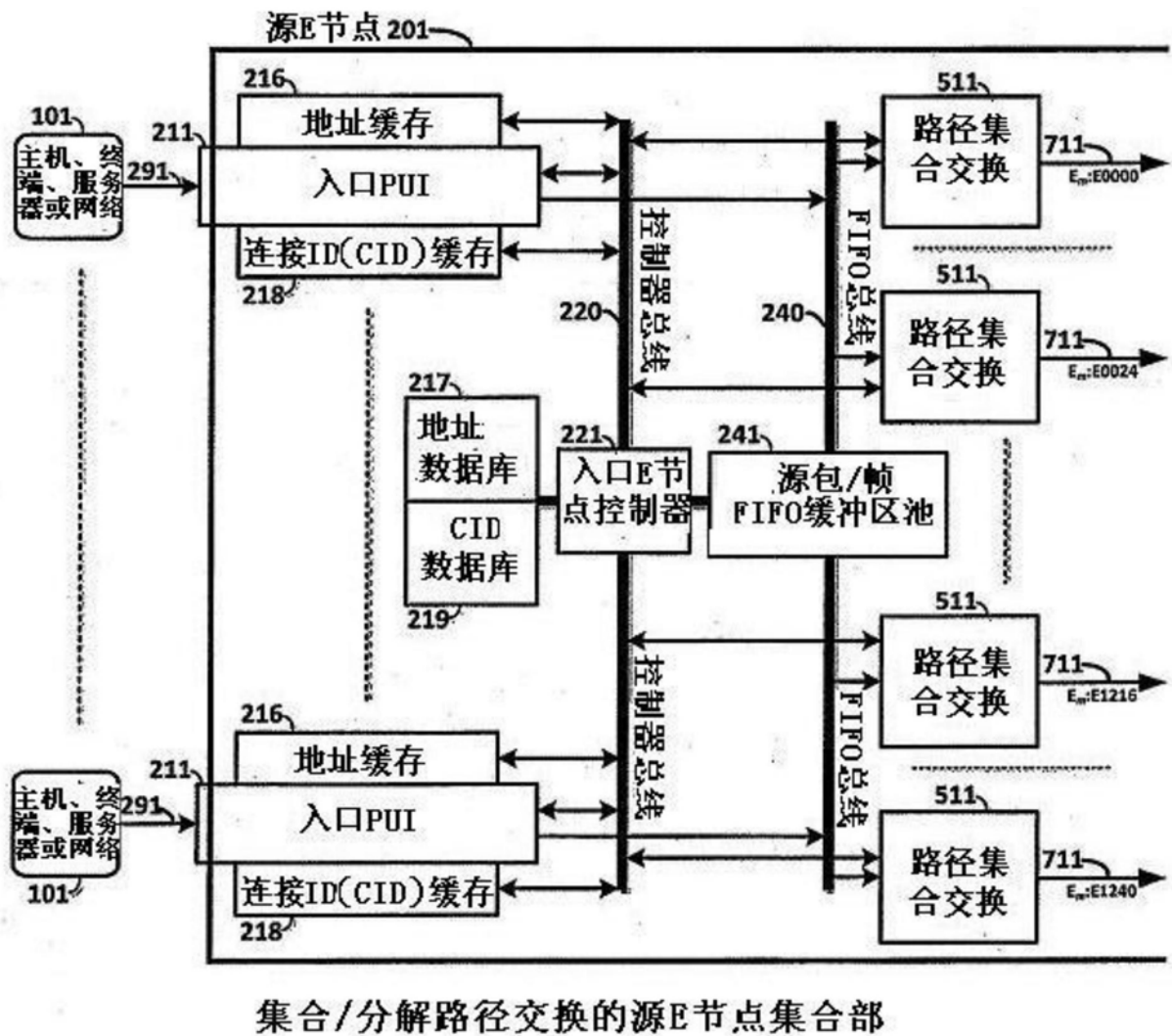


图19

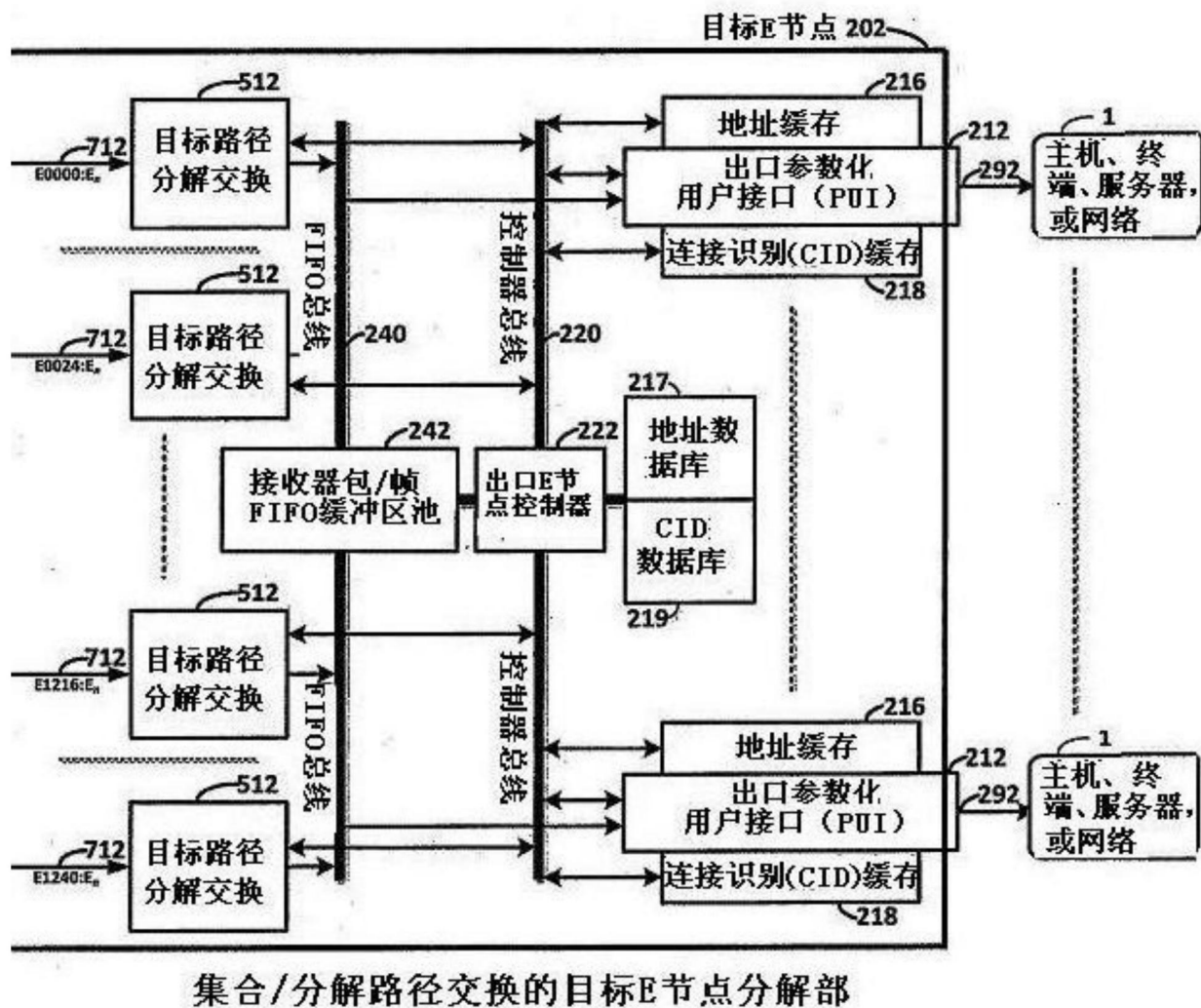
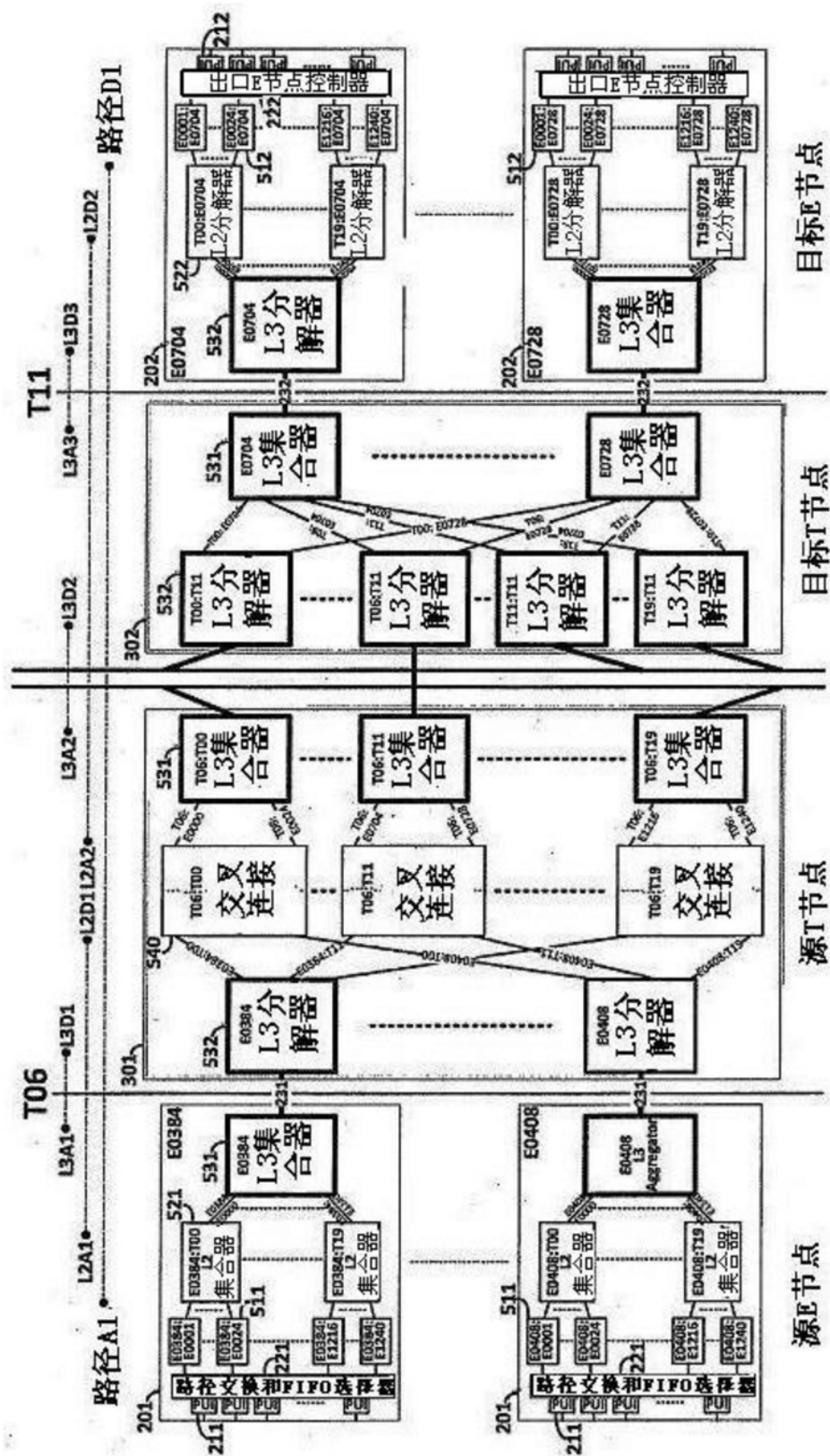


图20



具有1个路径层交换对、2个第2层对、以及3个第3层对，其将每个子源E节点连接到T节点（T11）的每个子目标E节点

图21

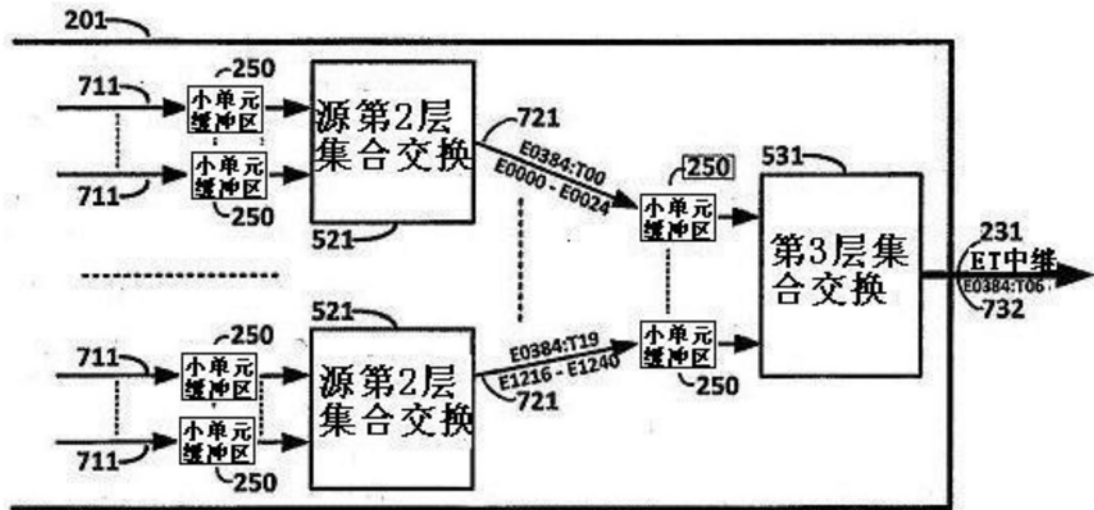


图22a

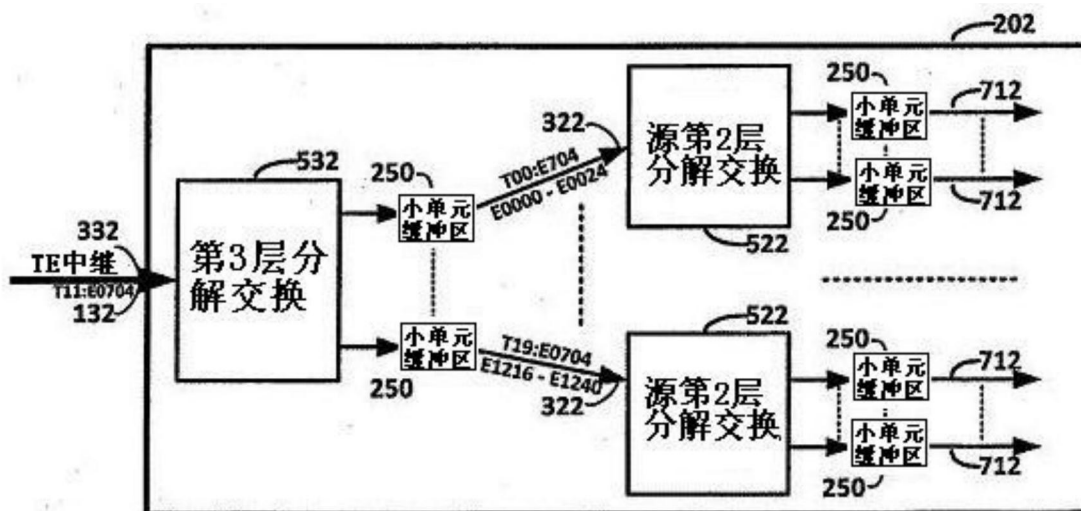
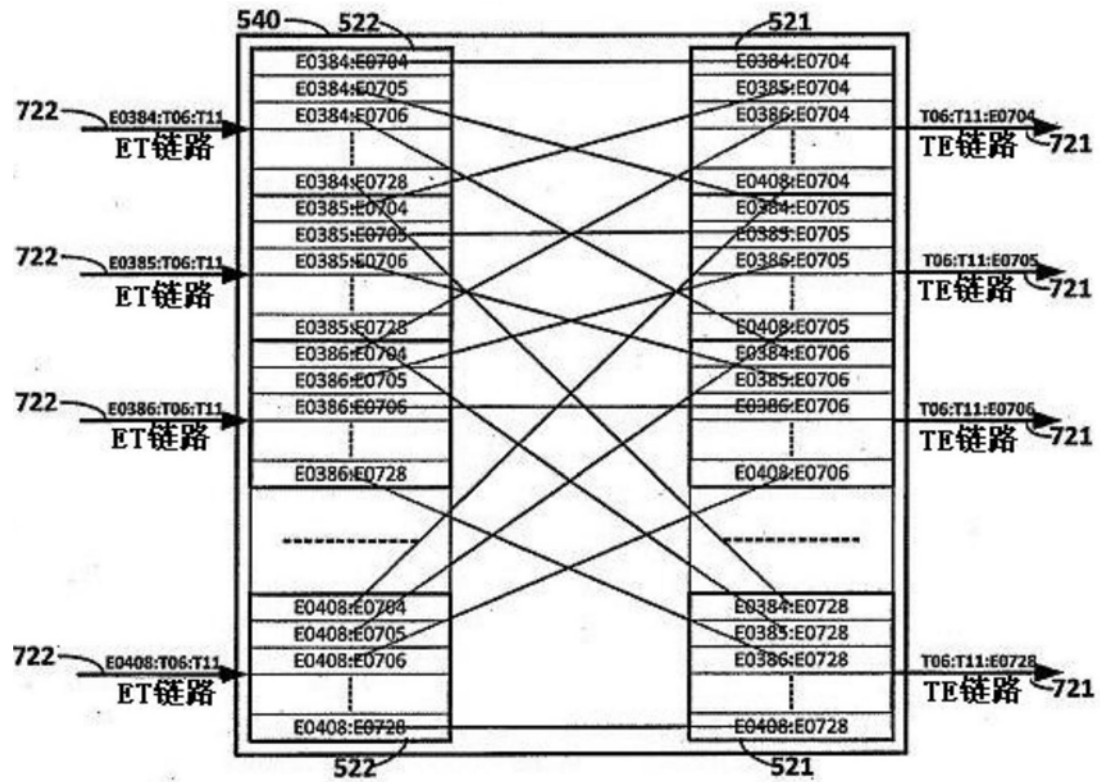
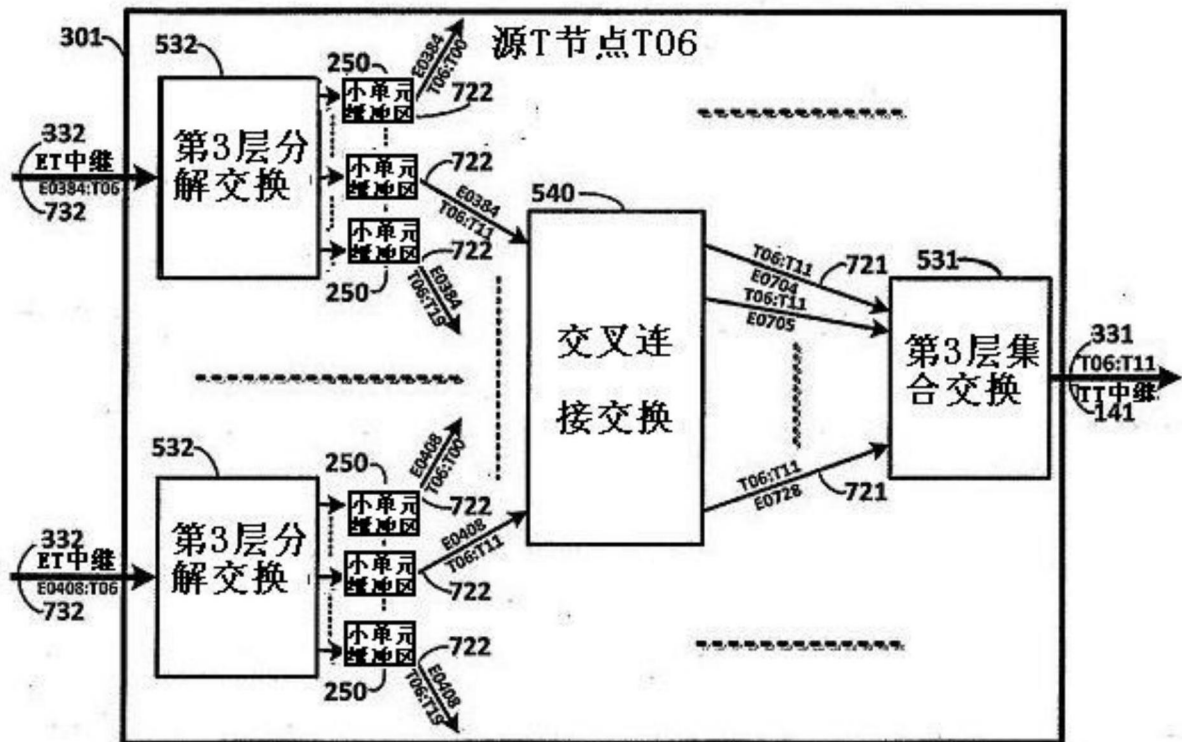


图22b



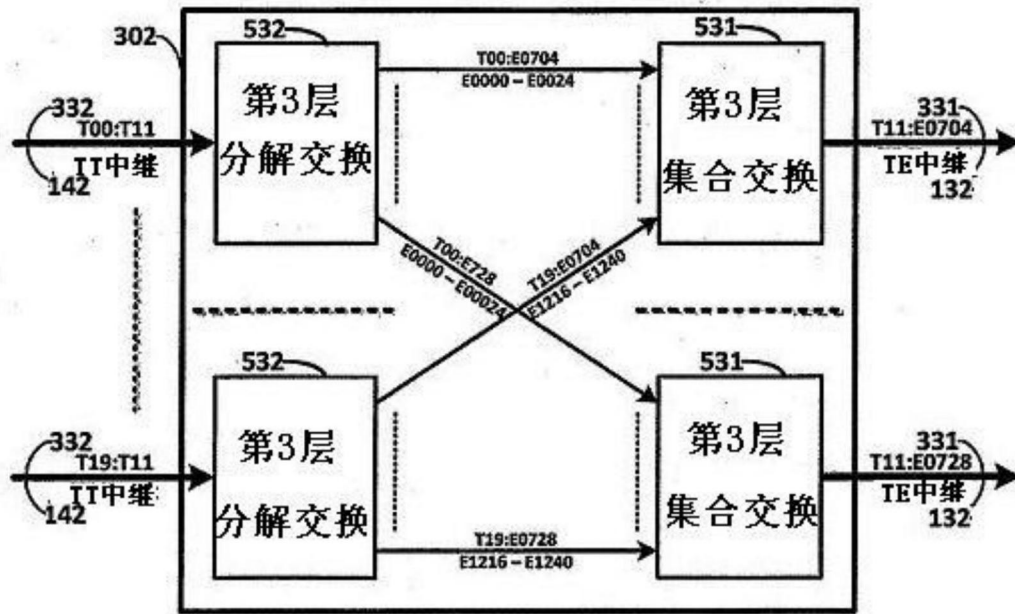
导致一个父T节点 (T06) 的每个子E节点连接到第二T节点 (T11) 的所有子E节点的交叉连接交换

图23



使用交叉连接交换发起将499个路径的25个第2层集合的每个从1对多重定向到多对1的源T节点06的示例

图24



使用交叉连接交换以完成从20个T节点到T节点11的25个子E节点的多对1映射的目标T节点11示例

图25

第3层交换的带宽和时延						Nbr时钟
L3时钟率	1.5625Gbps		时钟周期	0.64纳秒		
L3小单元大小	64	bits	L3帧段	1		周期
L3帧率	312.5	Mfps	L3子帧周期	3.2	nsec	5
L3帧周期	3.2	nsec	L3最小集合带宽	20	Gbps	5
L3 QDR	20.000	Gbps	L3最大集合带宽	100	Gbps	5
第2层交换的带宽和时延						
L2时钟率	1.562500Gbps		时钟周期	0.640纳秒		
L2小单元大小	16	bits	L2帧段	50		
L2帧率	6.25	Mfps	L2子帧周期	3.2	nsec	5
L2帧周期	160	nsec	L2最小集合带宽	5	Gbps	250
L2 QDR	100	Mbps	L2最大集合带宽	25	Gbps	5
路径层交换的带宽和时延						
路径时钟率	1.562500Gbps		时钟周期	0.640纳秒		
小单元大小	1	bit	帧段	50		
帧率	125	Kfps	子帧周期	160	nsec	250
帧周期	8	μsec	最小集合带宽	6.250	Mbps	12500
QDR	125	Kbps	最大集合带宽	1.563	Gbps	250

751 SAIN 集合层的第一个示例

图26a

第3层交换的带宽和时延						Nbr时钟
L3时钟率	1.280Gbps		时钟周期	0.781纳秒		
L3小单元大小	128	bits	L3帧段	1		周期
L3帧率	256.000	Mfps	L3子帧周期	3.906	nsec	5
L3帧周期	3.906	nsec	L3最小集合带宽	32.768	Gbps	5
L3 QDR	32.768	Gbps	L3最大集合带宽	163.840	Gbps	5
第2层交换的带宽和时延						
L2时钟率	1.280Gbps		时钟周期	0.781纳秒		
L2小单元大小	16	bits	L2帧段	32		
L2帧率	8.000	Mfps	L2子帧周期	3.906	nsec	5
L2帧周期	125.000	nsec	L2最小集合带宽	4.096	Gbps	160
L2 QDR	128.000	Mbps	L2最大集合带宽	163.840	Gbps	5
路径层交换的带宽和时延						
路径时钟率	1.280Gbps		时钟周期	0.781纳秒		
小单元大小	1	bit	帧段	64		
帧率	125	Kfps	子帧周期	125.000	nsec	160
帧周期	8	μsec	最小集合带宽	8.000	Mbps	10240
QDR	125	Kbps	最大集合带宽	1.280	Gbps	160

752 SAIN 集合层的第二个示例

图26b

第3层交换的带宽和时延						Nbr时钟
L3时钟率	4.194304Gbps		时钟周期	0.238纳秒		
L3小单元大小	8	bits	L3帧段	1		周期
L3帧率	1048.576	Mfps	L3子帧周期	0.954	nsec	4
L3帧周期	0.954	nsec	L3最小集合带宽	8.389	Gbps	4
L3 QDR	8.389	Gbps	L3最大集合带宽	33.554	Gbps	4
第2层交换的带宽和时延						
L2时钟率	4.194304Gbps		时钟周期	0.238纳秒		
L2小单元大小	1	bits	L2帧段	128		
L2帧率	8.192	Mfps	L2子帧周期	0.954	nsec	4
L2帧周期	122.070	nsec	L2最小集合带宽	1.049	Gbps	512
L2 QDR	8.192	Mbps	L2最大集合带宽	33.554	Gbps	4
路径层交换的带宽和时延						
路径时钟率	1.048576Gbps		时钟周期	0.954纳秒		
小单元大小	1	bit	帧段	128		
帧率	64	Kfps	子帧周期	122.070	nsec	128
帧周期	15.625	μsec	最小集合带宽	8.192	Mbps	16384
QDR	64	Kbps	最大集合带宽	1.049	Gbps	128

SAIN 集合层的第三个示例
753

图26c

第3层交换的带宽和时延							Nbr时钟
4.194304	L3时钟率	4.194304Gbps		时钟周期	0.238纳秒		
	L3小单元大	32	bits	L3帧段	1		周期
	L3帧率	1048.576	Mfps	L3子帧周期	0.954	nsec	4
	L3帧周期	0.954	nsec	L3最小集合带宽	33.554	Gbps	4
	L3 QDR	33.554	Gbps	L3最大集合带宽	134.218	Gbps	4
第2层交换的带宽和时延							
4.194304	L2时钟率	4.194304Gbps		时钟周期	0.238纳秒		
	L2小单元大	1	bits	L2帧段	128		
	L2帧率	8.192	Mfps	L2子帧周期	0.954	nsec	4
	L2帧周期	122.070	nsec	L2最小集合带宽	1.049	Gbps	512
	L2 QDR	8.192	Mbps	L2最大集合带宽	134.218	Gbps	4
路径层交换的带宽和时延							
1.048576	路径时钟率	1.048576Gbps		时钟周期	0.954纳秒		
	小单元大小	1	bit	帧段	128		
	帧率	64	Kfps	子帧周期	122.070	nsec	128
	帧周期	15.625	μsec	最小集合带宽	8.192	Mbps	16384
	QDR	64	Kbps	最大集合带宽	1.049	Gbps	128

754 SAIN 集合层的第四个示例

图26d

[illegible]

(a) PoT段按长度降序排列的62个小单元帧

[illegible]

(b) PoT段按长度降序排列的47个小单元帧

[illegible]

(c) PoT段按长度降序排列的32个小单元子帧

[illegible]

(d) PoT段不按长度排列的32个小单元子帧

图27