

基于 ATM 网络多播树自愈策略的综合研究^{*}

Studies of Comprehensive Solution of Multicast Tree Self-healing ATM Networks

刘宴兵 李秉智

幸云辉

(重庆邮电学院计算机系 重庆400065) (北京邮电大学 北京100080)

Abstract In this paper, we will study the self-healing technology in ATM network mainly. We firstly introduce the characteristic of ATM technology, the re-routing scheme of self-healing technology and the structure of ATM self-healing net. To satisfy different requirement of different service and adapt the development of single-node to multi-node communication, we discuss the self-healing technology for ATM multicast tree by improving the self-healing technology based on VP. It's important to ensure higher-PRI service to be restored firstly.

Keywords Self-healing, Survival, Virtual path, Multicast tree

二十一世纪是一个信息的时代,也就是说,它是一个与网络分不开的时代,因为现代化的信息传输工具就是网络。那么,怎样才能保证我们所需要的信息及时、准确地传送?当一段链路或某个节点发生故障时,信息会不会丢失?所谓自愈,是指网络在发生故障时,不需要人工干预,就能很快地恢复所涉及的业务。本文将对 ATM 网络的自愈技术进行综合讨论。

1 自愈技术的重构路由方案

1.1 自动保护切换(APS)

自动保护切换方法为每一个网络单元设置一个专用的备用单元,一旦检测到失效,立即将该网络单元上的业务切换到备用单元,然后可以对失效的单元进行修复。在 SDH/SONET 光缆保护中,备用光缆的路径往往与主用光缆的地理位置不相交,使得工作光缆被切断时能保护其上的业务。1:N/DP(Diverse Protection)线路保护系统(通常 N 的取值范围是1—14)是常用的 APS 方式,一根备用光缆用于对与其在地理位置上不相交的 N 个工作光缆进行保护;1:1/DP 保护系统可以为光缆的切断提供100%的恢复,但投资高,通常的做法是将两根路由不相交的光缆互做保护容量。自动保护切换是不基于 DCS(Digital Crossconnect System)的自愈算法。由于采用物理设备切换,速度非常快,一般能在50ms内完成,然而物理资源没有被充分利用,造价很高。

1.2 自愈环(SHR)

自愈环结构是由 SDH/SONET 的分插复用器(ADM)构成的环形拓扑结构,它利用了 ADM 的分插能力和智能性,通过 ADM 结点把两根或四根传输方向相反的光纤串接在一起,能够改善网络的生存性,减少远距离传输所需的再生器,降低网络成本。对于一个方向上的业务量,经常在相反方向上保留有相同的空闲容量,以达到任意两节点的链路失效或者一个节点的失效。

1.3 泛洪式恢复算法(Flooding)

泛洪式恢复算法是一种分布式控制算法,主要用于连接度高、业务量集中的网状网中,尤其是长途网中,网上的每个节点通过广播传递信息,对网络拓扑只有有限的了解。

2 两种基本的 ATM 网络自愈技术

2.1 备份 VP 恢复算法

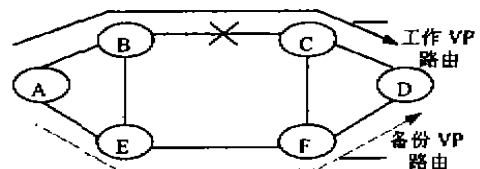


图1 备份 VP(Backup VP)恢复方案

ATM 网络的自愈以 VP 为基础,目前受到广泛重视的备份 VP 自愈算法的示意图如图1所示。在 ATM 网络中,对于某一工作 VP 预先分配一备份 VP,并存

^{*} 本文由信息产业部“九五”重大科技发展项目资助(编号:98048),青年教师科技基金资助项目。刘宴兵 讲师,硕士生,主要从事计算机网络性能研究,李秉智 教授,博士生导师,幸云辉 教授。

储于数据库之中。当传输链路中断时,VP 的两个终端点试图恢复通道,其中之一成为发送者,另外一个成为选择者,图1所示的网络发生故障时,发送者(业务信号的接收端点)检测到 VP 中断,沿着备份 VP 发送恢复消息,下一节点接收到恢复消息,同时获取适当的带宽,再把恢复消息传送到备份 VP 的下一节点。当选择者(业务信号的发送端点)接收到恢复消息后,把业务从失效的 VP 转接到备份 VP 上,从而对这一条 VP 完成了故障恢复。根据 ATM 的特点,VP 的带宽是动态分配的,对于备份 VP,其初始带宽可以置为0,所以,和基于物理通道的自愈方法相比,备份 VP 自愈算法提高了对网络资源的利用率。

2.2 泛洪式恢复算法

泛洪式恢复算法从 STM 的泛洪式恢复算法发展而来,是一种基于 ATM 的分布式控制算法,每个节点对网络拓扑只有有限的了解。当一条链路或节点失效时,失效链路的两个端点后受影响的 VP 的源、宿节点检测到失效,就企图去恢复该通道。两节点向与其相邻的所有有足够空闲容量的链路发送恢复消息。每个中间节点都继续向其邻接节点广播恢复消息,当源和宿发出的恢复消息在某个中间节点相遇时,就找到了一条变更通道。这个中间节点就发回一个响应消息给源、宿节点实现恢复切换。

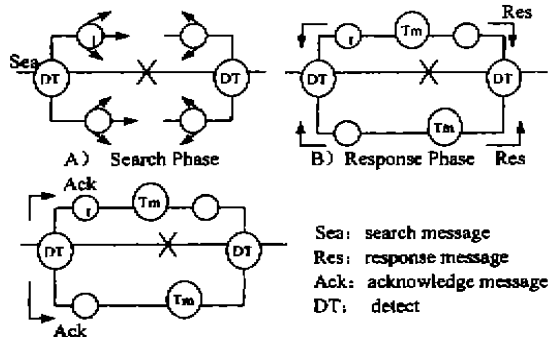


图2 泛洪式(Flooding)恢复算法示意图

目前比较流行的分布式泛洪式恢复算法是基于 LINK 的双向搜索算法,如图2所示,该算法共分三个阶段:当失效 VP 的两个端点检测到失效时,两个端点分别成为 Sender 节点(图2中的 DT 节点),且分别向相连的所有链路广播搜索(search)消息,然后两个 DT 节点将自己设成 Chooser 节点,每个中间节点在收到 search 消息后,试图再与其相连的所有有空闲容量的链路上保留空闲容量,并沿该链路继续广播 search 消息;当某个中间节点(Tm)收到了发自两个 DT 节点的 search 消息后,就在该节点发生了碰撞,Tm 节点将两个 search 消息合成为一个 response 消息,并分别向两

个 DT 节点(此时为 Chooser)沿所找到的恢复路由发送 response 消息;DT 节点收到 response 消息后,进入 acknowledge 阶段,将根据变更路由中所允许的空闲容量选择一组失效 VP 予以恢复,直到所有失效 VP 都被恢复为止。

3 ATM 多播树的自愈技术方案

3.1 AMT 自愈技术的提出

上面讨论的主要是点到点通信的 ATM 网络,这些技术已日趋成熟。实际上,随着多媒体、分布式计算、协同操作、群组通信等技术的日趋完善,视频会议、分布式数据库、网络游戏、VOD、远程教学等多播业务的应用变得日趋广泛。ATM 网络有许多特点适合多播通信,通过研究表明:有效的途径就是构造 ATM 多播树(AMT)。AMT 自愈算法也引起了普遍重视。

3.2 基于“链路”的 AMT 自愈机制的缺陷

将 ATM 网络中采用的差错恢复机制进行推广可形成多种 AMT 恢复机制(即自愈机制),其中应用较为广泛的是传统的基于链路的 AMT 自愈策略。采用这种机制,当 AMT 失效时,与失败链路直接相邻的下连接点成为查询消息发送方(Sender 方,简称‘S’),与失败链路直接相邻的上连接点成为选择重发方(Chooser 方,简称‘C’方),通过点到点的通信方式完成失败链路的恢复,其原理如图3所示,在 Sender-Chooser 对中仅存在1个 Chooser 方。

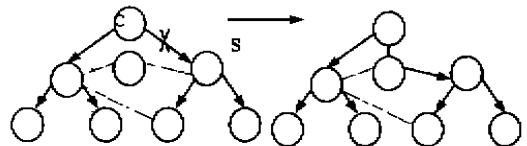


图3 基于链路的 AMT 恢复机制原理图

采用动态恢复机制,传送查询消息的同时要在其所经链路上预留恢复失败业务所需的带宽资源,一旦在某处得不到差错恢复所需的带宽资源,该路由将不能成为一条满足条件的可替代路由。从图可以看出:在基于链路的 AMT 恢复机制中,Sender 方与 Chooser 方形成了一对一的关系,也就是说,当一棵 AMT 失效时,在 Sender-Chooser(s) 对中仅存在一个 Chooser 方,这样在 Sender 与 Chooser(s) 间用于差错恢复的可选路由数目较少,在带宽资源竞争(如同时存在多棵 AMT)较为激烈的情况下,恢复率(主要指恢复的可靠度,其定义为:得到恢复的 AMT 数目与因链路或节点失败而需要恢复的 AMT 总数之比)将大打折扣;另一方面,在基于链路的恢复机制中,无论 AMT 的结构如何,恢复任务均由固定的 Chooser 方来完成,而在选定

的可替代路由上可能存在已经成功地接收到了 Sender 方未接收到的数据信息的中间节点,如果用与 Sender 方距离更近的中间节点代替 Chooser 方完成差错恢复,恢复时延将降低。上述两点正是基于链路恢复机制的主要缺陷,而造成这些缺陷的关键因素在于:基于链路的 AMT 恢复机制根本不考虑多播树的结构,差错恢复后,AMT 的结构根本不发生变化。

3.3 基于“树”结构的 AMT 自愈策略

实现基于“树”的混合优先级自愈机制的关键是如何选择多个合适的节点作为选择方。当链路失败时,原有的 AMT 被分成两棵子树,见图4。

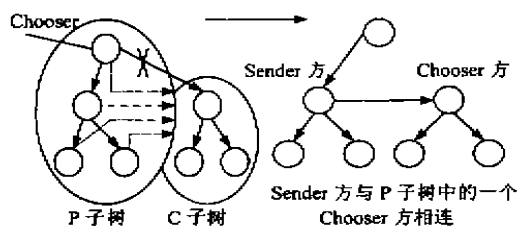


图4 P子树和C子树划分示意图

从图4可以看出,P子树以原AMT的根节点作为自己的根节点。如果将C子树的根节点与P子树中的任何节点相连,实现两棵子树的融合,重新构建AMT,利用新构建的AMT即可恢复失败业务。从这个意义上来说,P子树中的任何节点均可充当Chooser方角色。

当链路失败时,与失败链路直接相邻的下连接点成为发送方,它将立即向以其为根节点的下连子树(即C子树)上的所有节点发送AIS(Alarm Indication Signal)信号。所有接收到AIS信号的节点均被触发为失败模式。因为一个节点可能属于多棵AMT,所以它可能接收到来自不同AMT的多个AIS信号。处于失败模式的节点具有两个基本特点:(1)该节点一定位于失败AMT的C子树上,因此不能充当Chooser方角色(即只有P子树上的节点才有可能成为Chooser方);(2)不再发送查询消息,这就解决了无效NACK(否定确认消息)的重发问题。与此同时,Sender方将创建一查询消息并以广播的方式向其所有的输出链路上发送。当一个节点接收到查询消息时,首先判断自己是否是该消息正在搜寻的Chooser方,判断规则如下:如果该节点的GTID(全局“树”标识,每棵AMT有唯一的GTID)与查询消息中所携带的GTID相同并处于非失败模式,则该节点就是查询消息正在搜寻的Chooser方之一。一旦接收到查询消息,Chooser方立即创建一“连接建立”消息并将该消息返送回Sender以建立一条可替代路由。与基于链路的恢复机制不同,在基于

“树”的恢复机制中,由于存在多个Chooser方,所以Sender方可能接收到不同Chooser方发来的多个“连接建立”消息。Sender方按照FCFS(先来先服务)原则只接纳最先到达的消息,并通过返送“丢弃”消息的方式将Sender与Chooser方(返回的“连接建立”消息被接纳的Chooser方除外)间预留的带宽资源释放掉。采用FCFS原则也是为了选出代价最小(时延最小或链路负载最轻等)的路由充当可替代路由以恢复失败的业务,从而提高了恢复机制的整体性能。当Sender方接收到“连接建立”消息时,立即返送“确认”消息至Chooser方,同时在C子树中广播“连接建立”消息以改变节点的GTID,C子树中的所有节点归入新的AMT,具有新的GTID。利用新建AMT即可完成失败业务的恢复。

为方便起见,以上过程都是基于链路失败这一情况进行讨论的。其实,链路失败和节点失败都会引起AMT中多播业务的失败。如果一个节点具有2条或多条(假设有N条)直接相邻的下连支路,当该节点失败时,就会有多个(N个)节点检测到失败并成为具有相同GTID的Sender方,原AMT被分成一棵P子树和多棵(N棵)C子树。由于以广播方式发送AIS信号存在一定的时延,可能会出现某棵C子树上的节点在接收到该子树根节点发送来的AIS信号之前已经被其它C子树的根节点选为Chooser方(即被误认为该节点位于P子树中,这种Chooser方可以称为“虚”Chooser方),当“虚”Chooser方接收到AIS信号后,立即被触发为失败模式,从而失去作为Chooser方的全部功能。也就是说原先被设定为Chooser方的节点在接收到被拖延的AIS信号后失去作为Chooser方的全部功能,同时通过发送丢弃消息释放先前在Sender方与该“虚”Chooser方之间预留的所有带宽资源。该节点也返回消息通知相应的Sender方,Sender方需要重新开始其恢复过程。由此可见,节点失败后的恢复过程要比链路失败后的恢复过程复杂得多。

4 两种恢复方案的评价

基于“树”结构、具有多优先级的AMT恢复机制在性能上明显优于基于链路的恢复机制。这主要是因为新机制充分考虑到AMT的结构并引进了具有优先级的“等待-重启”策略,利用多对Sender-Chooser组协同工作以完成失败业务的恢复。在前人所做的仿真实验中,当链路失败时,基于“树”的恢复机制的恢复率最大可达到99.9%,而基于链路的恢复机制的恢复率最大仅达到94.2%;此外,采用基于“树”的恢复机制所需的恢复时延将显著缩短,采用基于“树”的恢复机制,恢

(下转第59页)

且对于巨型的复杂的应用来说该层服务器分布式技术合适且现实。

3 框架性能的分析

显然该框架组织对只有少量信息的 Web 服务或只需要少量信息的网上应用,没有什么益处,反而由于存取数据库,构建模型资源对象而大大降低反应时间。但是对于需要复杂多媒体信息的一些网上商场,电子商务,网上教学确能很好地提高效率,主要是由于对象的组织使人们在各个网页之间链接来、链接去的查找时间减少,基本上找到一个资源对象就可以找到一批相关的资源,从而提高工作效率。我们可知:

$$\text{总体时间 } T = t_{TP} + t_{BO} + t_{DB}$$

t_{TP} 网络传输时间 $t_{TP} = V$, 总体数据量 V 网络传输率

显然 t_{TP} 与网络硬件操作系统的关系密切,我们能做的是减少重复传送,而其它一般没有办法。 t_{BO} 对象构建和析构时间 t_{BO} 数据库查询等操作时间。

$$t_{BO} = \begin{cases} t_{模板} + t_{建立对象} + t_{封装HTML} \\ t_{模板} + t_{析构对象} + t_{解封装HTML} \end{cases}$$

$$t_{DB} = t_{建物理连接} + t_{查询} + t_{输出}$$

由于缓冲区的建立,因此

$$T = t_{TP} + t_{建缓冲区} + t_{传输缓冲区}$$

由于缓冲区是牺牲相应内存和硬盘空间,故不会太大

$$t_{建缓冲区} + t_{传输} \ll t_{BO} + t_{DB}$$

所以如果能提高缓冲池命中率,效率必然会大大提高。由于在某些情况下对象请求共用一数据库物理连接,故 t_{DB} 又可为

$$t_{DB} = t_{建物理连接} / n + t_{查询} + t_{输出}$$

一般情况下建立物理连接起动一个新进程是比较耗时的,复用技术可使物理连接时间由 n 个请求平摊,但我们又必须注意在某一微观时刻只能有一个请求占有物

理通道,故增大 n 必然使 t 轮询时间增大(即各个请求平均等待时间)所以必须对 n 取一个合理的值范围。

必须注意到以上都是在没有进入临界状态的情况,因为我们知道任何系统的资源总是有限的,随着客户连入增大,进程不断开设,系统可用资源将减少,到最后进入临界状态,即系统已无可利用资源,当有一个新客户到达时,系统为了替其建立进程和分配资源,必然导出其它进程到硬盘,从而引起系统抖动,不断导出,性能急剧下降,所以必须通过实验测出系统临界状态,确定用户上限。

结论 本文针对大量无组织零散的分布于网上各处的信息数据,提出一种在逻辑上有组织的统一的多媒体数据对象,它有效地帮助用户检索到所需要的信息,适应复杂和大规模应用,有效地保证信息数据的一致性,同时也有利于网站的有效管理。另外,本文提出了分析该框架性能的基本参数模型。该框架结构对 Web 资源进行了有效的组织,建立了基于面向对象观点的资源对象模型,比较适用于复杂的多媒体信息数字网上应用。

参考文献

- 1 Konopnicki D, Shmueli O. W3QS: a query system for the World Wide Web. In: Proc the 21st Very Large Data Base. Zurich Switzerland, 1995. 54~65
- 2 Lakshmana L V S, Sadri F, Subramanian L N. A declarative language for querying and restructuring the Web. In: Proc of RIDE'96. New Orleans, 1996
- 3 Mendelzon A O, Mihaila G A, Milo T. Querying the World Wide Web. In: Proc of PDIS'96. 1996. 80~91
- 4 Fiebig T, Weiss J, Moerkotte G. RAW: a relational algebra for the Web. In: Proc of Workshop on Management of Semistructured Data. Tucson, Arizona, 1997
- 5 陈彦, 徐宏炳, 王能斌. 基于标记图的 Web 数据模型. 计算机学报, 1999, 3: 306~312

(上接第24页)

复率达到90%所需的平均时延大约为240ms,而采用基于链路的恢复机制时,恢复率达到90%所需的时延大约为440ms。这主要是因为基于“树”的恢复机制可能利用一条更短的路径完成恢复任务。当节点失败时,无论采用何种机制,恢复性能都将大幅度降低,但基于“树”的恢复机制的性能还是明显优于基于链路的恢复机制。

结论 BVP 算法和泛洪式恢复算法满足了不同业务和用户对网络生存性的不同要求,在实际网络中应用很广泛,是 ATM 网络自愈技术的基础,很多以后的改进算法都是在此基础上进行的。当然这两种传统的自愈算法并不是十全十美的,本文所讨论的 ATM 多播树就是一个例子,当采用这两种算法时,恢复率并

不是很高,且时延大,我们针对多播树的结构特点,对传统的自愈算法进行改进,以适应多播树中失效信息的恢复,改进后的算法明显地提高了 ATM 多播树的性能。

参考文献

- 1 Kawamura, Sato K, Tokizawa I. Self-rerouting ATM networks base on virtual path concept[J]. IEEE J SAC, 1994, 12(1): 120~127
- 2 Anderson J. Fast Restoration of ATM Network[J]. IEEE J SAC, 1994, 12(1): 128~138
- 3 马丁·德·普瑞克(程时端译). 异步传递方式. 邮电出版社, 1995
- 4 Tanenbaum A S. Computer Networks 清华大学出版社(影印版), 1996
- 5 刘夏兵, 等. 基于 ATM 构造现有网络技术. 数字通信, 2000(1): 42~44