

文章编号:1001-9081(2006)07-1536-03

基于服务元网络体系结构的可靠数据传输机制研究

王 焱¹, 郑俊辉², 曾家智¹

(1. 电子科技大学 计算机科学与工程学院, 四川 成都 610054;

2. 西南民族大学 计算机科学与技术学院, 四川 成都 610041)

(w_y0409@sohu.com)

摘 要:服务元网络体系结构采用端到端的虚电路实现数据的传输。针对这种数据传输方式,提出用改进的滑动窗口协议保证服务元网络体系结构的可靠数据传输,并使用着色 Petri 网(CPN)建模工具验证了该机制的可靠性和完备性。

关键词:服务元;数据传输;虚电路;滑动窗口;着色 Petri 网

中图分类号: TP393.02 **文献标识码:** A

Research on the reliable data transmission mechanism for SUNA

WANG Yan¹, ZHENG Jun-hui², ZENG Jia-zhi¹

(1. School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu Sichuan 610054, China;

2. School of Computer Science and Technology, Southwest University for Nationalities, Chengdu Sichuan 610041, China)

Abstract: Service Unit based Network Architecture(SUNA) uses end-to-end Virtual Circuit (VC) structure to transmit data. In order to guarantee reliable data transmission for SUNA, an improved sliding window protocol was proposed. Colored Petri Net(CPN) was used to verify the dependability and maturity of this mechanism.

Key words: service unit; data transmission; virtual circuit; sliding window; colored Petri net(CPN)

0 引言

目前实用的网络体系结构都是层次结构,TCP/IP 是目前互联网所使用的主流技术体系。由于最初的 TCP/IP 协议栈是为了窄带文本数据而开发的,随着全球互联网的蓬勃发展,人们对网络的利用和依赖的增加,各种新的网络服务不断涌现,从而对网络的性能提出了更高的要求,TCP/IP 层次网络体系所带来的局限性越来越突出。

从 20 世纪 90 年代开始,人们采用不同的方法来提高网络的性能^[1~4]。近年来,出现了非层次的计算机网络体系结构,这就是基于角色的网络体系结构^[5]和服务元计算机网络体系结构(Service Unit based Network Architecture,SUNA)^[6]。

由于 SUNA 采用端到端的虚电路方式传输数据,这种数据传输方式不同于 TCP/IP 协议的数据报传输方式,因此本文提出了用改进的滑动窗口协议保证服务元网络体系结构的可靠数据传输。

CPN^[7]从 Petri 网演化而来,在系统建模、计算机软件技术、自动控制系统、网络通信协议工程等方面得到广泛应用^[8]。本文给出了 SUNA 的可靠数据传输机制,使用 CPN 建模工具对该机制进行了建模、仿真与分析验证。分析结果表明,SUNA 的可靠数据传输机制是正确、可行并且是完备的。

1 服务元网络体系结构(SUNA)

SUNA 是一种新型的无层次网络体系结构。该结构中不再划分层次,其网络功能部件是服务元。各个服务元完成相对独立的网络功能,相互之间并不传递服务。不同服务元可以协调配合

实现各种不同的网络功能,为应用提供各种不同的服务。

SUNA 是服务元及其相互之间交互使用的规则的集合。服务元是能够提供服务而又隐藏内部细节的最小实体。服务元提供服务是通过服务数据单元 SDU 完成的。服务元是 SDU 的发送者、接收者、转发者或变换者。

由于每个服务元都完成一项基本网络功能,因此可以非常方便地对服务元系统进行扩展和定制;而服务元之间并不传递服务,这就确保了网络功能不会出现冗余,同时服务元之间的接口和交互非常简单。由服务元组合形成的网络系统具有逻辑简单、实现方便、高效无冗余、扩展性好等优点,是一种能够满足现代应用向网络提出的提高服务质量、多媒体信息宽带传输和保障安全三个方面的新型网络体系结构。

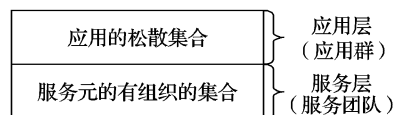


图1 SUNA 的节点模型

SUNA 的节点模型如图 1 所示。节点模型分为两部分:应用层和服务层。应用层只接受服务,服务层只提供服务。由于它们都不是典型的层,所以我们分别称之为应用群和服务团队。应用群包括应用基础(网络管理和域名解析)、典型应用(WWW,E-mail 和 FTP 等)和一般应用;应用群包含了所有的应用,而不只是共性的应用;应用群是各种应用的松散集合。服务团队是服务元的有组织的集合,它除了向本节点应用层提供服务外,还能和其他节点服务元合作向整个网络系统提供服务或向某一节点提供服务。

收稿日期:2006-01-02;修订日期:2006-03-07 基金项目:国家自然科学基金资助项目(69871005)

作者简介:王焱(1977-),女,黑龙江佳木斯人,博士研究生,主要研究方向:新型计算机网络体系结构; 郑俊辉(1975-),男,河南漯河人,讲师,主要研究方向:计算机网络与通信; 曾家智(1939-),男,四川成都人,教授,博士生导师,主要研究方向:计算机网络与通信。

2 SUNA 的数据传输方式

分组交换网络有两种数据传输方式:数据报方式和虚电路方式。TCP/IP 协议采用数据报方式传输数据。在数据报方式中,节点间不需要建立从源主机到目的主机的固定连接。源主机所发送的每一个分组都独立地选择一条传输路径。每个分组在通信子网中可以通过不同传输路径,从源主机到达目的主机。由于传输路径不固定,所以数据报传输方式不能保证源主机与目的主机之间的分组按顺序到达。

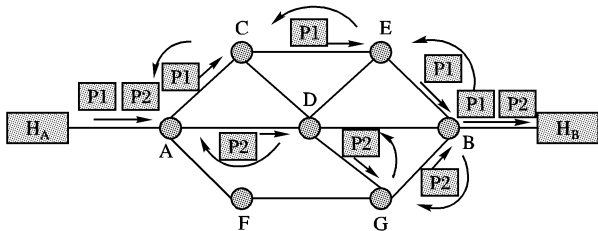


图2 数据报方式的数据交换过程

与TCP/IP协议不同的是,SUNA采用端到端的虚电路方式传输数据。虚电路就是指从源节点到目标节点之间由软件按网络地址建立起来的通道。而端到端的虚电路意味着,对于具有相同的源、目的节点地址而服务类型不同的数据,将构建不同的虚电路来传送。采用虚电路方式在分组发送前,需要在发送方与接收方之间建立一条逻辑通路。每个数据包除了包含数据之外还包含一个虚电路标识符。在预先建好的路径上的每个节点都知道把这些数据包引导到哪里去,不再需要路由选择判定。由于传输路径固定,所以虚电路方式可以保证源主机与目的主机之间的数据包按顺序到达。

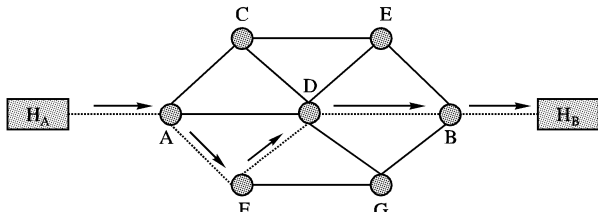


图3 虚电路方式的数据交换过程

3 SUNA 的可靠数据传输机制

3.1 传统的滑动窗口协议存在的问题

TCP/IP协议采用滑动窗口保证数据的可靠传输。如果SUNA的数据传输直接使用传统的TCP/IP滑动窗口机制,则存在着网络利用率不高的问题。这是因为在SUNA的数据传输过程中,只要数据包不丢失,就不会出现数据包错序,因此接收方只要收到错序的包,就可以判断出有数据包丢失了。若直接使用传统的TCP/IP滑动窗口机制,那么接收方即使知道数据丢失了,也不能通知发送方重传,而必须等到发送方的重传定时器溢出后才重传丢失的帧。

下面具体说明存在的问题,图4是一个3位滑动窗口的接收窗口。

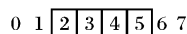


图4 一个3位接收窗口

该接收窗口下界为2,上界为5。此时接收方希望收到2#包,却收到了3#包,接着又收到了4#包、5#包,因为收到的这3个包都在接收窗口内,所以被接收方接收了。接收方又陆续收到了6#包、7#包,由于2#包一直没有收到,所以接收窗口不能向前滑动,而刚收到的6#包和7#包不在接收窗口内,因此把它们丢弃。当发送方超时定时器溢出后,发送方才重传

2#包,收到2#包后,接收窗口向前滚动,可以继续接收6#包,7#包,……

从上面的例子可以看到,如果直接采用传统的TCP/IP的滑动窗口机制,即使接收方知道2#已经丢失,也不能通知发送方重传,而必须等到发送方的重传定时器溢出,导致接收窗口迟迟不能向前滚动,因此只好丢弃窗口外6#包和7#包。这样不但降低了网络吞吐量,而且增加了数据包的传输时延。

3.2 改进的滑动窗口协议

因为SUNA是一种无层次的网络体系结构,不存在数据链路层,所以只需使用端到端的滑动窗口机制就可实现可靠数据传输。发送窗口和接收窗口设置在虚电路的两端,分别用于控制发送方发包和接收方收包的节奏。

改进的滑动窗口协议与传统的滑动窗口协议的主要不同在于:接收方可以判断出是否有包丢失,一旦有包丢失,则立即通知发送方重传,而不需要等到发送方重传定时器溢出才重传。接收方是根据数据包序号来判断数据包是否错序的,为了避免重传数据包的干扰,需进行以下改进:在发送方为重传数据包做标记;在接收方设置一个变量 $CURRENT$,用来表示最近收到的非重传包的包号。

首先看接收方的实现机制。假设滑动窗口的大小为8,在传输过程的某一时接收方收到包号为 r 的数据包,当该数据包为非重传数据包,而且包号 r 在接收窗口内时,接收方的操作与传统的滑动窗口协议是不同的。针对包号 r 的情况分别进行以下操作:

$$1) r = (CURRENT + 1) \bmod 8$$

这种情况表明,接收方收到的数据包没有错序,因此没有数据包丢失。修改 $CURRENT$,令 $CURRENT = r$,然后向发送方发回对 r #数据包的确认,并向前滑动接收窗口。

$$2) r \neq (CURRENT + 1) \bmod 8$$

这种情况表明,接收方收到了错序的数据包,因此有数据包丢失。需要进行以下操作:

① 修改 $CURRENT$,令 $CURRENT = r$ 。

② 计算需要重传的数据包个数 $count$ 。如果 $r < CURRENT$,那么 $count = r + 7 - CURRENT$;否则, $count = r - CURRENT - 1$ 。

③ 向发送方发送重传请求。重传的包号分别为: $(CURRENT - 1) \bmod 8$, $(CURRENT - 2) \bmod 8$, ..., $(CURRENT - count) \bmod 8$ 。

发送方只需进行如下改进:当发送方收到接收方的重传 n #包请求时,首先为重传的 n #包做标记,然后重传该包。而收到其他确认包时的操作与传统的滑动窗口协议一致。

4 SUNA 可靠数据传输机制 CPN 模型

4.1 CPN 模型的假设条件

在本模型中,主要考虑滑动窗口的状态、行为以及对滑动窗口的管理,而对于数据包的长度和数据包头的格式不需要进行详细讨论,因为它们不影响验证整个数据传输机制的正确性和可行性。为了简化模型,我们进行如下假设:

1) 滑动窗口协议不仅用于可靠数据传输,同时也用于拥塞控制。本模型只对与可靠数据传输有关的操作及控制行为进行建模研究。

2) 假设发送窗口与接收窗口均为2位滑动窗口,即发送窗口与接收窗口的容量均为4个数据包。

3) 本模型不讨论端系统中间路由器的拓扑结构,因为路由器的结构不影响发送窗口和接收窗口的行为。将端系统中

间路由器简化为两条独立的信道,一条用于发送数据,另外一条用于发送 ACK 包。

4) 忽略数据包头部的其他字段,只考虑头部中的序列号字段。

4.2 CPN 模型

SUNA 可靠数据传输机制 CPN 模型如图 5 所示。模型包括三个部分:发送方、接收方以及传输数据的信道。发送方包括两个位置:“S_win”和“NextSend”以及三个变迁:“SendData”,“timeout”和“ReceiveAck”。接收方包括一个位置“R_win”和一个变迁“ReceiveData”。信道包括两个位置“DataChannel”和“AckChannel”。

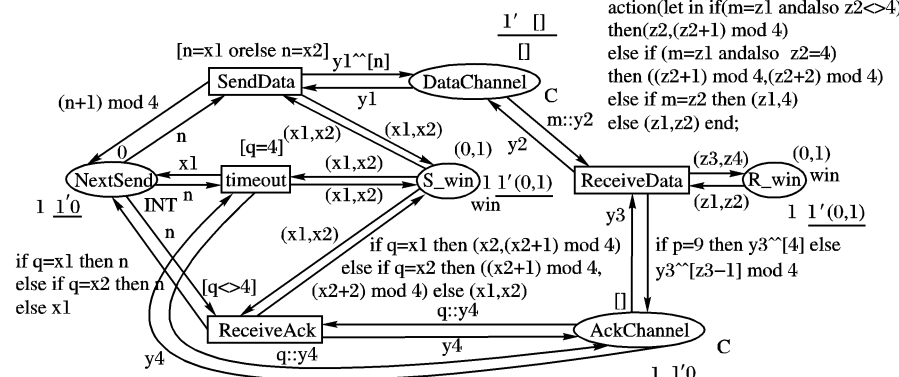


图 5 SUNA 可靠数据传输机制 CPN 模型

模型的左边描述发送方行为。位置“S_win”表示发送窗口的状态,它的初始标识是 $1'(0,1)$,表示会话开始时发送窗口的下界是 0,上界是 1。位置“NextSend”表示发送方发送的下一个数据包,由 ACK 的值和当前发送窗口共同决定。位置“NextSend”的初始标识是 $1'0$,表示发送方发送的第一个数据包是 0#数据包。变迁“SendData”将数据包发送到信道上,并且返回下一个数据包的包号。变迁“timeout”通知发送方重传丢失的数据包。变迁“ReceiveAck”从信道接收 ACK,接着改变滑动窗口上、下界,并且返回下一个数据包的包号。

模型的右边描述接收方行为。位置“R_win”表示接收窗口的状态,它的初始标识是 $1'(0,1)$,表示会话开始时接收窗口可以接收 0#数据包和 1#数据包。变迁“ReceiveData”从信道接收数据包,接着改变接收窗口,并且向信道发送 ACK。

模型信道部分包括两个位置。位置“DataChannel”的 Token 值表示正在从发送方向接收方传输的数据包序列号,位置“AckChannel”的 Token 值表示正在从接收方向发送方传输的 ACK 包序列号。

5 状态空间分析

CPN Tools 提供的一个重要工具就是状态空间分析工具。

通过状态空间报告可以对系统的有界性(Boundedness)、回归性(Home)、活性(Liveness)以及公平性(Fairness)进行分析。

表 1 SUNA 可靠数据传输机制 CPN 模型状态空间分析的部分报告

Statistics	Liveness Properties	Home Properties
Nodes: 749	Dead Markings: None	
Arcs: 1648	Dead Transitions Instances: None	Home Markings: All
Secs: 0	Live Transitions Instances: All	
Status: Full		

表中第 1 列描述了状态空间的统计信息。数据显示,本模型的状态空间包括 749 个节点和 1648 条弧;生成状态空间

所用的时间不到 1s。第 2 列描述了状态空间的活性。本模型没有死标识和死变迁,所有变迁都是活的。因为当发送数据序列号为滑动窗口上界时,发送方继续发送序列号为滑动窗口下界的数据,这是一个不中断的循环发送过程。第 3 列表明所有标识都是回归标识,因为整个传输过程是一个循环过程,经过若干步后总会回到当前标识。以上报告显示,SUNA 可靠数据传输机制是完全正确、可靠的,满足系统的有界性、回归性以及活性等方面的要求。

参考文献:

- [1] TENNENHOUSE D, WETHERALL D. Towards an Active Network Architecture[J]. Computer Communication Review, 1996, 26(2).
- [2] LAZAR A. Programming Telecommunication Networks [A]. Proceedings of 5th International Workshop on Quality of Service [C]. New York, USA, 1997. 3-24.
- [3] BOECKING S. 面向对象的网络协议[M]. 严伟,译. 北京:机械工业出版社, 2000.
- [4] BOECKING S, SEIDEL V, VINDEBY P. CHANNELS - A Run-Time System for Multimedia Protocols [A]. 4th International Conference on Computer Communications and Networks [C]. Las Vegas, USA, 1995. 178-185.
- [5] BRADEN B, FABER T, HANDLEY M. From Protocol Stack to Protocol Heap-Role-Based Architecture [A]. First Workshop on Hot Topics in Networking [C], 2002.
- [6] 曾家智,徐洁,吴跃,等. 服务元网络体系结构及其微通信元架构[J]. 电子学报, 2004, 32(5).
- [7] JENSEN K. Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use, Volume1 - Basic Concepts [M]. Springer-Verlag, 1992.
- [8] JENSEN K. Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use, Volume3 - Practical Use [M]. Springer-Verlag, 1997.

(上接第 1522 页)

参考文献:

- [1] 史美林,向勇,杨光信,等. 计算机支持的协同工作理论与应用 [M]. 北京:电子工业出版社, 2000.
- [2] SIBERSCHATZ A, KORTH HF, SUDARSHAN S. Database System Concepts (Fourth Edition) [M]. 北京:机械工业出版社, 2003.
- [3] 邵佩英. 分布式数据库系统及其应用 [M]. 北京:科学出版社, 2000.
- [4] CHEN G, LI T, LIAO G. A Multi-granularity Locking Protocol Based on Ordered Sharing Locks in Engineering Databases that Sup-

ports Cooperative Design [A]. 2004 International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES 2004) [C], 2004. 552-558.

- [5] 李陶深,廖国琼. 支持工程合作设计事务的扩充分层封锁机制 [J]. 计算机应用与工程, 2002, 38(7): 47-49.
- [6] 齐进,张家明,周伯鑫,等. 工程数据库中一种支持合作设计事务的并发控制机制 [J]. 计算机研究与发展, 1998, 35(11): 987-990.