

支持自愈恢复的 WMPLS 自组网组播协议

陈 静¹, 杨星海², 马秀明¹

(1. 山东省计算中心, 济南 250014; 2. 济南大学信息科学与工程学院, 济南 250022)

摘要: 基于无线多协议标签交换(WMPLS)协议体系, 结合无线移动自组网的特点, 提出一种支持自愈恢复的 WMPLS 信令建立标签交换路径的自组网组播协议 RA-WMPLS。该协议基于 LDP 信令协议做了组播扩展, 使用 LDP 消息进行组播树的建立和维护, 并建立 LSP 自愈恢复机制, 既实现组播的功能, 又能够快速恢复中断的链路, 提高网络的性能。通过构建 Ad Hoc 网络仿真模型, 仿真对比 RA-WMPLS 和 ODMRP 路由协议的性能, 并验证了自愈恢复特性。

关键词: 无线多协议标签交换; 自组网; 组播协议; 自愈恢复

WMPLS Multicast Protocol for Ad Hoc Network Supporting Recovery

CHEN Jing¹, YANG Xing-hai², MA Xiu-ming¹

(1. Shandong Computer Science Center, Jinan 250014; 2. School of Information Science and Engineering, Jinan University, Jinan 250022)

【Abstract】 Based on framework of Wireless Multi-Protocol Label Switching(WMPLS) protocol and characteristic of mobile Ad Hoc network, this paper proposes a multicast protocol RA-WMPLS for Ad Hoc network using WMPLS signaling protocol supporting recovery to establish Label Switch Path(LSP). With label messages and the extending of existing messages introduced to LDP signaling protocol, RA-WMPLS sets up and maintains multicast tree. It realizes multicast function, recovers from a link or node failure, improves network performance. This paper builds an Ad Hoc network simulation model, analyses and compares the performance of RA-WMPLS and ODMRP, and validates its recovery.

【Key words】 Wireless Multi-Protocol Label Switching(WMPLS); Ad Hoc network; multicast protocol; recovery

1 概述

移动 Ad Hoc 网络^[1]是一类具有移动、多跳、自组织特性的无线网络, 节点故障或节点移动经常造成链路故障, 使网络拓扑结构发生变换、路由的生命期很短, 因此, 路由的故障恢复在移动 Ad Hoc 网络中显得非常重要。尽管目前已经提出许多 Ad Hoc 网络组播协议(如 AMRoute, ODMRP^[2], AMRIS, CAMP), 但都很少考虑链路故障的恢复问题, 并且无法与具有应用前景的有线网 MPLS 技术实现有效的兼容。

将 MPLS 扩展成 WMPLS 的研究在国际上刚刚起步, Jong-Moon Chung 最早建立了 WMPLS 的协议体系^[3], 阐述了在移动通信网和自组网实现 WMPLS 的基本思路。国内成都电子科技大学余敬东等人提出了一种基于 WMPLS 的 Ad Hoc 网 QoS 路由协议^[4], 主要为 Ad Hoc 自组网提供了一种实现 QoS 的方法; 吉林大学李梦阳等人提出了一种基于 WMPLS 无线自组网组播机制^[5], 将 MPLS 的分组转发以及支持服务质量、流量工程等方面的技术优势引入无线自组网协议设计中, 没有考虑链路恢复问题。

鉴于此, 本文提出一种支持自愈恢复的 WMPLS 自组网组播协议 RA-WMPLS(Recoverable Ad Hoc-WMPLS)。

2 基于WMPLS的Ad Hoc网络基本结构

本文扩展 LDP 信令系统来支持组播, 采用源节点发起组播树的建立。基于 WMPLS 技术的 Ad Hoc 网络的基本结构, 包括以下几个模块(图 1): 当节点收到上层来的组播分组时, 将具有相同组播组地址和源地址的组播分组指定到相同的 Fi 上, 然后查看节点维护的 LSP 表, 如果存在 LSP 通道, 则打上标签转发该组播分组。若未找到相应的 LSP 通道, 则进入

路由协议模块进行路由请求过程。

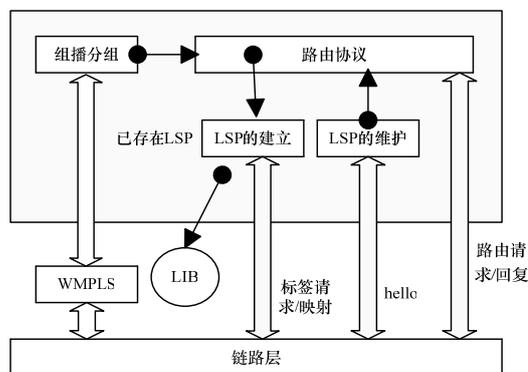


图 1 基于 WMPLS 技术的 Ad Hoc 网络结构

(1)路由协议模块

路由协议利用 2 种消息: 路由请求和路由回复。当需要建立一条路由时, 产生一条包含访问节点记录序列的路由请求消息, 中间节点将加入到记录序列中。一旦请求消息到达目的节点, 将产生一个路由回复消息, 沿着反向路径传回。

(2)LSP 建立模块

LSP 建立协议基于 2 种消息: 标签请求和标签映射。标

基金项目: 山东省自然科学基金资助项目(Q2005G02)

作者简介: 陈 静(1978—), 女, 助理研究员、硕士, 主研方向: 无线局域网测试; 杨星海, 讲师、在职博士研究生; 马秀明, 高级工程师

收稿日期: 2008-11-05 **E-mail:** chenj@keylab.net

签请求消息传到目的地后,会产生标签映射消息,标签映射消息携带用于 LSP 建立的标签绑定,修改反向路径上的每个节点的标签信息库(LIB),WMPLS 将利用此表通过标签交换来转发组播分组。标签请求和标签映射消息分别被包含在路由请求和路由回复消息中。

(3)LSP 维护

LSP 维护模块通过节点周期性发送 hello 消息,发现是否离开了邻节点的覆盖范围,判断路径是否中断。如果一定时间内未收到此消息,将触发 WMPLS 自愈恢复。

3 RA-WMPLS路由协议

本文采用 LDP 信令协议进行选路和建立 LSP, LSP 的建立采用被动的按需路由机制;标签分发方式采用下游按需分发;标签控制方式采用有序的控制;标签信息的保持方式为保守的保持方式;采用的信令消息包括 hello、标签请求、标签映射、标签出错。RA-WMPLS 路由协议从 LSP 建立机制出发,扩充了 LDP 信令协议,添加了 hello 消息,扩展了标签请求和标签映射消息。

3.1 协议中符号和路由表定义

源节点分配给该包唯一序列号,每次启用路由机制时,该数值加 1; FEC TLV 为标签请求对应的 FEC 单元; LSPID 为 LSP 标识符。初始时刻,上一跳 IP 节点地址列表为空。

路由表由节点按需建立和维护,存储节点在路由请求和路由回复过程中记录路由信息,包括组播组地址、源地址、序列号、到达源节点的下一跳地址、路由有效期,当节点进行包的转发时,查找该表以获得相应的转发路由。

3.2 邻居发现

每经过一个 hello interval 的时间,节点就向相邻节点广播一个 hello 消息,每个节点维护一张邻居节点列表,用于存储邻居节点的基本信息,并根据新收到的 hello 信息,周期性地对表项的更改维护,若未收到邻居节点的 hello 消息,节点将删除其在邻居节点中的相应表项。

3.3 路由请求

源节点构造一个包含标签请求的路由请求消息向整个网络进行广播,其格式如表 1 所示。

表 1 路由请求消息格式

消息类型标识符(0X0400)	标签映射消息长度
标签映射消息标识符	
FEC TLV	
LSPID	
Label TLV	
序列号	
组播组 IP 地址	
源节点 IP 地址	
下一跳节点 IP 地址	

中间节点收到广播的请求消息后,将执行如下操作:

(1)比较路由请求消息中与路由表中目的节点序列号的大小,如果小于,则说明该消息不是最新或是出现路由环路,丢弃该请求。

(2)如果大于,则比较该节点地址与组播接收者地址是否相同,若相同,则进行路由回复;若不同,则将请求消息中组播组地址与路由表中的记录比较,如果两者相同,则更新路由表,若不同,则保存相应路由信息到路由表中。

(3)添加上一跳节点 IP 地址到邻节点表中,如果邻节点列表中没有关于上一跳节点的信息,则创建一条记录。

(4)验证 TTL,若到期,则丢弃该消息;若未到期,则更新路由请求消息中上一跳节点 IP 地址为该节点 IP 地址,继

续广播该消息,直到遇到组播接收者。

3.4 路由回复

路由请求消息遇到组播接收者后,将执行以下操作:

(1)比较路由请求消息与路由表中序列号大小,若小于,则该消息不是最新,丢弃该消息。

(2)若大于,则对最先到达的路由请求消息进行回复,沿源路由的反向路径发送一个路由回复消息,格式见表 2,初始发送时,Label TLV 为 FEC 到标签映射中的标签,下一跳节点 IP 地址为路由请求消息中上一跳节点 IP 地址。

表 2 路由回复消息格式

消息类型标识符(0X0401)	请求消息长度
请求消息标识符	
FEC TLV	
LSPID	
序列号	
组播组 IP 地址	
源节点 IP 地址	
上一跳节点 IP 地址	
生存期 TTL	

中间节点收到路由回复消息后,将执行以下操作:

(1)比较路由回复消息中与路由表中序列号的大小,若小于,则丢弃该消息。

(2)若大于,则将路由回复消息中的下一跳节点 IP 地址与邻居节点列表中的相比较,验证路由节点的正确性和完整性。

(3)若验证通过,则在相应的路由表中查找源路由信息,若不存在路径,丢弃该消息。

(4)若存在路径,则分配一个新的标签作为该节点的入标签和其上游节点的出标签,并将该标签写入标签映射消息中,如果该节点为路由回复消息的第 1 跳节点,则保存入标签和组播组地址到该节点的 LIB 表中。

如果是其他中间节点,则在该节点 LIB 表中添加(Li, Lo, nexthop),其中 Li 为该节点新分配的入标签,Lo 为标签映射消息中携带的该节点的出标签, nexthop 为下游节点的地址。

源节点收到路由回复消息后,验证序列号是否最大,若验证通过,则在源节点的转发信息表(FIB)中添加(Fi, Li, nexthop),在 LSP 表中添加(Fi, LSPID),这样一连串的标签就构成了一条合法的路径 LSP。

3.5 路由维护

当节点发送 hello 消息,发现离开了邻节点的覆盖范围,常会引起节点或链路中断。这时将触发路由出错消息通知源节点,触发自愈恢复过程。

3.5.1 路由协议响应

当节点或链路失败时,上游节点立即构建一个路由出错消息(表 3)通知源节点,消息发送过程中,中间节点将执行:

(1)比较路由错误消息与路由表中序列号的大小,若小于,丢弃该消息。

(2)若大于,继续转发该消息,并删除路由表、LIB 表、FIB 表、邻接点表中相关信息记录。

表 3 路由出错消息

消息类型标识符	消息长度
路由出错消息标识符	
FEC TLV	
LSPID	
序列号	
组播组 IP 地址	
源节点 IP 地址	
发送节点 IP 地址	
下一跳节点 IP 地址	

3.5.2 LSP自愈恢复过程

当发生链路或节点故障时,假设在 N_f 出现了节点故障或在 N_f 与 N_{f+1} 间出现了链路故障,通过邻居发现机制, N_{f+1} 都将检测到故障并发送自愈恢复信号。因此, N_{f+1} 可视为自愈恢复的发起者(Initiator Of Recovery, IOR)。IOR N_{f+1} 将尝试绕过故障与上游路由器 N_{f-1} 建立一条路径,采用下游自主标签分发方式。

在 N_{f-1} 处恢复路径将与原 LSP 合并,因此, N_{f-1} 称为恢复的合并点(Merge Point, MP)。如果是特殊情况, S 与 N_1 间的链路中断,则输入节点 S 将成为 MP。IOR 向 MP 发送一个恢复 LSP 设置的消息(Recovery LSP Setup, RLS), RLS 消息包含一个恢复标志、MP 的 IP 地址、恢复 LSP 的 LSP 标识符 LSP_1 、原 LSP 的路由序列以及标签映射消息。

通过基本的逐跳路由机制, RLS 消息被送往 MP。当恢复路径上的节点 N_r 收到 RLS 消息,将采取下游自主标签分发方式。从而 N_r 将进行下列过程:

- (1) N_r 接收到包含在 RLS 消息中的标签,存储标签并将它和 LSP_1 绑定。
- (2) 若 N_r 不属于 LSP_1 , N_r 将把自己插入到路由序列中合适的位置上,刷新路由序列。
- (3) 若 N_r 属于 LSP_1 , 且 $r > f+1$, 即 N_r 位于 IOR 的下游,在这种情况下,丢弃该消息。
- (4) 若 N_r 属于 LSP_1 , 且 $r < f-1$, 即 N_r 位于 IOR 的上游,则替换新的 LSP 绑定,取消恢复标志设置,向 N_{r+1} 发送 RLS 的消息。
- (5) 若 $N_r = MP$, 替换新的 LSP 绑定,若 N_f 有故障,终止;若 N_f 无故障,向 N_f 发送 RLS 消息,在 N_f 去除绑定后,终止。

图 2 描述当 N_3 与 N_4 间发生链路故障时, N_4 成为 IOR, N_2 为 MP。 N_4 沿通向 N_2 的最短路径,向 N_x 发送 RLS 消息。 RLS 消息经过 N_x 和 N_y , 最后被 N_2 接收,因此,首先是 N_x 成为 N_r , 然后是 N_y , 最后是 MP, N_2 。一旦 N_2 的标签绑定完成,数据流量就可以转向新路径。

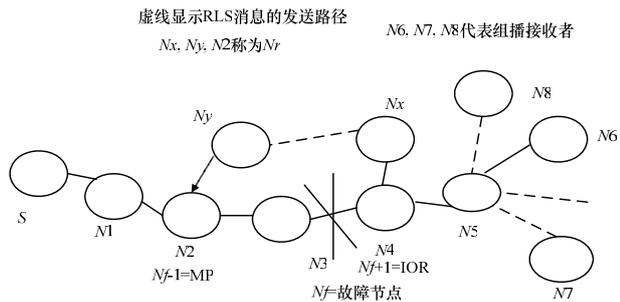


图 2 故障自愈恢复过程

3.5.3 LSP 更新过程

MP 产生一个更新通知消息(Update Notification Message, UNM),包含更新后的路由序列。MP 向源节点 S 发送 UNM,源节点收到后,通过新的路径发送 LSP 参数更新消息(Parameter Refresh Message, PRM)。中间节点收到 PRM 后,更新它们的路由序列。如图 2, N_2 向 S 发送 UNM,然后, S 触发 LSP 更新过程,发送 PRM,PRM 通过 $N_1, N_2, N_y, N_x, N_4, N_5$, 最后到达组播接收者。这些节点更新 LSP_1 的路由序列。

4 仿真及结果分析

RA-WMPLS 路由协议的仿真采用 NS2 网络仿真器,设置 100 个节点,节点采用 Random Waypoint 运动模型,随机分布于 $1500 \text{ m} \times 1500 \text{ m}$ 的矩形区域中,最大速度为 20 m/s 。运动模型的 pause time 取 $0, 100 \text{ s}, 200 \text{ s}, 300 \text{ s}, 450 \text{ s}, 600 \text{ s}, 750 \text{ s}$ 和 900 s ,即改变网络中节点的移动性设置 8 种仿真环境。为了更直观地评估路由协议性能,网络流量由 CBR 数据源产生,定义源每秒产生 5 个数据包(5 packets/s),每个包 512 Byte。本文模拟了 3 种情况,如图 3~图 5 所示。

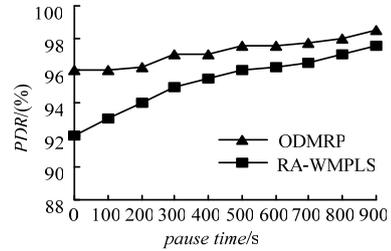


图 3 1 个组播源 10 个组播接收者的分组传递率

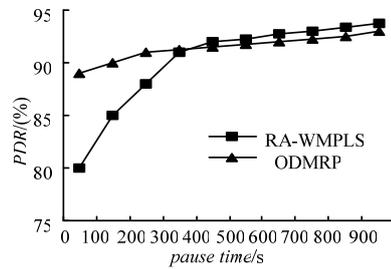


图 4 2 个组播源 10 个组播接收者的分组传递率

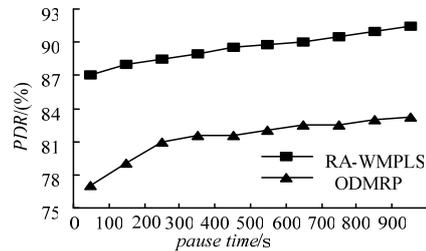


图 5 5 个组播源 10 个组播接收者的分组传递率

图 3 表明 1 个组播源 10 个组播接收者情况下 ODMRP 和 RA-WMPLS 的分组传递率都比较高,总体上 RA-WMPLS 比 ODMRP 差,主要是单源节点下,RA-WMPLS 没有区域的冗余路径,在高速移动场景下,丢包率较 ODMRP 高。图 4 为 2 个组播源 10 个组播接收者的情况,看到在 $pause \ time = 300 \text{ s}$ 后 RA-WMPLS 分组传递率高于 ODMRP,这是由于增加了 1 个组播源,ODMRP 需要引入更多的网络节点作为转发节点,在转发数据包时造成大量的竞争和冲突,因此丢包率增加,数据分组传递率比 1 个组播源时要低。RA-WMPLS 虽然随着新组源的增加,分组传递率也有所下降,但由于 RA-WMPLS 采用标签转发机制,数据转发被控制在有效的路径上,并采用自愈恢复机制,能够快速恢复数据分组传输路径,因此分组发生冲突的概率小。图 5 为 5 个组播源 10 个组播接收者的情况,ODMRP 使用大量节点作为转发节点,在传输数据包时造成大量的竞争和冲突,而且每个源节点需要周期性全网洪泛更新路由,加剧了拥塞程度,性能随着源节点个数的增多而显著下降。而 RA-WMPLS 将数据包控制在有效路径上,并采用自愈恢复机制,使数据包受拥塞的影响比较小,性能优于 ODMRP。

(下转第 88 页)