摘　要

IPv6 最明显的优势是拥有巨大的地址容量，能够满足互联网飞速发展的需求，是集移动性、安全性和服务质量为一体的最佳技术选择。移动 IPv6 技术的引入更为 IPv6 带来了许多新的特性和应用，移动通信将成为 IPv6 技术最先得到大规模应用的领域。本文首先介绍了移动 IPv6 的技术原理，接着给出了移动 IPv6 协议在实时嵌入式操作系统 Vxworks 下的实现方案，并从功能与性能两个方面对本文所实现的协议栈软件系统进行了测试；最后在无线局域网环境下实际构建了一个基于移动 IPv6 技术的多媒体演示环境。

Abstract

The most advantage of IPv6 is its vast capacity of addresses. It can meet the need of high-speed development of Internet, and it is also the best choice of technologies for combination of mobility, security and quality. The introduction of mobile IPv6 technology brings more new properties and applications to IPv6. Mobile communications will be the first area in which IPv6 is first widely used. In this paper, we introduce first the basic principle of Mobile IPv6 protocols and present the implementation scheme of the protocols in Vxworks. Then we analyze the function and performance of the implementation. Finally, we build a multi-media environment of Mobile IPv6 technology in WLAN to demonstrate the conclusions of the paper.

引 言

IPv6 的出现引起了世界重要研究机构和公司的重视。目前 IETF 正在制定大量的 IPv6 相关标准，包括地址结构、域名解析、安全、自动配置、邻居发现和路由协议等方面，同时为了对 IPv6 协议特性进行研究并积累 IPv6 组网经验，IETF 于 1996 年建立了全球范围的试验床，称作 6Bone。6Bone 是一个虚拟的网络，以隧道的方式通过基于 IPv4 的互联网实现互联。

中国对 IPv6 的研究始于 1998 年，主要参与者是一些高等院校和研究机构。对中国而言，IPv6 的发展将带来巨大机遇。作为互联网和移动通信大国，基于 IPv6 的互联网将在我国从基础设施、服务与应用、媒体与内容、设备制造等层面形成新的巨大产业，拉动经济的发展。2003 年下半年，国家下一代互联网示范工程（CNGI）正式启动，至此我国以 IPv6 为基础核心协议的下一代互联网产业的发展进入了一个实质性阶段。项目资金 14 亿元，项目有 20 多个接点，分布在全国主要省会城市，被认为是目前世界上覆盖范围最广的 IPv6 试商用项目。根据 CNGI 的规划，我国将在 2005 年底建成一个覆盖全国的 IPv6 网络，届时将成为世界上最大的 IPv6 网络之一。

IPv6 与移动通信的结合将为下一代互联网的应用开辟一个全新的领域，无线将成为 IPv6 的第一个“杀手级”应用。移动互联网上有许多新型而精彩的服务，IPv6 将是实现这些服务的关键。不久的将来，当每个人都要携带一个或多个移动终端时，IPv6 将为所有的移动终端提供唯一的 IP 地址。不论 IPv6 的其他优点，单就这一项功能就可以实现个人之间的直接通信。IPv6 对移动性的支持简称移动 IPv6，移动 IPv6 技术是 IPv6 与移动通信相结合的基石。IPv6 技术的迅猛发展及其与移动通信的密切结合，将使移动 IPv6 得到广泛的应用。

本论文对移动 IPv6 技术进行了较为深入的研究。首先介绍了移动 IPv6 的技术原理；接着给出了在实时嵌入式操作系统 Vxworks 下移动 IPv6 协议的实现方案，并从功能与性能两个方面对协议实现进行了测试和分析；最后，在无线局域网环境下实际构建了一个基于移动 IPv6 技术的多媒体演示环境。

第一章 移动 IPv6 技术概述

2000 年 5 月，负责制订 3G 规范的 3GPP 决定在未来的 3G 网络中采用 IPv6，这为 IPv6 的推广和应用提供了实际的驱动力。经过几年来的 IPv6 开发和推广工作，世界各地的研究机构以及各大设备制造商也开始认识到，由于市场因素的影响，IPv6 要取代 IPv4 如果仅仅依靠技术上的优势或者等待 IPv4 地址真正耗尽是不现实也是消极的。必须利用 IPv6 的新特性，开发新的业务，吸引运营商积极采用和推广 IPv6 技术及网络设备，以此来促进 IPv6 走向实用。由于移动通信技术的发展，互联网上已经出现了越来越多的移动终端，其发展速度超过了任何预测。如果这些移动设备都要实现与互联网互联，所需的巨大地址资源是 IPv4 所不能提供的。基于以上原因，目前业内普遍认为，IPv6 协议将首先在移动通信领域获得大规模的应用。

IPv6 对移动性的支持简称移动 IPv6 [1-3]，本文在叙述过程中对术语“移动 IPv6”和“移动 IP”不作严格的区分，它们都表示 IPv6 的移动性。IPv4 对移动性的支持简称为移动 IPv4。本章首先对 IPv6 技术进行简要的介绍[4-15]，主要是在移动性支持中涉及到的 IPv6 新特性，然后具体介绍 IPv6 对移动性的支持，主要是移动 IPv6 的工作原理，最后指出本论文的课题背景和主要工作。

1.1 IPv6 概述

图 1-1 给出了 IPv6 的报头格式。从图中可以看到，除了地址长度定义为 128 比特以外，相对于 IPv4，IPv6 还对报头进行了简化和修改，其中比较引人注目的就是在 IPv6 中取消了报头校验和字段。

版本	优先级	流标识	
净荷长度		下一协议头	跳数限制
源IP地址			
目的IP地址			
扩展报头			

图 1-1 IPv6 的报头格式

另外，在 IPv6 中取消了选项字段，而代之以扩展报头，当一个 IPv6 报文带有额外的信息时，这些信息就包含在扩展报头中，IPv6 报头中的“Next Header”字段指明是否有扩展报头以及扩展报头的类型，扩展报头中也带有“Next Header”字段，这样，可以形成一个扩展报头链，在最后一个扩展报头后面就是实际的报

文数据。

在 IPv6 中定义了一种目的地选项扩展报头，这种报头中包含了某些选项，一般只在数据包到达最终的目的地之后才进行处理。IPv6 中还定义了一种路由报头，在其中列出了数据包在转发过程中必须经过的路由节点，即完成类似于 IPv4 中的“宽松的源路由”选项的功能。在移动 IPv6 中主要使用了这两类扩展报头。

在 IPv6 中，取消了广播的概念，只有组播的概念，而把 IPv4 中原来的广播功能作为组播的一种特殊形式。在 IPv6 中还支持了一种新的地址类型，称为任播地址，任播地址和组播地址有些类似，发送节点向一组接收节点发送数据，但和组播不同的是，并不是这组接收节点的所有成员都能收到任播报文，实际接收到报文的节点是离发送方最“近”的那个节点，这里,“近”的具体含义由网络的路由策略决定。

在 IPv6 中，新定义了一种称为邻机发现的自动配置机制，当网络节点完成引导以后，就利用邻机发现完成一些基本的配置功能，其中的几个自动配置包括：

路由器发现，用于搜索节点所连接的网络上的路由器；

前缀发现，IPv6 的地址采用类似 CIDR 的结构，通过一个网络地址（即地址前缀）指明节点所在的网络，前缀发现就用来查找当前网络的网络地址；

自动地址配置，网络节点利用当前的网络地址前缀结合一个唯一的节点接口标识来形成一个 128 位的地址，这个地址一般只在本链路范围内有效，称为链路局域地址；

地址解析，完成类似于 IPv4 中的 ARP 协议的功能。

网络上的路由器利用路由器通告周期性地发布当前的网络前缀，其他节点就利用这些地址前缀结合一个唯一的接口标识（对于以太网，通常就基于网卡的以太网地址），形成一个链路局域地址。网络上的其他节点也可以通过主动地向路由器发送路由器请求来主动查找网络上的路由器，或者要求路由器响应一个包含网络地址前缀的路由器通告，然后利用获得的前缀信息配置地址。

在网络节点利用 IPv6 的自动配置功能真正给接口分配 IP 地址之前，需要检查这一个自动配置得到的地址是否已经被链路上的其他网络节点使用，这在 IPv6 中称为“重复地址检测”，在节点配置好地址之后分配给接口之前，这一地址称

为“试探性地址”，节点通过向该试探性地址的“被请求组播地址”发送邻机请求报文来检查该试探性地址是否已经被其他节点使用，这个被请求组播地址由一个专用的组播地址前缀和试探性地址的后面若干位结合得到。如果已经有其他节点正在使用这个地址，那么使用这个地址的节点就会回应一个邻机通告，声明该地址已经被使用。如果经过重复地址检测没有发现其他节点使用这一地址，那么这个地址就可以真正分配给所配置的接口了。在 IPv6 中通过自动配置得到的地址一般都有一定的生存期，在这一生存期快结束的时候，如果节点还要使用该地址，就需要向分配地址前缀的路由器发送路由器请求，以扩展该地址的生存期。

在 IPv6 中，邻机发现机制还用来完成原来在 IPv4 中的 ARP 的地址解析功能，当一个网络节点需要进行地址解析的时候，就向“所有节点组播地址”发送邻机请求报文，在这个报文中包含有需要解析的 IP 地址，当正在使用该地址的节点接收到这种邻机请求报文后，就会向发送该请求的网络节点回应一个邻机通告，告知对方与自己的 IP 地址对应的链路层地址。

以上介绍的这些 IPv6 基本机制都在移动 IPv6 中得到了应用。

1.2 移动 IPv6 概述

移动 IPv6 的 RFC 规范还没有最终形成，但大量的工作正在开展，在 IETF 的讨论组中，几乎每天都有邮件涉及到移动 IPv6。目前定义移动 IPv6 的互联网草案已经是第 24 个文本。和早期的草案文本相比，其篇幅要大得多，主要是引入了对安全性的考虑，因此对早期的文本改动很多，包括报文格式等都有较大的改动，并加入了必要的安全性机制。但是，移动 IPv6 的基本工作原理已经比较固定，因此本章不过多地涉及具体的报文格式等内容，只对移动 IPv6 的基本原理进行介绍。

1.2.1 移动 IPv6 和移动 IPv4 的比较

当网络节点从一个网络移动到另一个网络后，其 IP 地址必然要随着改变。所谓 IP 节点的移动性是指在 IP 网络当中，某些网络节点可以在与其他节点进行通信的同时，离开当前的网络，接入到另外一个网络，而在移动的过程中保持通信不会中断。任何与移动节点进行通信的其他网络节点都称为一个通信节点，这个通信节点本身也可以是移动的节点。如果移动节点在移动的过程当中需要保持与通信节点之间的通信，比如一个 TCP 连接，就需要保证移动节点在从一个网

络移动到另一个网络后，其 IP 地址的改变对 TCP 连接是“透明的”，即 TCP 连接仍然可以使用移动节点原有的 IP 地址。移动 IP，包括移动 IPv4 和移动 IPv6 的目的，就是要提供一套机制，来将移动性屏蔽起来，保证 IP 层以上的协议，如 TCP 协议以及应用程序等，能够不受节点移动的影响在不需要任何改动的情况下照常工作，有了移动 IP 的支持，这些协议和应用程序就感觉不到移动性的存在，好像节点并没有移动一样。

在 IPv4 中，也可以实现对移动性的支持，下面对移动性支持在 IPv4 和 IPv6 中的主要区别做一个比较，IPv4 和 IPv6 对移动性的支持主要存在如下的不同：IPv4 中的移动性是作为一种后期附加的功能提供的，并不是所有的 IPv4 网络节点都能够支持移动性，特别是早期的 IPv4 移动性支持中，普遍存在所谓的“三角路由”问题。在移动 IP 中，节点原来所在的网络称为本地网络，如果节点离开本地网络，接入到与本地网络不同的另外一个网络中，这个新的网络称为异地网络或者被访问网络。在移动 IPv4 中，为了完成移动功能，需要在本地网络有一个路由器作为移动节点的家乡代理，同时还需要在异地网络中有一个路由器作为外地代理。如果不支持路由优化，那么由通信节点发往移动节点的数据包就要经过“通信节点→家乡代理→外地代理→移动节点”这样一个近似三角形的转发路径，路由效率比固定节点之间直接的通信要低得多，这就是所谓的三角路由问题。后期的移动 IPv4 支持路由优化，可以克服三角路由问题，但并不是所有的移动 IPv4 都支持路由优化。不同于 IPv4，在 IPv6 中，对移动性的要求是内嵌的，在设计 IPv6 协议的同时就考虑到了对移动性的支持。在移动 IPv6 中，要求支持路由优化，从根本上消除了三角路由问题。移动 IPv6 利用邻机发现和自动配置机制，使得节点不需要被访问网络提供任何特殊的机制就能实现移动性，取消了外地代理。

IPv6 中，当移动节点移动到异地之后，利用地址自动配置在异地网络中自动配置一个新的地址，称为移动节点的转交地址，移动节点利用这个地址作为自己向外发送的数据包的源地址，这样，数据包可以正常的通过实行侵入过滤的路由器。移动节点在本地网络中使用的地址称为本地地址，移动节点的本地地址包含在其向外发送的数据包的本地地址目的地选项中，在数据包到达目的节点后，IP 层会将源地址替换为移动节点的本地地址，从而使转交地址的使用不为上层

协议所知。

IPv6 中，利用转交地址作为 IP 报文的源地址，简化了组播的路由。在 IPv4 中，为了能够在组播报文中“透明”地使用本地地址，移动节点需要把报文通过隧道方式发送到自己的家乡代理。

在移动 IPv4 中，报文都通过隧道的方式发送到移动节点。而在 IPv6 中，除了通过家乡代理截获的报文以外，所有发送到移动节点的报文都使用路由扩展报头进行发送，比隧道方式简单，降低了发送移动 IP 报文的开销。

IPv6 中，当移动节点不在本地网络的时候，家乡代理利用邻机发现机制截获报文，其效率比移动 IPv4 使用的 ARP 高。

移动 IPv6 利用任播地址而不是组播地址进行家乡代理地址搜索，大大减少了返回移动节点控制报文的流量。

在移动 IPv6 中，定义了一种返回可路由过程，用于验证移动节点在其声明的本地地址以及转交地址上可达，从而使通信节点可以对发送绑定更新报文的移动节点进行认证。

以上列举了移动 IPv6 和移动 IPv4 的几个主要的区别，从中可以看到，IPv6 对移动性的支持要比 IPv4 好得多。

1.2.2 移动 IPv6 中的基本工作原理

为了能够支持移动性，在 IPv6 中，移动节点的本地网络至少需要一个家乡代理，家乡代理负责在节点移动之后，截获通信节点仍然发送到移动节点本地地址的报文，并通过隧道的方式把截获的报文发送到移动节点的转交地址，即移动节点在异地网络中的地址。家乡代理还完成对移动节点的本地地址进行重复地址检测等功能。

移动 IPv6 的基本工作过程如下：移动节点不断地进行移动检测，检测自己是否移动到了另外一个网络，这个过程主要利用了邻机发现机制，当移动节点收到包含新的地址前缀的路由器通告，并且不再能够收到原来的路由器通告后，就认为自己发生了移动。当移动节点接入异地网络之后，首先利用邻机发现机制自动配置一个与异地网络中其他节点使用相同前缀的地址，这个地址称为该移动节点的转交地址；同时，移动节点仍然占有其在本地网络中的本地地址。移动节点的转交地址和本地地址之间的对应关系称为地址绑定，每一个绑定有一个生存

期。移动节点可以有多个转交地址，特别是在无线网络中，当移动节点移动到两个网络的传输重叠区域时，同时具有这两个网络的转交地址，对于移动节点的平滑移交非常重要。移动节点在配置好转交地址之后，需要向家乡代理以及通信节点通告这一转交地址，这主要通过移动节点主动地向家乡代理和通信节点发送绑定更新报文来实现。绑定更新报文中有一个比特的标识位，称为“本地注册标识位”或“H 标识位”，当这一标识位被设置，就说明移动节点在向本地网络通告自己的转交地址的同时，希望本地网络中接收这一绑定更新的路由器作为自己的家乡代理，这一过程称为“注册”。移动节点可以同时具有多个转交地址，但只能向家乡代理注册一个转交地址，这个地址称为移动节点的基本转交地址。

每一个支持移动 IP 的 IPv6 节点都有一个绑定缓存，其中保存了所知的各个移动节点的地址绑定信息。如果通信节点本身就是移动节点的家乡代理，那么在向移动节点发送 IPv6 报文的时候，就直接把报文发送到移动节点的基本转交地址；如果通信节点不是家乡代理，那么首先检查自己的绑定缓存中是否有与目的节点（即移动节点）的本地地址对应的转交地址，如果有，就利用路由扩展报头将报文发送到移动节点的转交地址，当报文到达移动节点以后，通过对路由扩展报头的处理，移动节点会自动地把 IPv6 报文中的源地址替换为扩展报头中包含的移动节点本地地址，这样，使得移动性对于移动节点的上层协议如 TCP 和应用程序都是透明的。绑定缓存中的每一个地址绑定都有一定的生存期，如果生存期将要结束的时候，通信节点还在使用这一绑定，那么就需要向移动节点发送绑定请求，移动节点在收到绑定请求之后，会发送一个对应的绑定确认，更新地址绑定的生存期。

如果通信节点没有目的移动节点对应的地址绑定，即不知道移动节点的转交地址，那么就把报文发送到移动节点的本地网络。移动节点的家乡代理利用代理邻机发现来截获这些报文。本地网络上，当其他节点利用邻机发现机制对移动节点的 IP 地址进行解析的时候，家乡代理就会发送对应的邻机通告，在通告中包含自己的链路层地址，家乡代理也会周期性地发送邻机通告，把自己的链路层地址作为移动节点的链路层地址向本地网络进行组播。这样，本来发往移动节点的报文就会被家乡代理截获，家乡代理通再过隧道封装的方式向移动节点转发这些截获的报文。移动节点接收到家乡代理利用隧道封装发送过来的其他通信节点的

报文后，就知道对应的通信节点没有自己的转发地址，于是就会向该通信节点发送绑定更新，向对方通告自己的当前位置，即自己的转交地址，在通信节点获得移动节点的绑定更新之后，后续的报文就直接利用路由扩展报头发送到移动节点，不再通过家乡代理。

在移动 IPv6 中，定义了一个新的目的地选项扩展报头，称为本地地址选项，移动节点向通信节点发送 IPv6 报文的时候，在发送到通信节点的报文中带有一个目的地选项扩展报头，其中包含一个本地地址选项，移动节点通常利用自己的一个转交地址作为 IPv6 报文的源地址，在报文到达通信节点之后，通信节点的 IP 处理模块利用扩展报头本地地址选项中包含的移动节点本地地址替换 IPv6 报文源地址字段中的转交地址，这样就实现了对通信节点 IP 以上的各个协议层屏蔽移动性的目的。

以上介绍了通信节点、移动节点和家乡代理之间通信的基本工作原理和工作过程。在移动 IPv6 中，还要处理一些相关的问题。

由于提供了邻机发现和地址自动配置机制，因此在 IPv6 中，网络的重新编号要比 IPv4 容易得多，IPv6 的地址一般都有一个有限的生存期，当发生网络重编号时，新的地址和老的地址可以共存一段过渡期，网络上的路由器通过路由器通告逐渐地减少老地址的生存期，同时配置新的地址，这样就可以实现平滑的地址重编号。在移动 IPv6 中，当节点移动到了异地网络之后，该移动节点原来所在的本地网络可能会发生重新编号。此时家乡代理需要利用代理邻机发现机制为该移动节点配置新的本地地址，这包括地址前缀的配置以及重复地址检测等。

在本地网络中，可以同时有多个路由器作为家乡代理。家乡代理和移动节点都保存有一个家乡代理列表，列表中的家乡代理有一个优先级，移动节点选择优先级最高的路由器作为自己的家乡代理，当优先级最高的路由器不可用时，就选择优先级次高的路由器，如果列表中某些家乡代理的优先级相同，则采取轮换的策略。

当节点配置了一个新的转交地址后，通常需要向家乡代理进行注册。如果本地网络上的家乡代理在移动节点移动的过程中发生了改变，即由原来不作为代理的路由器来代替原来的家乡代理。那么移动节点就可能不知道当前所有家乡代理的地址，此时移动节点需要向本地网络询问可用的家乡代理。在移动 IPv6 中，

设计了动态家乡代理发现机制来实现这一功能。如果移动节点不知道本地网络上家乡代理的地址，那么就向“移动 IPv6 家乡代理”任播地址发送 ICMP 请求报文，本地网络上的代理路由器都加入了这一任播地址组，因此离移动节点最“近”的那个路由器能够收到从移动节点发来的家乡代理地址发现请求报文。接收到请求报文的家乡代理路由器将向移动节点回应一个“家乡代理地址发现回应”消息，该消息包含了该家乡代理的全局单播地址，以及本地网络上所有家乡代理的广域全局单播地址列表。移动节点在收到家乡代理地址发现回应后，将把发送这一回应的路由器作为优先级最高的家乡代理，也可能使用家乡代理列表中的其他路由器作为家乡代理。

当移动节点通过移动检测发现自己又重新回到本地网络后，首先向家乡代理发送绑定更新，通知家乡代理不再为自己截获和转发来自其他通信节点的报文，即不再为移动节点完成代理邻机发现的功能。之后，移动节点通过向本地网络组播邻机通告，或者回应来自其他节点的邻机请求，通知其他节点，自己已经回到了本地网络。例如，移动节点通过组播邻机通告，向本地网络发布自己的链路层地址，或者，当本地网络上的某个节点通过组播邻机请求对移动节点的本地地址进行地址解析的时候，移动节点就回应一个单播的邻机通告，告知对方自己的链路层地址。网络上的其他节点获取了移动节点的链路层地址后就可以象对待非移动节点一样直接把报文发送到移动节点。

1.3 课题背景和论文工作

1.3.1 项目背景

本文的主要工作是在中兴通讯技术中心研究部 IPv6 项目组进行的，中兴通讯 IPv6 项目组为本文的工作提供了良好的实验条件和工作环境。中兴通讯于 2000 年开始切入 IPv6 研发，并将 IPv6 协议栈的研究和开发列入数据领域重点发展项目。中兴通讯是在 IPv6 的国家标准的积极的倡导者和参与者，参加了所有 IPv6 国家标准的起草和制定工作，并承担了国家 863 重大研究课题“高性能 IPv6 路由器关键技术及系统开发”。这些工作都是为了使公司在 IPv6 的研究上掌握核心技术，及早进行技术积累，为公司将来开发其他的 IPv6 产品奠定基础。

IPv6 协议栈软件项目是中兴通讯为满足下一代互联网需求而展开的纯软件项目，该项目的目标是研究掌握最新的 IPv6 规范体系，形成一套具有自主知识

产权的 IPv6 协议栈软件，掌握协议栈实现中的关键技术，此外，这个项目的研发成果将应用于国家 863 项目“高性能 IPv6 基础平台及试验系统”中去。本文对移动 IPv6 的研究是作为中兴通讯 IPv6 协议栈软件项目的一个研究课题而开展的。

1.3.2 本文工作

本文工作主要包括以下几个方面：

1. 本文确定了移动 IPv6 协议必须实现的功能集，完成了移动 IPv6 功能模块的划分，并给出了系统设计方案。
2. 在对协议和 KAME 协议栈进行对照分析的基础上，采用代码移植的方法，完成了移动 IPv6 协议在实时嵌入式操作系统 Vxworks 下的实现工作。这是本文最主要的工作之一。
3. 立足于现有的条件，对协议实现进行了基本功能测试，并具体给出测试环境、测试方法和测试结果。测试表明，我们实现了一个移动 IPv6 的基本功能集。
4. 对移动 IPv6 协议栈的性能进行了测试，并对测试数据进行了分析；
5. 利用论文工作成果，实际构建了一个无线环境下的移动 IPv6 多媒体演示环境，进行技术演示。

第二章 移动 IPv6 协议在 Vxworks 下的设计与实现

到本章撰写时为止，移动 IPv6 尚未形成正式标准。本章的主要设计目标是以 IETF 的最新草案为依据建立起移动 IP 的基本框架，为现有的 IPv6 协议栈提供移动性支持。移动 IPv6 模块的引入为 IPv6 协议栈加入了新的特性，使得网络节点改变网络连接点时，运行在节点上的应用程序不需修改或配置仍然可用。这些特性使得移动节点总能够通过家乡地址通信。这种机制对于 IP 层以上的协议层是完全透明的。移动 IPv6 影响了数据包的路由，但是却又独立于路由协议(如 RIP，OSPF 等)本身。本章着重介绍了移动 IPv6 协议在实时嵌入式操作系统 Vxworks 的实现方案。

2.1 需求分析

移动 IPv6 模块在利用 IPv6 协议栈所提供的邻机发现、自动地址配置、IPSec 机制和隧道机制的基础上，需要基本实现如下功能：

- 支持将使用该协议栈的网络节点配置为家乡代理或者移动节点，当该节点不被配置为家乡代理或者移动节点即作为通信节点时，能够协助其他节点完成移动功能；
- 维护“绑定缓存列表”；
- 维护“绑定更新列表”；
- 维护“家乡代理列表”；
- 支持发送和接收“绑定更新”报文；
- 支持发送和接收“绑定确认”报文；
- 支持发送和接收“家乡代理地址发现请求”报文和“家乡代理地址发现回应”报文；
- 支持第二种类型的路由扩展报头；
- 支持移动扩展报头及其选项；
- 如果接收到的 IPv6 报文中包含本地地址选项，能够正确处理本地地址选项；
- 支持使用该协议栈的网络节点完成“返回可路由过程”；
- 支持在路由器通告中使用通告间隔选项，并且该选项是可配置的；
- 可以配置自动发送路由器通告的时间间隔；
- 支持在路由器通告中发送包含完整路由器地址的路由器通告（包含 R 比特的

路由器通告）；

- 当配置为家乡代理时支持利用代理邻机发现截获发往本链路移动节点原来位置的报文；
- 支持隧道机制（包括封装与解封装），能够将截获的报文转发给位于异地网络的移动节点；

2.2 初步设计思路

为了实现对移动性的支持，移动 IP 模块将支持把协议栈的网络节点配置为家乡代理或者移动节点，当该节点不被配置为家乡代理或者移动节点即作为通信节点时，能够协助其他节点完成移动功能；此外，家乡代理和移动节点也有可能同时作为通信节点而存在，所以它们也必须能够完成通信节点所具备的全部功能。

经过对协议的阅读和分析，移动 IPv6 模块可以设计成如图 2-1 所示架构。在内部结构的设计过程中，依据实现的要求，可以将模块从功能上分解成为若干个子模块，包括绑定缓存表维护模块、绑定更新表维护模块、家乡代理上家乡代理表维护模块、移动节点上的家乡代理表维护模块、各种输入报文的处理模块、路由优化模块以及移动检测模块等等，每个模块将能够完成自身特定的相对单一的处理功能。同时，移动 IP 工作流程的特征要求从逻辑上将整个系统划分为三个子系统，即家乡代理子系统，移动节点子系统，通信节点子系统。但是这并不意味着在系统设计时必须将这三个子系统截然分开，设计成三个功能各异的独立模块。可以考虑将它们有机地糅合在一起，这样每个系统中功能相同、相近或者相关联的部分，可以使用同一个模块来加以实现。子系统完成某一特定功能的过程中，从相应功能模块中调用自己需要的那些函数，并执行自身在函数中所对应的流程，完成这一子系统在工作过程中所对应的处理。通过各子系统的协同工作，从而实现 IPv6 协议栈对移动性的支持。

在图 2-1 中，因为家乡代理和移动节点需要同时能够作为通信节点而工作，所以这两个子系统应该包含通信节点子系统所包含的全部功能模块。考虑到由于家乡代理 HA 和移动节点 MN 对家乡代理表的维护工作差别很大，因此为它们分别设计家乡代理表维护模块，即图中 HA 上的家乡代理表维护模块和 MN 上家乡代理表维护模块；而对于绑定缓存表，它们就可以使用同一个处理模块。此外，

移动节点在处理邻机通告时，可以先将 IPv6 协议栈原有的处理邻机通告的函数 nd6_ra_input 按照移动 IP 协议的要求进行修改，然后加以调用。鉴于家乡代理工作的复杂性和特殊性，另行设计一个 ra_input 函数（位于图中移动 IP 路由广播处理模块），用来进行移动 IP 的路由广播处理。

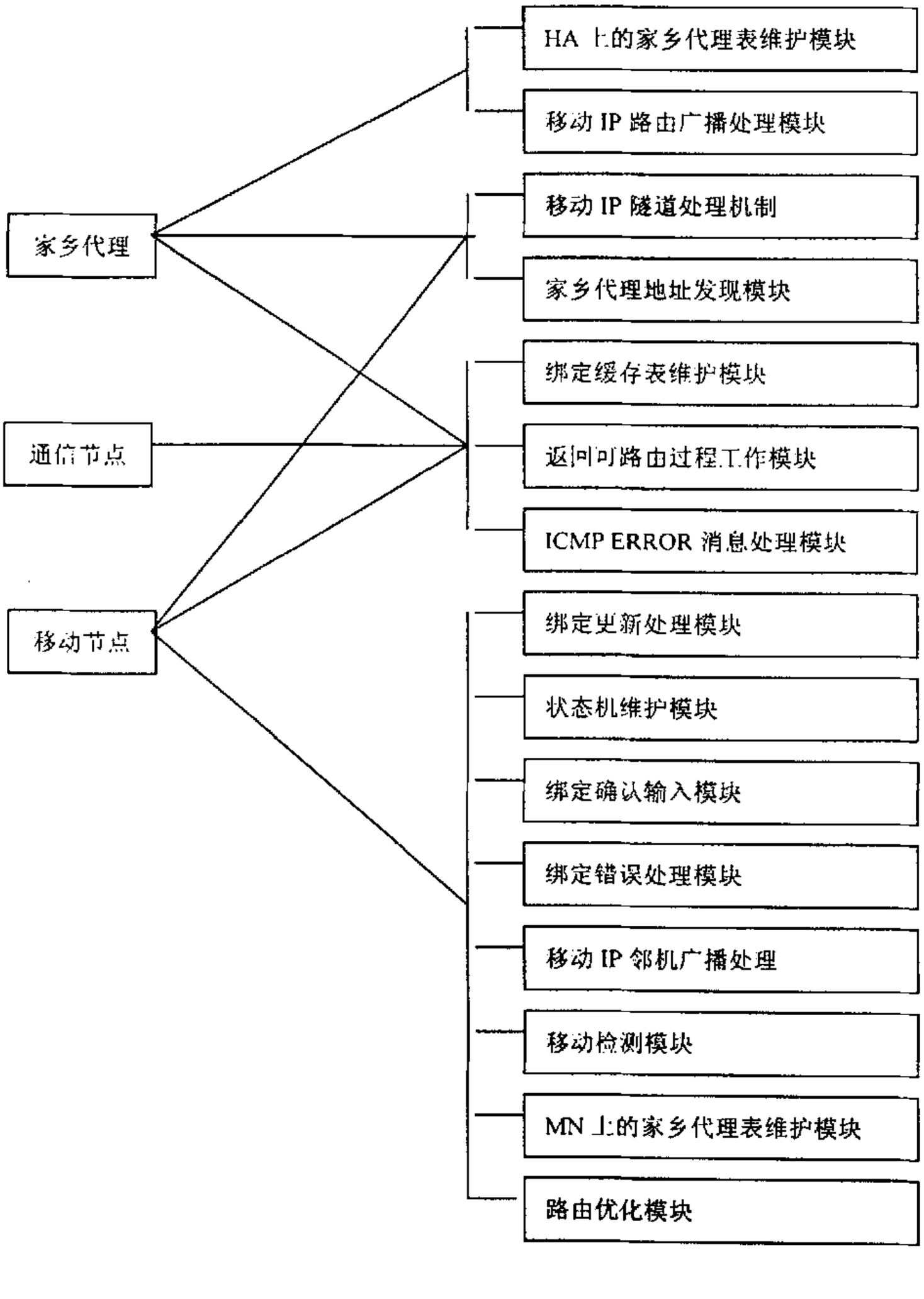


图 2-1 移动 IP 模块结构初步考虑

在实现移动 IPv6 的过程中，必将牵涉到大量的基本协议栈中的函数，比如 IP 报文的接收函数 ip6_input、IP 报文的发送函数 ip6_output、IP 报文的转发函数

ip6_forward 以及其它一些相关函数，这些函数都需要按照移动 IPv6 协议的要求进行修改，以满足移动 IPv6 的一些特殊要求。可以考虑利用编译开关#ifdef MIP6 和#endif 并结合其它机制，将移动性代码和基本协议代码有效地区分开来。打开编译开关 MIP6 以后，协议栈提供对移动性的支持。

2.3 移动 IPv6 协议实现的软件设计方案

在现有 IPv6 协议栈中引入移动性支持后，移动 IPv6 将作为整个 IPv6 协议栈的一个模块而存在，并与 IPv6 协议栈其它各个模块有机的融合在一起，共同完成 IPv6 数据报的转发工作。它在整个系统中的位置如图 2-2 所示：

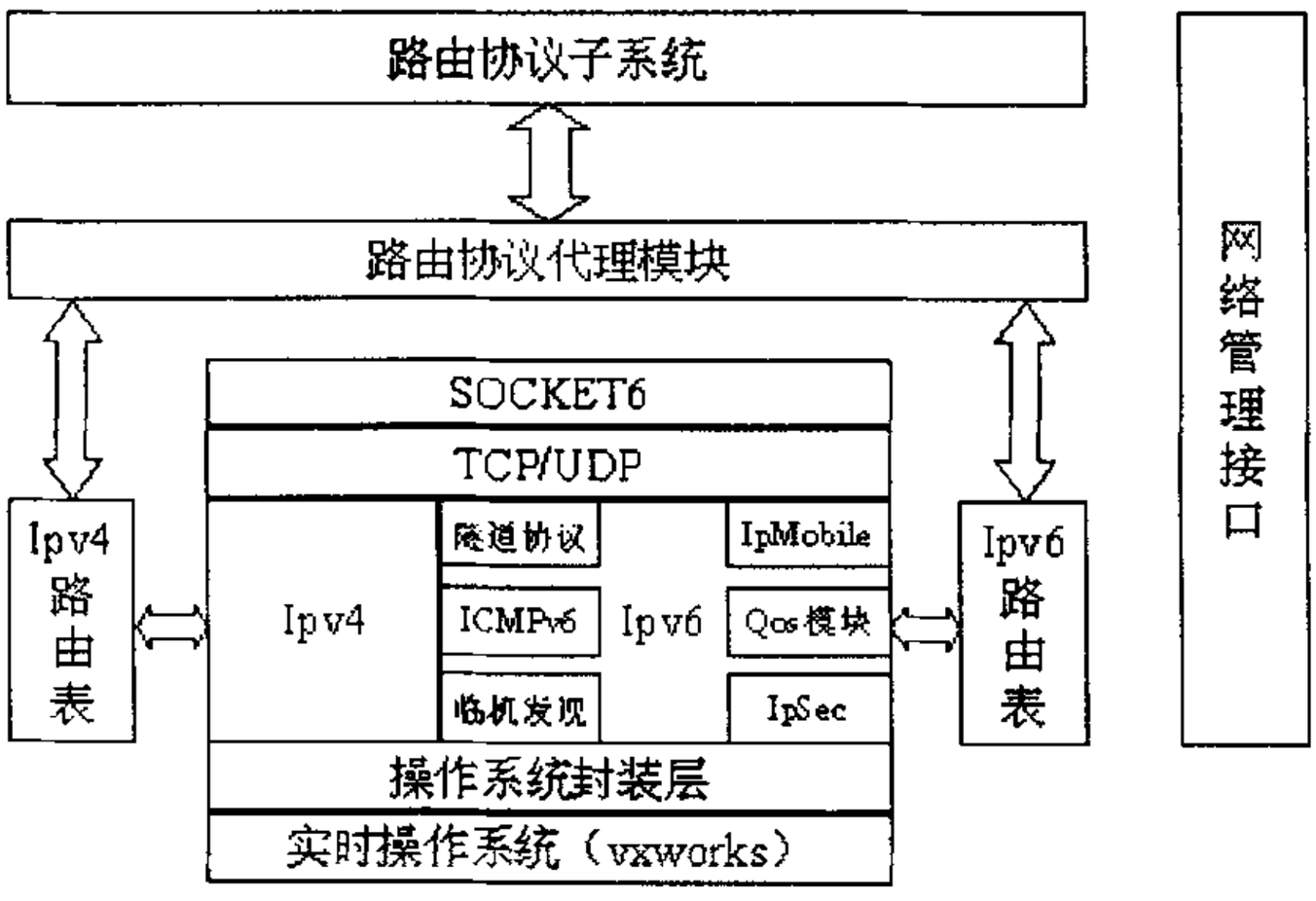


图 2-2 移动 IPv6 模块在系统中的位置

移动 IPv6 模块利用 IP 模块、IPSec 模块、ND 模块及 ICMP 模块提供的接口和功能，在协议栈内部完整实现对移动性的支持。在提供了对移动性的支持以后，运行本协议栈的设备在无需配置的情况下就具有支持路由优化的通信节点的功能。在协议栈运行起来后，操作人员能通过管理命令把运行本协议栈的设备配置成家乡代理或移动节点，实现家乡代理或移动节点的功能，同时该设备仍具有支持路由优化的通信节点的功能。

在前文对移动 IPv6 进行分析的基础上，结合图 2-1 的分析，并考虑到具体实现过程中可能遇到的实际问题，进一步将整个模块整合成为九个子模块，这些子模块协同工作，完成移动功能。移动 IPv6 模块的框架如图 2-3 所示：

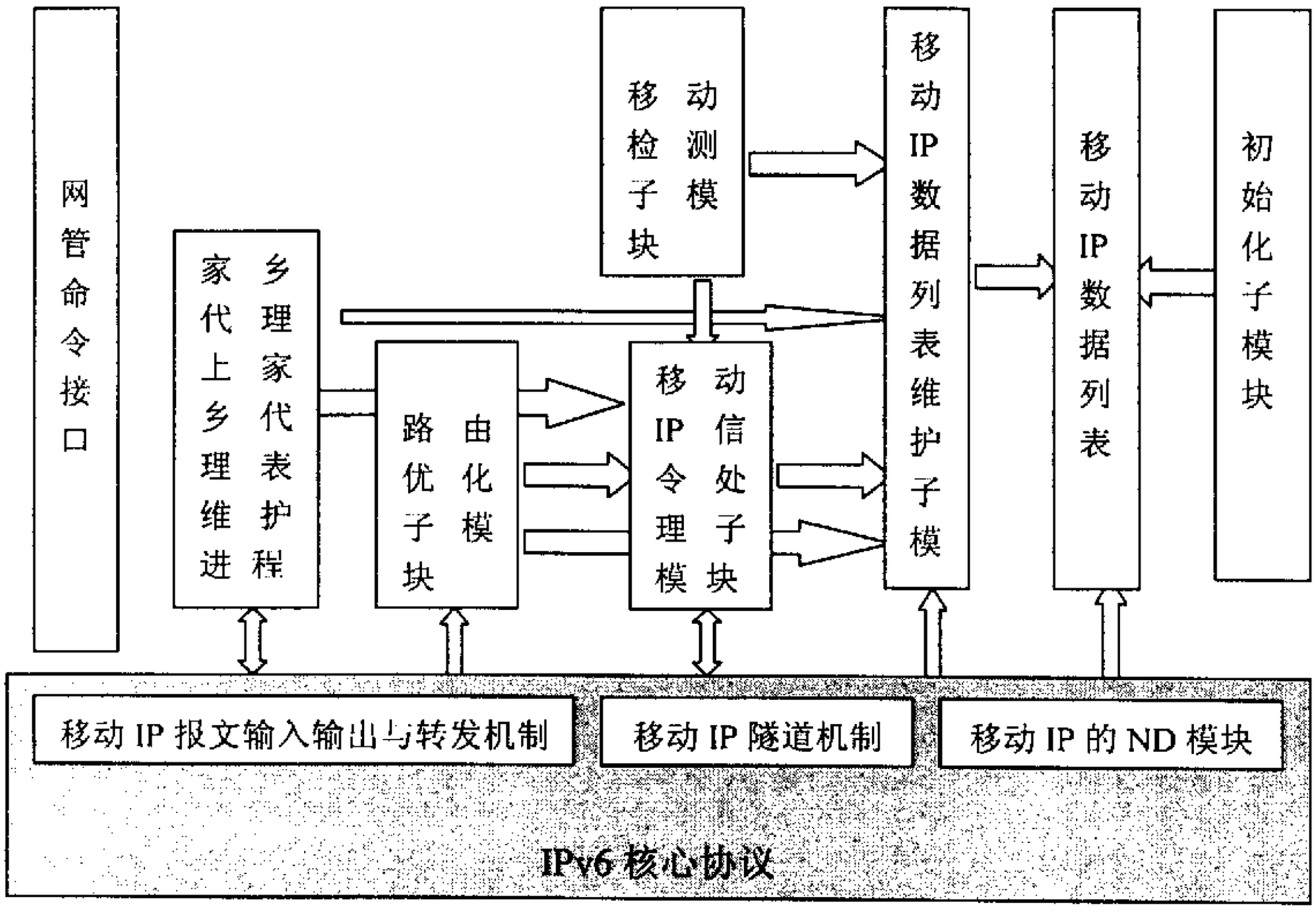


图 2-3 移动 IPv6 模块结构框架

2.3.1 初始化子模块

初始化子模块在整个协议栈进行初始化的时候调用。初始化子模块的功能是初始化配置参数和移动 IPv6 模块最主要的数据结构，包括绑定缓存列表、移动节点上家乡代理列表、移动节点上子网前缀列表、移动节点上子网列表、移动节点上绑定更新列表，另外，它还要初始化作为支持路由优化的通信节点所需的各种安全参数，并启动定时更新数组的定时器。初始化子模块的处理流程如图 2-4。初始化模块结束后，运行本协议栈的设备作为通信节点自动支持路由优化功能。

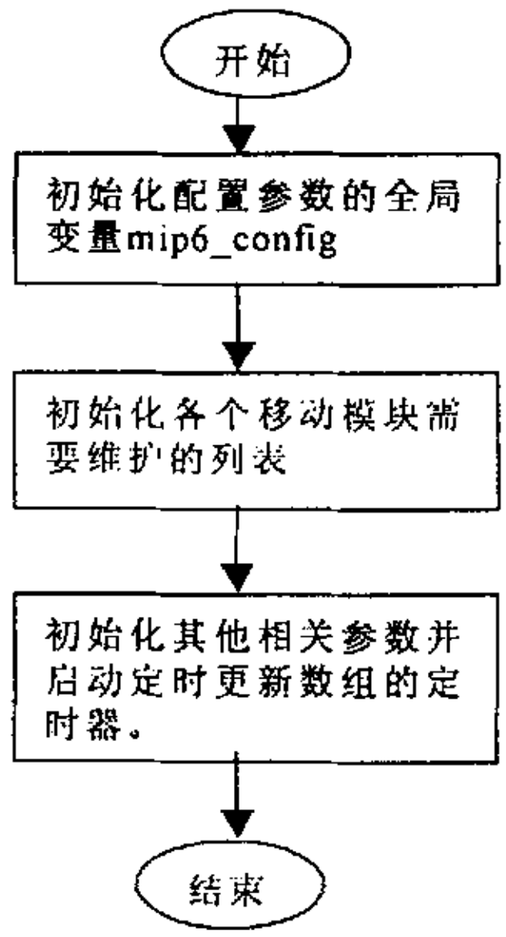


图 2-4 初始化流程

2.3.2 移动 IP 数据列表维护子模块

移动 IPv6 模块在数据包路由功能上对传统的根据路由表来处理数据包的机制的基础上增添了新的功能。这种新的功能使得节点在离开家乡网络后，其上层应用程序仍能用在家乡网络上的地址和网上的其他节点进行通信，这样移动 IPv6 对上层应用程序屏蔽了节点移动到其他子网的事实，实现了节点在移动时仍能正常通信的功能。

在协议栈中，移动 IP 功能的实现被抽象为对一系列列表的操作。这些列表包括绑定缓存列表、移动节点上家乡代理列表、移动节点上子网前缀列表、移动节点上子网列表、移动节点上绑定更新列表、家乡代理上家乡代理列表。本子模块提供对这些列表进行操作和维护的接口，供其他子模块调用。对列表的操作主要包括：新建、增加、删除、更新一个表项，根据给出关键字段查找列表，以及列表定时器维护等。这些列表都是移动 IP 模块的最基本也最重要的数据结构，它们都将作为全局变量加以实现。

- **绑定缓存列表**：绑定缓存列表的表项包含了移动节点的转交地址和家乡地址的对应关系，是移动节点通过在家乡代理和支持路由优化的通信节点上注册其转交地址之后形成的。通信节点利用绑定缓存列表向移动节点发送数据包、发送绑定刷新请求，家乡代理利用绑定缓存列表转发数据包到移动节点的转交地址，并进一步引发移动节点和通信节点的路由优化。所以绑定缓存列表

是实现移动 IP 功能最重要的数据结构。

- **移动节点上家乡代理列表**：为了保存从家乡代理通过各种可能机制学习到的在家乡网络上具有家乡代理功能的路由器的信息，移动节点需要维护一个家乡代理列表。在移动节点要向家乡网络注册一个新的家乡地址时，通过查找家乡代理列表，找到最优的家乡代理，向其进行注册。
- **移动节点上子网前缀列表**：为保存从家乡代理通过动态前缀发现机制学习到的在家乡网络上的前缀信息，移动节点要在其上维护一个前缀列表。移动节点把通过动态前缀发现机制学习到的家乡网络上的前缀及邻机发现机制学习到的外地链路上的前缀信息都保存在前缀列表中。
- **移动节点上子网列表**:移动节点上要利用通过动态发现机制学习到的路由器信息和前缀信息来组成子网表项。子网表是移动节点判断是否在家乡网络、形成家乡地址、向家乡代理注册、进行移动性检测等一系列功能都要用到的重要数据结构。
- **移动节点上绑定更新列表**：移动节点上要对每个虚拟家乡接口维护一个绑定更新列表，用于记录移动节点已经或将要向家乡代理或通信节点发送的绑定更新。移动节点向家乡代理注册、向通信节点注册、处理绑定应答、绑定更新请求时都要查找绑定更新列表。
- **家乡代理上家乡代理列表**：为支持移动节点的家乡代理动态发现机制，协议栈在家乡代理上要维护一个家乡代理列表。移动节点的家乡代理要收集移动节点的家乡链路上的具有家乡代理功能的路由器的路由通告，生成移动节点的家乡链路上的家乡代理列表。当收到移动节点的家乡代理请求时，作出应答，把家乡代理列表中的表项发送给移动节点，帮助其实现家乡代理的动态发现机制。
- **移动节点上的 hif 接口信息**：略。

2.3.3 移动 IP 信令处理子模块

在原有 IPv6 协议的基础上，移动 IPv6 新定义了几个信令，节点之间通过发送和处理这些信令来维护 2.3.2 中提到的列表，实现移动 IPv6 的功能。

2.3.3.1 绑定刷新请求消息的发送/接收

通信节点上维护一个绑定缓存表，表项记录了与该通信节点利用路由优化机

制进行通信的移动节点的家乡地址和外地转交地址的对应关系。绑定缓存表项有一个有效时间，当有效时间到时，通信节点要将其删除，等待移动节点下一次发送来绑定更新信令，再次建立绑定缓存表项。为提高通信的效率，当表项将要过期而通信节点还有和移动节点进行通信的需求时，通信节点主动向移动节点发送一个绑定刷新请求消息，移动节点收到该消息后，向通信节点发送一个绑定更新消息作为回应。通信节点收到该绑定更新消息后，刷新将要过期的绑定更新表项，这样通信节点可以利用该表项继续通过路由优化路经和移动节点进行通信。

2.3.3.2 RRP 过程

移动 IPv6 的引入对网络的安全性也相应地带来了隐患，使网络上的节点更容易受到很隐蔽的攻击。为消除、减弱这些安全隐患，移动 IPv6 在各节点交互移动信令尤其是绑定更新信令时提供了相应的安全机制，保护移动信令的完整性、秘密性。移动节点向家乡代理发送的绑定更新消息有 IPSec 来保护。移动节点向通信节点发送绑定更新消息时引入了 RRP 机制来保护该消息。RRP 过程由移动节点发起，通过移动节点和通信节点的信令交互，最终在移动节点生成一个通信节点能识别的密码，移动节点用该密码来加密绑定更新消息，通信节点用该密码来认证绑定更新消息的合法性。RRP 具体的描述参见移动 IPv6 的标准草案。协议栈中对 RRP 的实现可以设计如下。

1. 移动节点发送 hoti、coti 消息

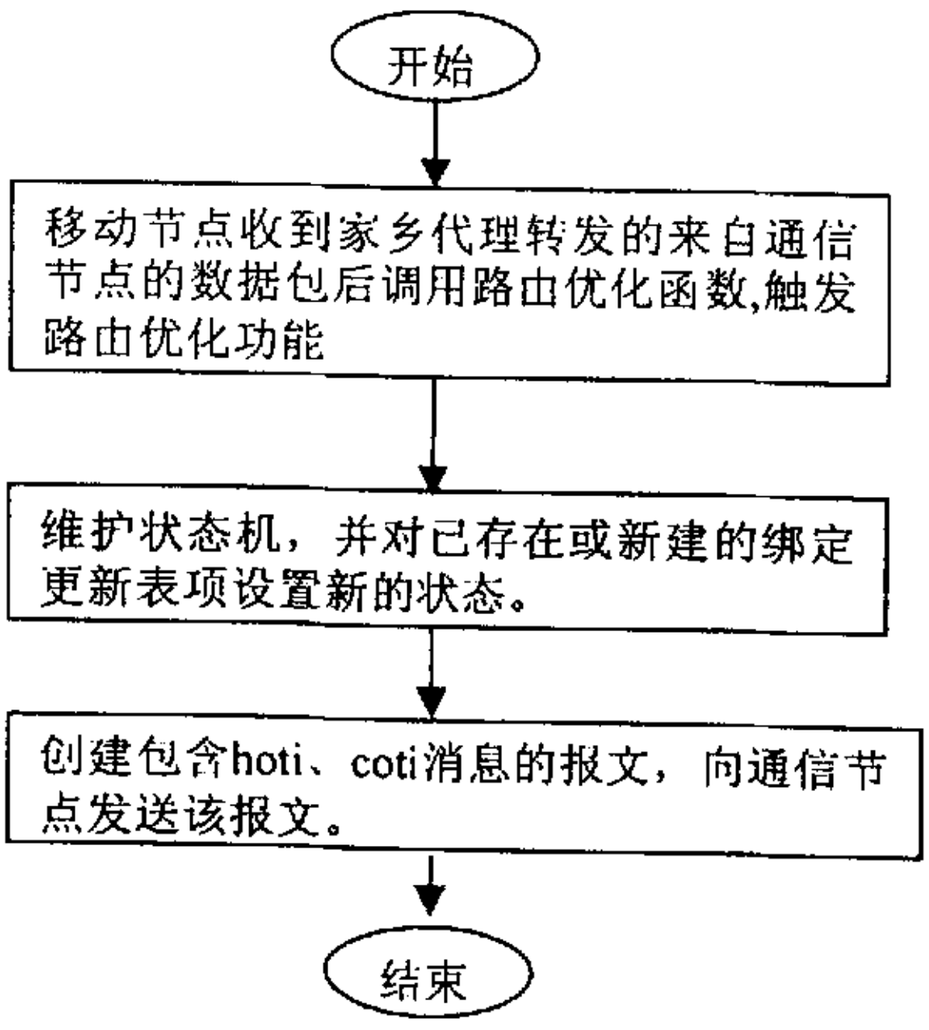


图 2-5 移动节点发送 hoti、coti 消息

2. 通信节点接收到 hoti、coti 消息，发送 hot、cot 消息

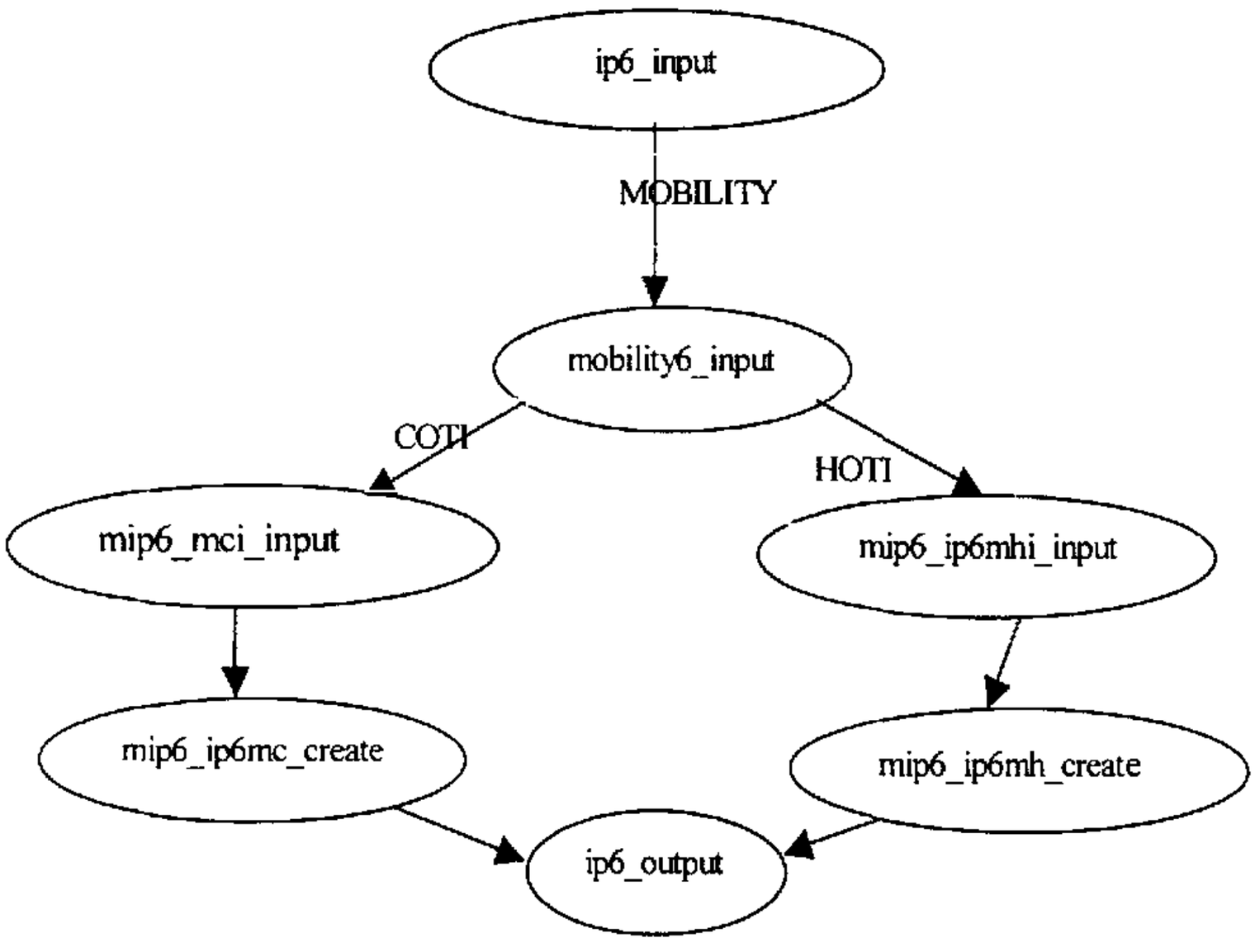


图 2-6 通信节点接收到 hoti、coti 消息处理流

3. 移动节点接收到 hot、cot 消息

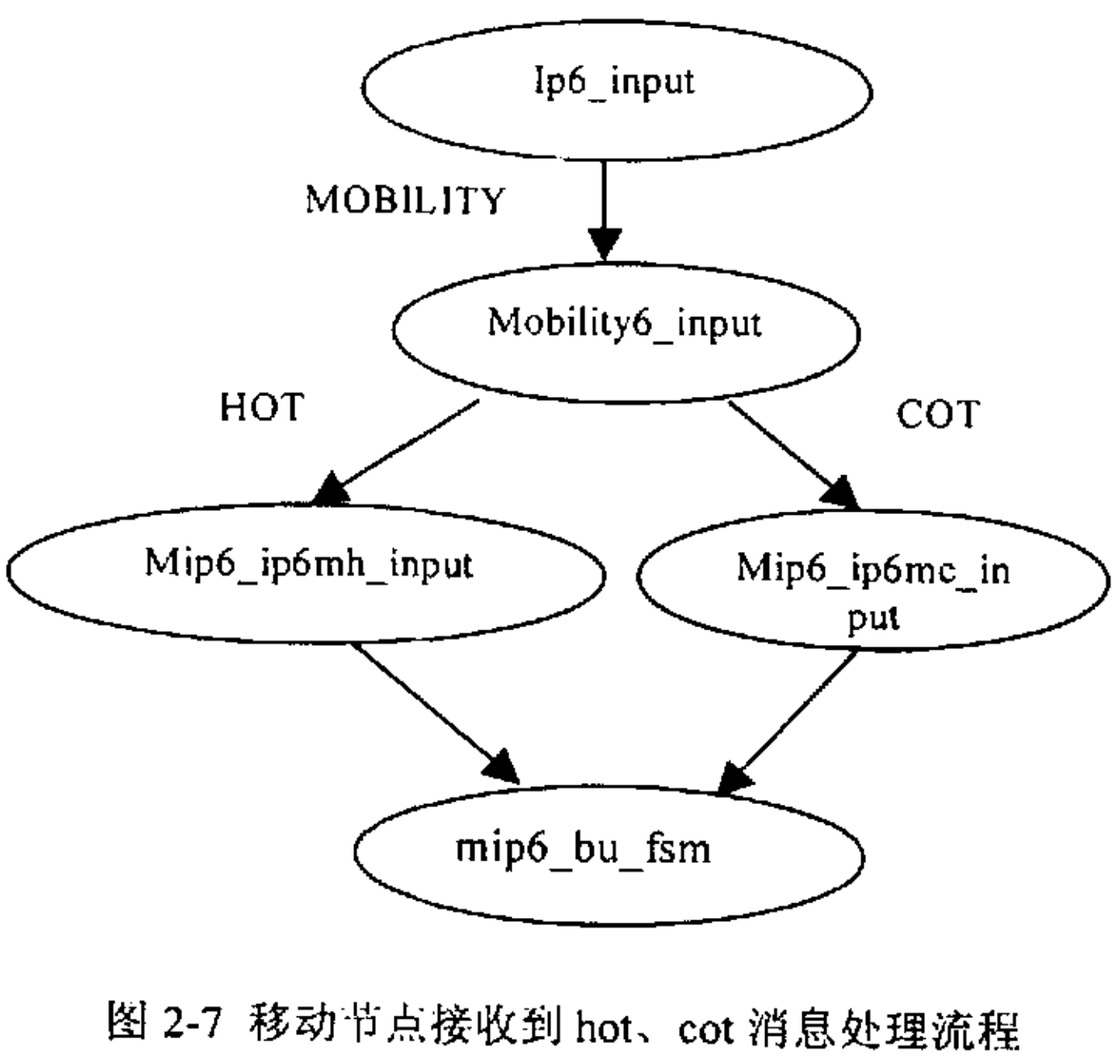


图 2-7 移动节点接收到 hot、cot 消息处理流程

2.3.3.3 绑定更新消息的发送/接收

当移动在外地得到新的转交地址、或移动节点从外地回到了家乡链路，节点要向家乡代理和通信节点发送绑定更新消息，注册新的转交地址或注销绑定。协

议栈中对绑定更新消息发送/接收子模块的实现如下：

1. 移动节点向家乡代理发送绑定更新

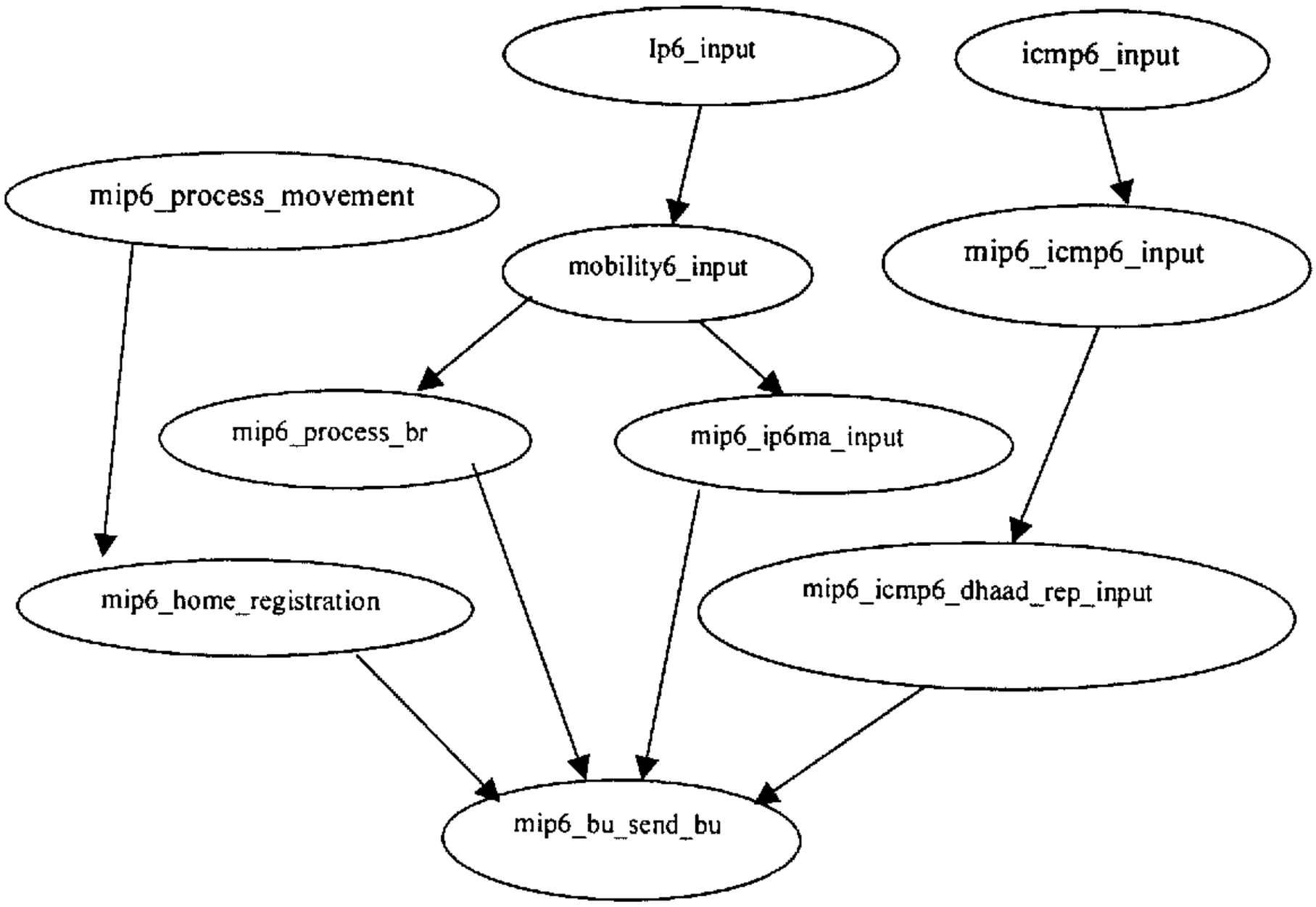


图 2-8 移动节点向家乡代理发送绑定更新

移动节点向家乡代理发送绑定更新被以下情况触发：

a) 移动节点检测到自己的移动后，调用函数 mip6_home_registration 发送绑定更新。

b) 移动节点收到绑定应答后，调用函数 mip6_ip6ma_input 发送绑定更新。

c) 移动节点收到绑定请求后，调用函数 mip6_process_br 发送绑定更新。移动节点收到家乡代理请求的应答后，调用函数 mip6_icmp6_dhaad_rep_input 发送绑定更新。

2. 移动节点向通信节点发送绑定更新

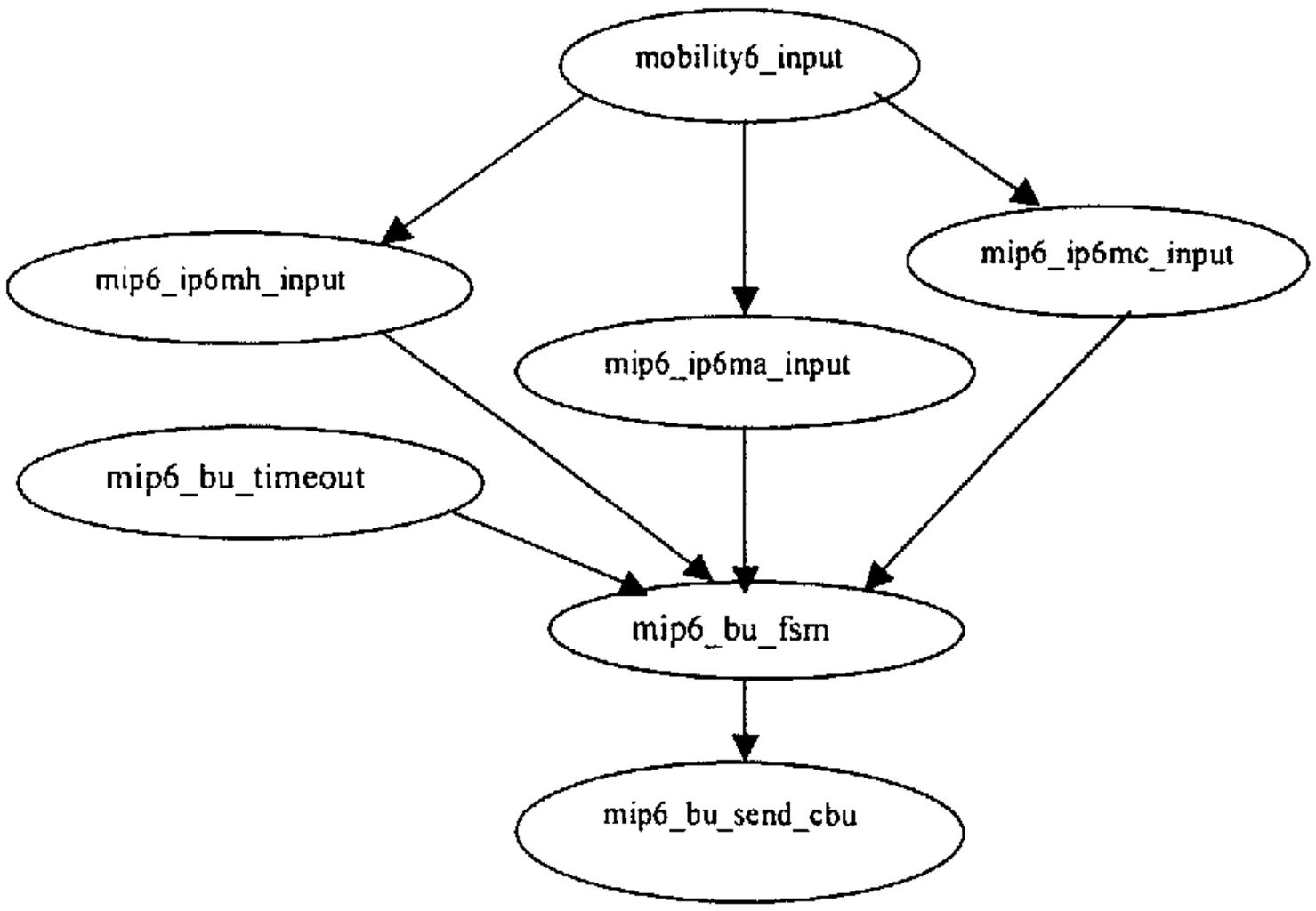


图 2-9 移动节点向通信节点发送绑定

移动节点向通信节点发送绑定更新被以下情况可能被触发：

a) 移动节点定时器检测发送绑定更新表时；

b) 移动节点收到 HoT 消息后；

c) 移动节点收到 CoT 消息后；

d) 移动节点收到绑定应答后。

3. 家乡代理接收到绑定更新

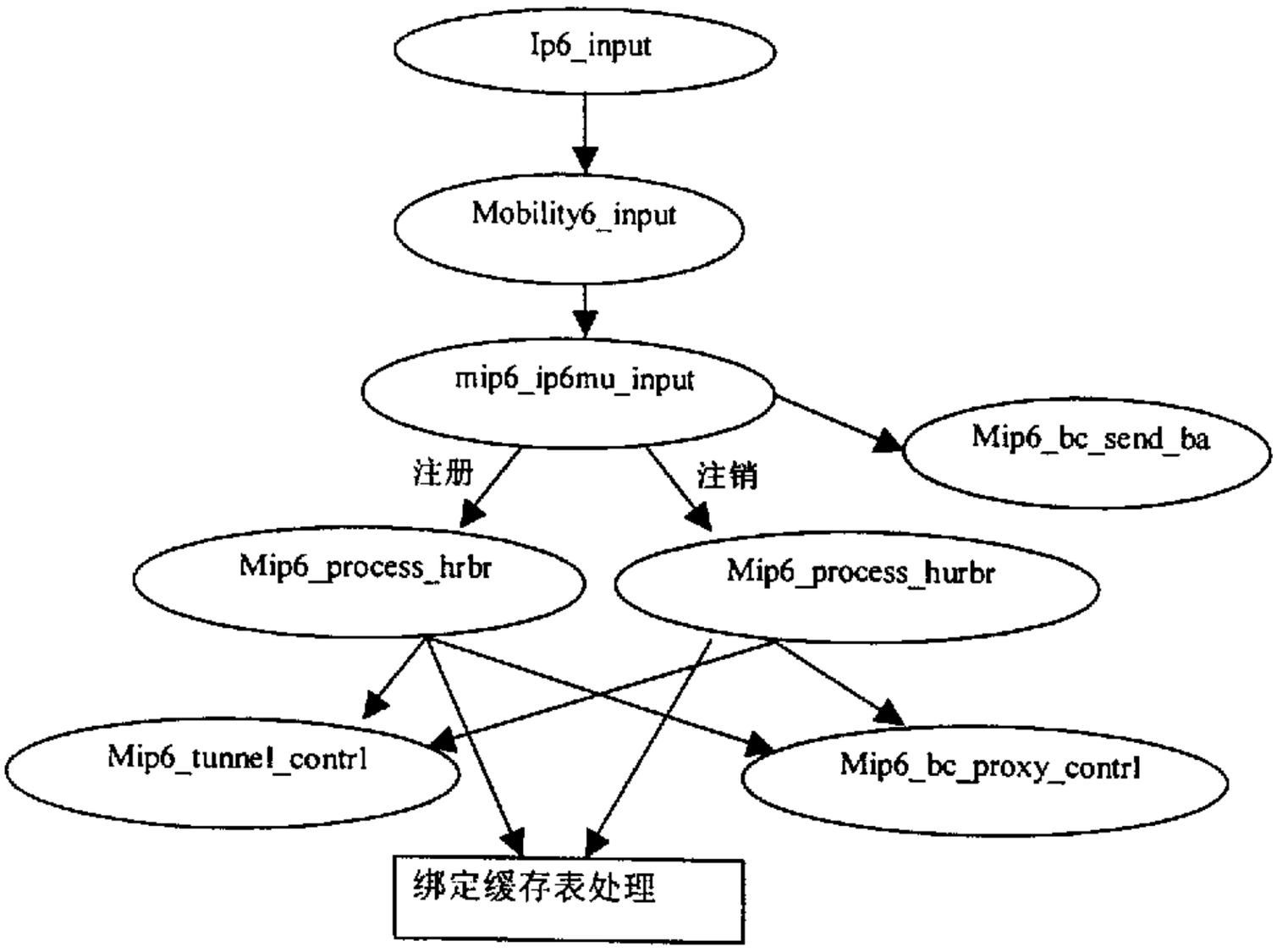


图 2-10 家乡代理接收到绑定更新

家乡代理接收到绑定更新后，调用函数 Mip6_bc_send_ba 发出应答；调用函数 mip6_process_hrbu 和函数 mip6_process_hurbu，结合绑定更新报文的要求和绑定缓存表的现状，调用绑定缓存表处理函数来添加、更新或删除相应绑定缓存项；调用函数 mip6_tunnel_contrl 来建立、更新或删除一个隧道；或者调用函数 mip6_bc_proxy_contrl 来启动、停止家乡代理代替移动节点发送邻机广播报文的工作。

4. 通信节点接收到绑定更新

通信节点接收到绑定更新后，调函数 mip6_bc_send_ba 向分组源地址返回一条 BA 报文，以及对应绑定缓存项是否已存在于绑定缓存表（体现通信节点绑定缓存项的现状），在绑定缓存表增加、更新或删除一条绑定缓存项。它们分别可以调用绑定缓存维护模块中的 mip6_bc_register 函数、mip6_bc_update 函数或 mip6_bc_delete 来完成对绑定缓存表的维护。

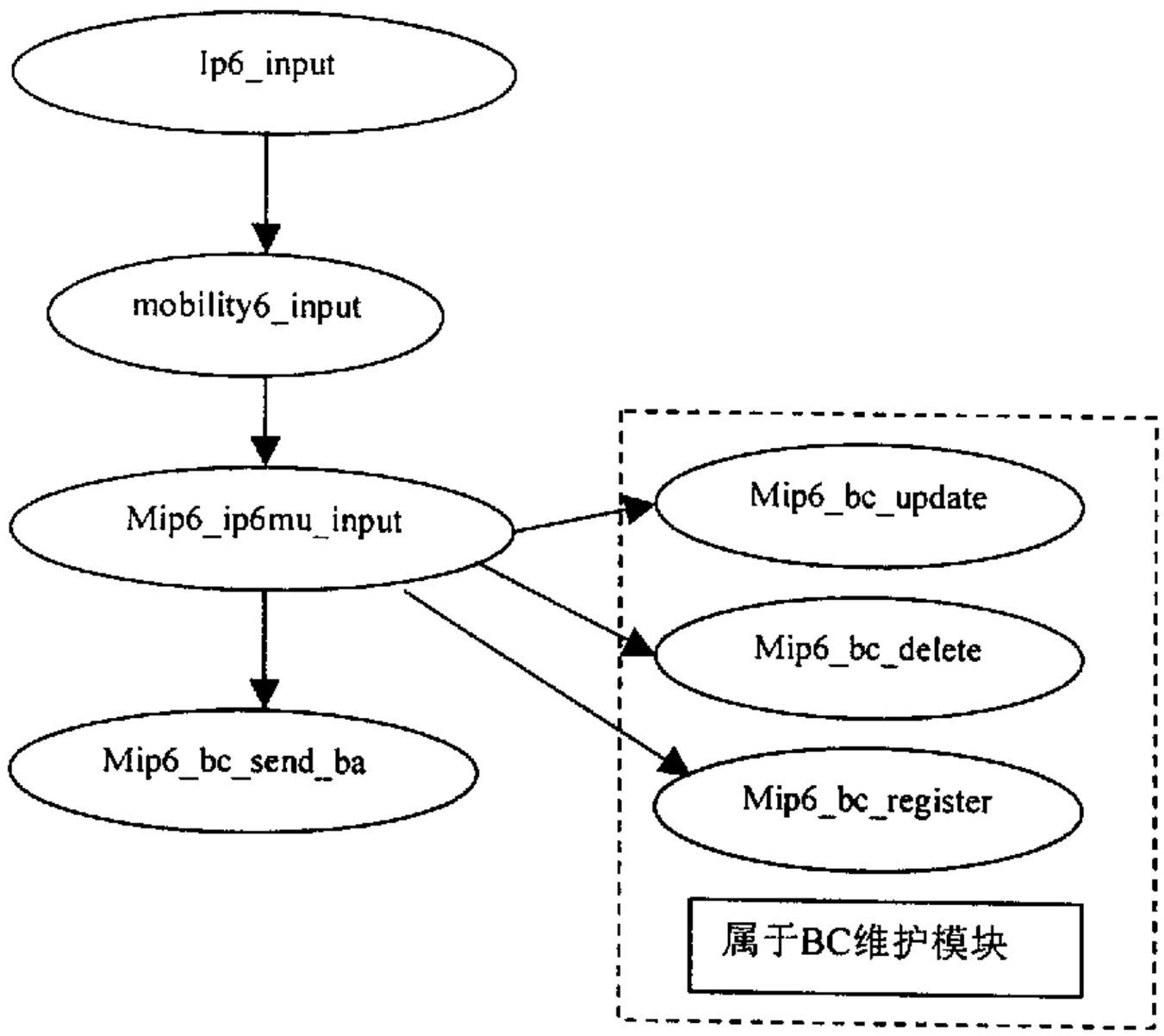


图 2-11 通信节点接收到绑定更新

2.3.3.4 绑定确认消息的发送/接收

通信节点和家乡代理发送绑定确认消息的过程见上小节中的通信节点接收到绑定更新、家乡代理接收到绑定更新。

移动节点接收绑定确认消息

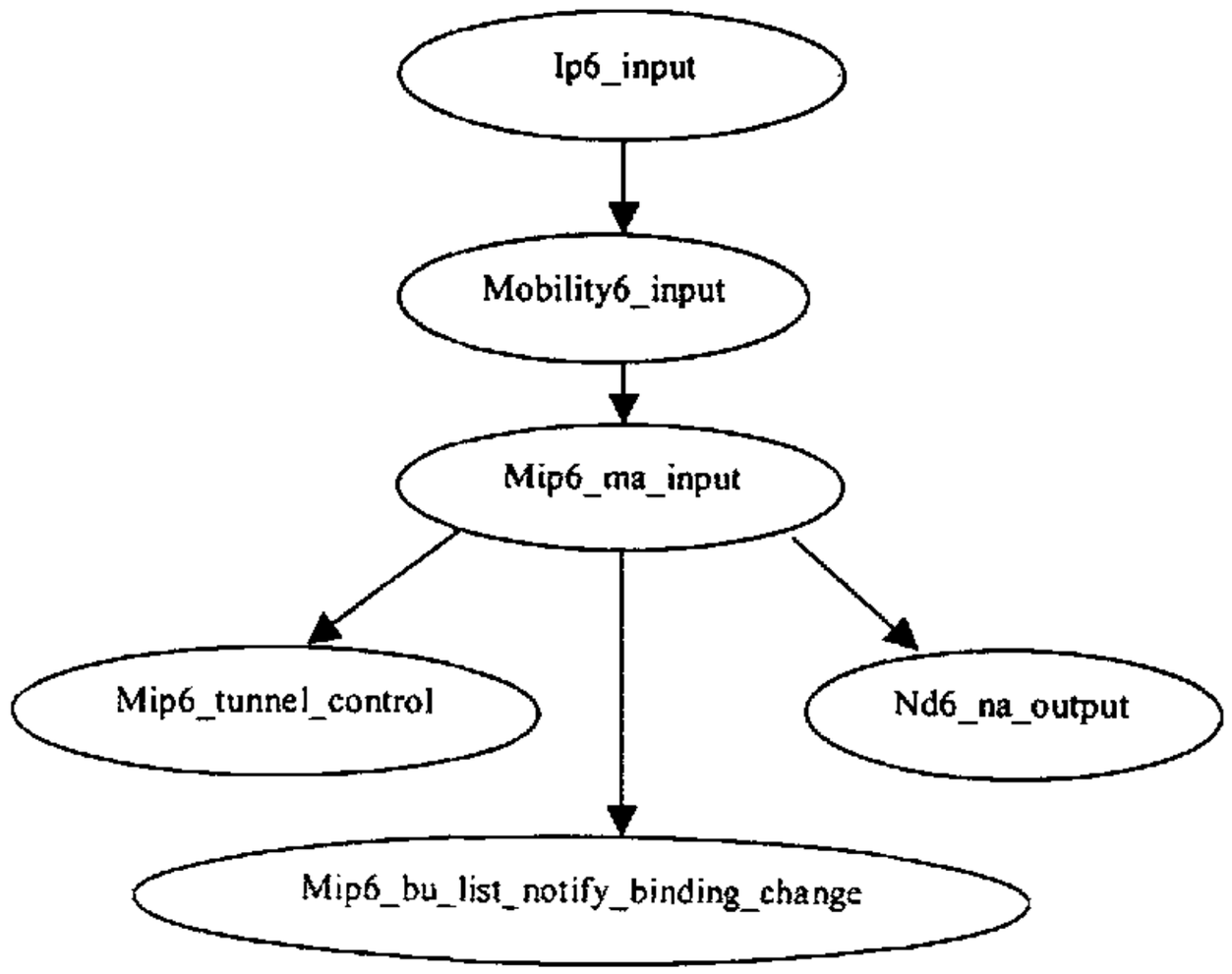


图 2-12 移动节点绑定确认消息接收子模块处理流程

移动节点接收绑定确认消息后，调函数 mip6_ip6ma_input 处理该绑定确认报文。函数 mip6_ip6ma_input 要读取自身参数判断自身状态，并结合绑定确认报文的标志设置情况，如果确认移动节点的转交地址已经发生变化（包括回到家乡链路），可以选择调用下面几个函数提供的功能，根据具体情况完成处理。mip6_bu_list_notify_binding_change 用于通知绑定更新表中所有对端节点本移动节点的转交地址已经发生变化，nd6_na_output 作邻机广播，调 mip6_tunnel_control 完成对相关隧道的配置更改。

2.3.3.5 绑定错误消息的发送/接收

通信节点绑定错误消息发送处理过程如图 2-13。

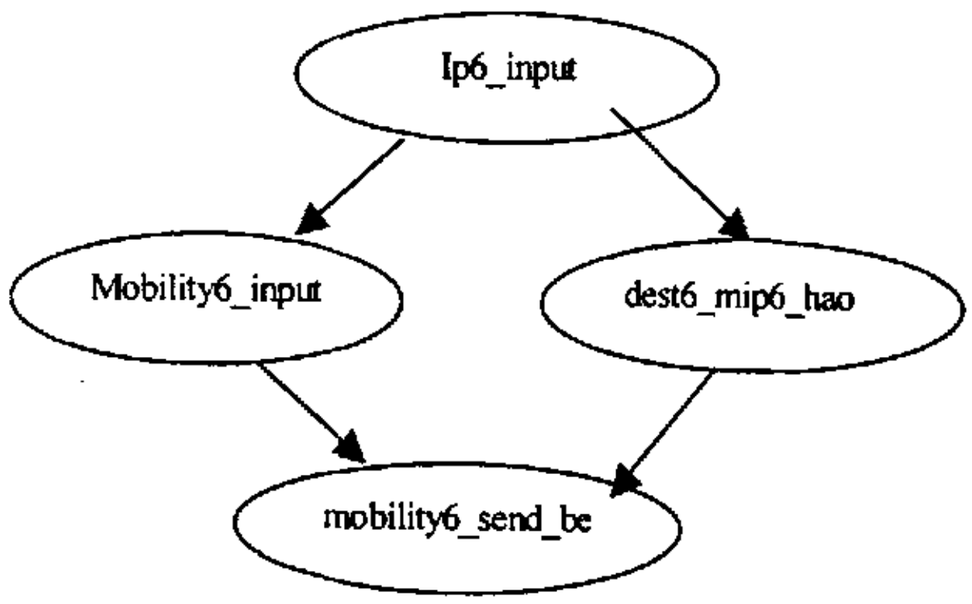


图 2-13 通信节点绑定错误消息发送处理流程图

通信节点绑定错误消息发送被以下情况触发：

1）通信节点收到不能识别的移动头消息时，调用函数 mobility6_send_be 发送绑

定错误消息。

2）通信节点收到带有 HOA 的目的头，而绑定缓存表中没有关于该家乡地址的绑定表项时，调用函数 mobility6_send_be 发送绑定错误消息。

移动节点绑定错误消息接收处理如图 2-14。

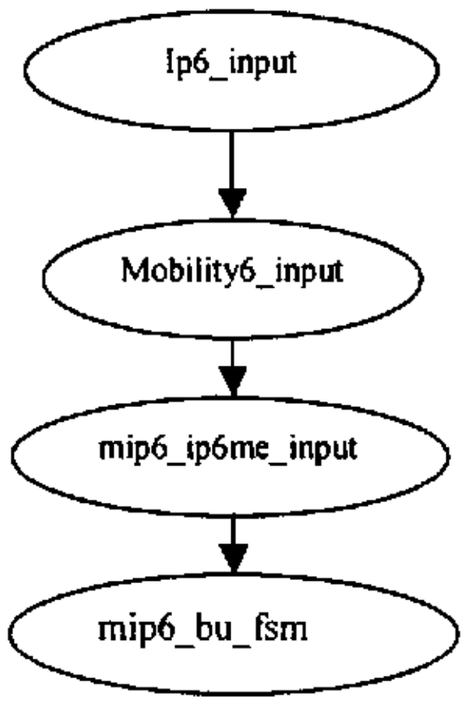


图 2.14 通信节点绑定错误消息接收处理流程图

移动节点接收到绑定错误消息后，调 mip6_ip6me_input 来处理该绑定错误报文, mip6_ip6me_input 调用 mip6_bu_fsm 实现对状态机的改变。

2.3.3.6 家乡代理地址发现机制

家乡代理上家乡代理列表的学习过程如果 2-15

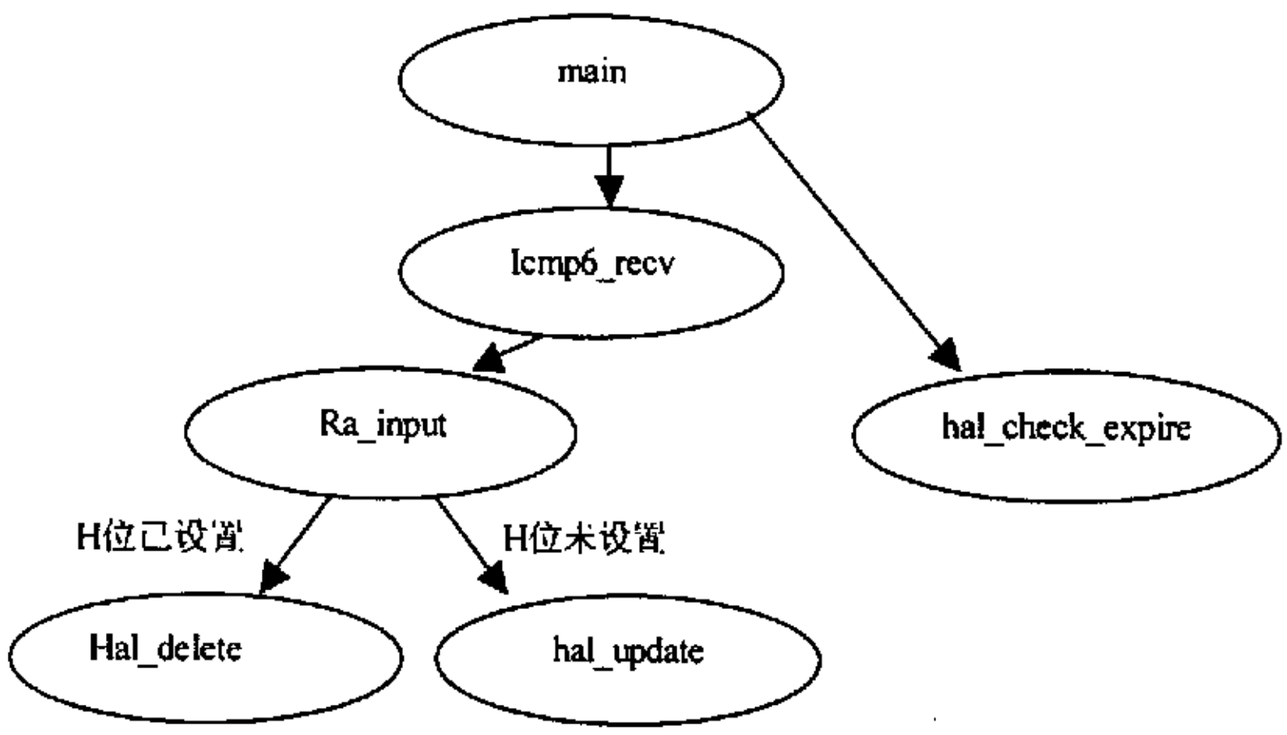


图 2-15 家乡代理上家乡代理列表的学习

移动 IP 模块为家乡代理设计一个专门程序来处理与家乡代理表有关的一些工作，包括监听其它路由器发送的路由广播并利用路由广播来更新家乡代理表；接收家乡代理地址发现请求消息并返回一条家乡代理地址发现应答消息等处理在此完成。完成两个工作：检查家乡代理表中的家乡代理是否过期，如果过期则

删除该家乡代理；通过 select 系统调用等相关工作，获得相关 ICMP 包，调函数专用 ICMP 处理函数接收并处理。

2.3.3.7 移动节点动态发现家乡代理机制

移动节点发送家乡代理请求过程如图 2-16。

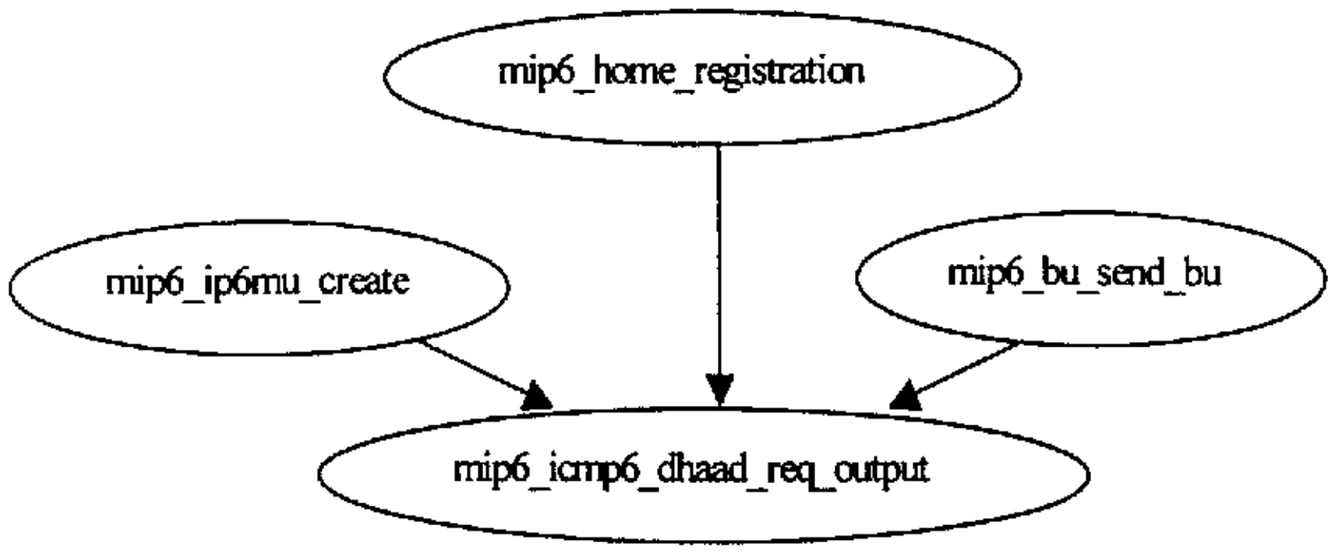


图 2-16 移动节点发送家乡代理请求

移动节点发送家乡代理请求被以下情况触发：

a) 移动节点检测到自己移动后，调用函数 mip6_home_registration 向家乡代理注册，但没有可用的家乡代理时，调用函数 mip6_icmp6_dhaad_req_output 发送家乡代理请求。

b) 移动节点建立一个新的向家乡代理注册的绑定更新时，若没有可用的家乡代理时，调用函数 mip6_icmp6_dhaad_req_output 发送家乡代理请求。

c) 移动节点向家乡代理发送绑定更新时，若需发送的绑定更新项没有家乡代理的地址时，调用函数 mip6_icmp6_dhaad_req_output 发送家乡代理请求。

家乡代理接收到家乡代理请求过程如图 2-17。

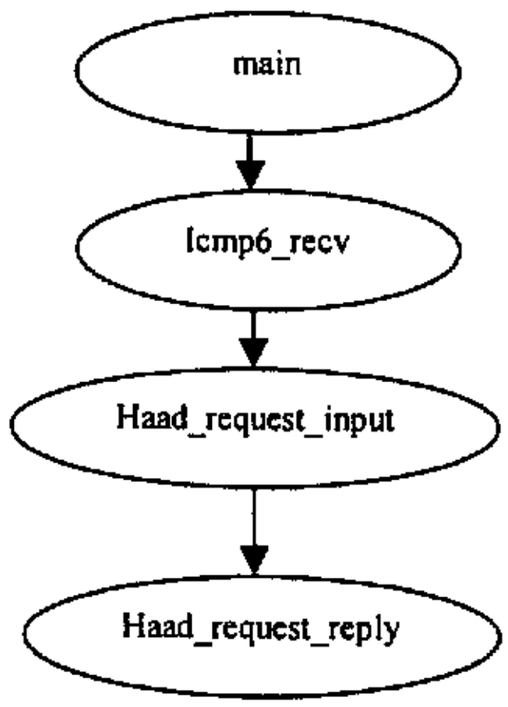


图 2-17 移动节点发送家乡代理请求

家乡代理接收到家乡代理请求后，调用 Haad_request_reply 发送一个应答报

文。

移动节点接收到家乡代理请求的应答

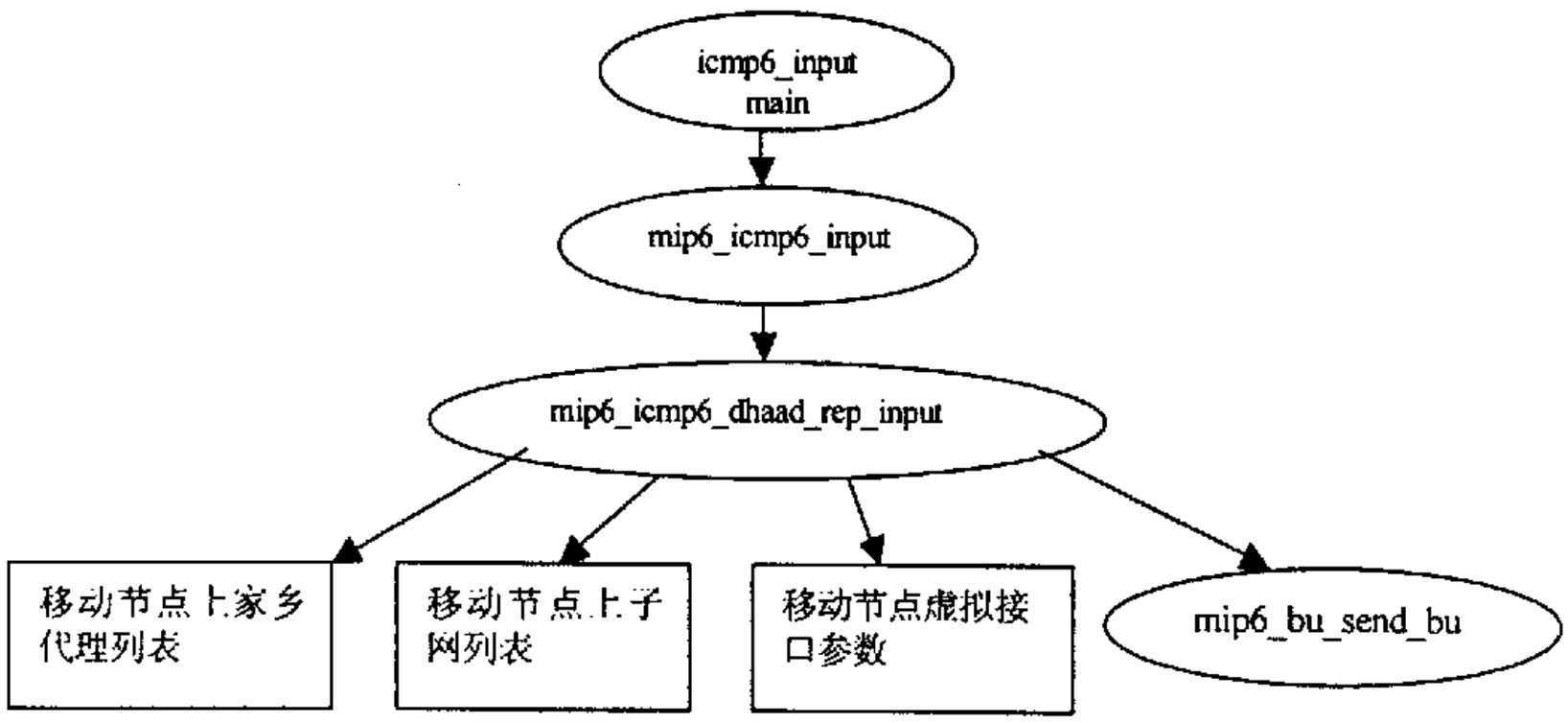


图 2-18 移动节点接收到家乡代理请求的应答

移动节点接收到家乡代理请求的应答后，对移动节点上家乡代理列表、移动节点上子网列表、移动节点虚拟接口参数等数据结构进行维护，并调用函数 mip6_bu_send_bu 把那些在等待家乡代理应答的绑定更新项发送出去。

2.3.3.8 移动前缀发现机制

该功能没有实现，但预留接口。

2.3.3.9 状态机

移动节点的状态机工作主要牵涉到如下内容。

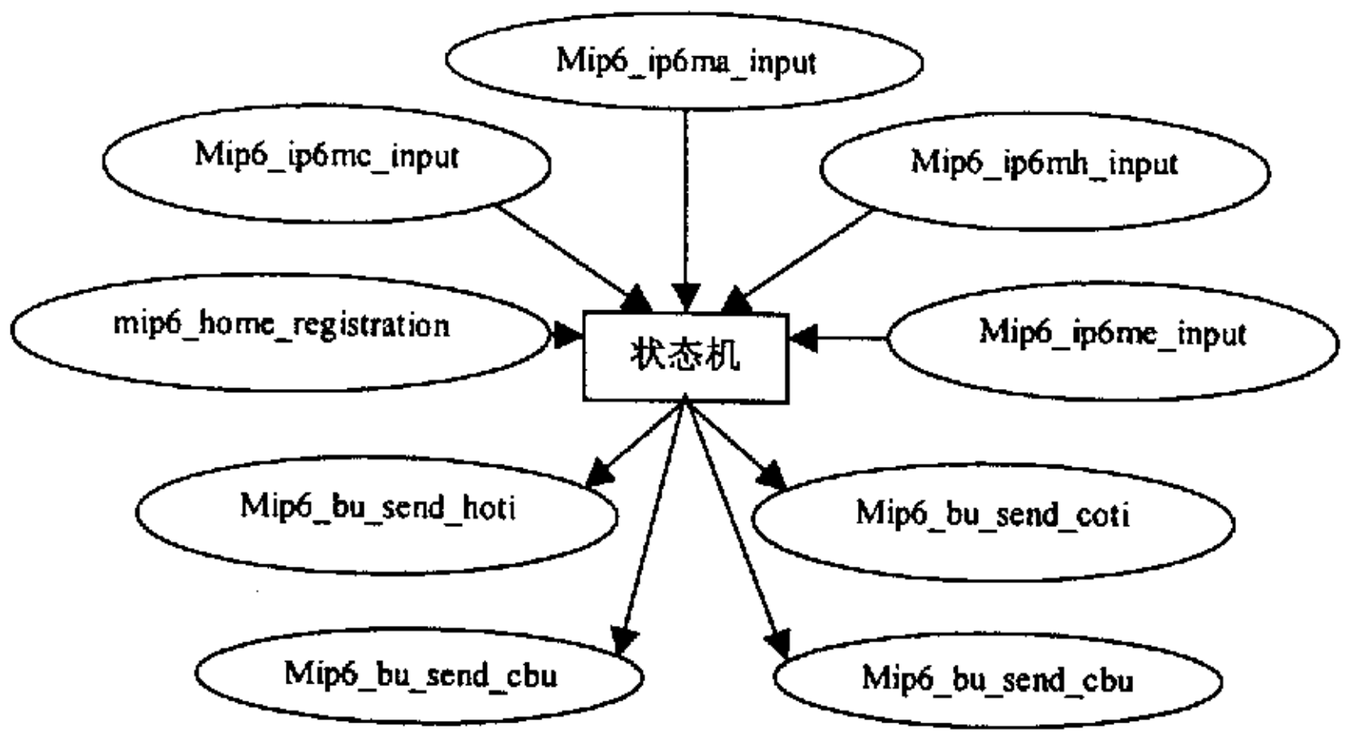


图 2-19 移动节点状态机简图

2.3.4 移动检测子模块

移动节点的移动性检测功能被以下情况触发：

a) 当网管命令把节点配置为移动节点时，网管命令模块调用函数 mip6_ioctl，进

行移动性检测，以判断节点现在所处的位置。

b) 移动节点收到所在链路上的路由器发送来的路由通告后，调用 Nd6_ra_input 对该通告进行处理，Nd6_ra_input 先调用 mip6_prefix_list_update 对节点上的数据结构进行更新，再调用 Pfx_onlink_check 触发移动性检测。

c) in6_control 模块触发节点进行移动性检测。

d) 移动性检测功能主要由函数 mip6_process_movement 来完成。该函数首先用 mip6_select_coa2 选择一个转交地址，再调用 mip6_register_current_location 判断当前节点在家乡还是在外地，转交地址是否发生了变化，移动节点最后根据这些参数调用 mip6_process_pfxlist_status_change 向家乡代理注册或注销。

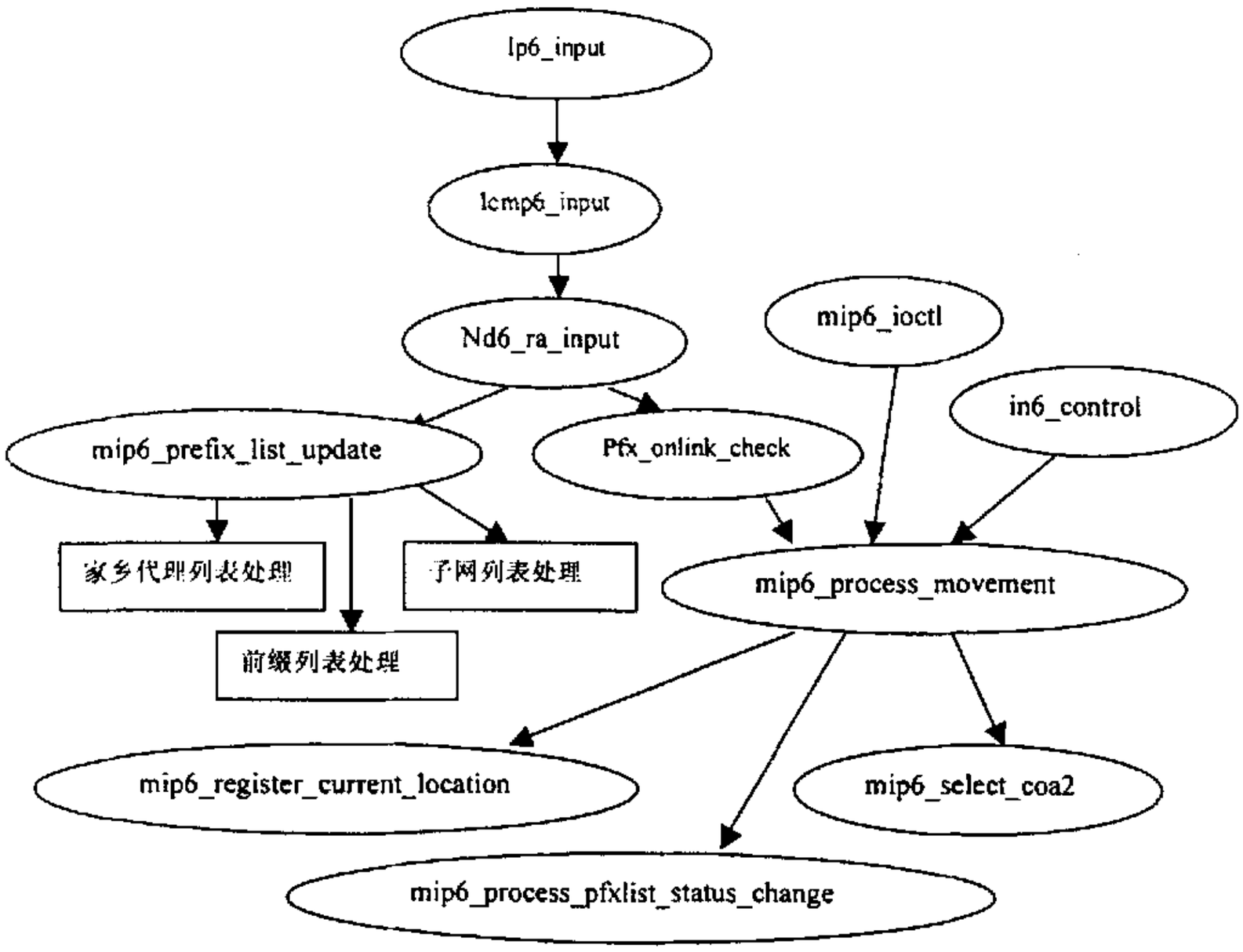


图 2-20 移动检测模块

2.3.5 路由优化子模块

移动节点的路由优化功能被以下情况触发：

当移动节点收到家乡代理通过隧道转发来的通信节点的数据包时，Ip6_input 函数会调用 mip6_route_optimize 触发移动节点的路由优化功能。

若节点上没有关于该通信节点的绑定更新表项，mip6_route_optimize 创建一

个新的表项，并将其插入绑定更新表中。然后 mip6_route_optimize 调用 mip6_bu_fsm，对绑定更新表项的状态进行修改，触发新的 RR 过程。

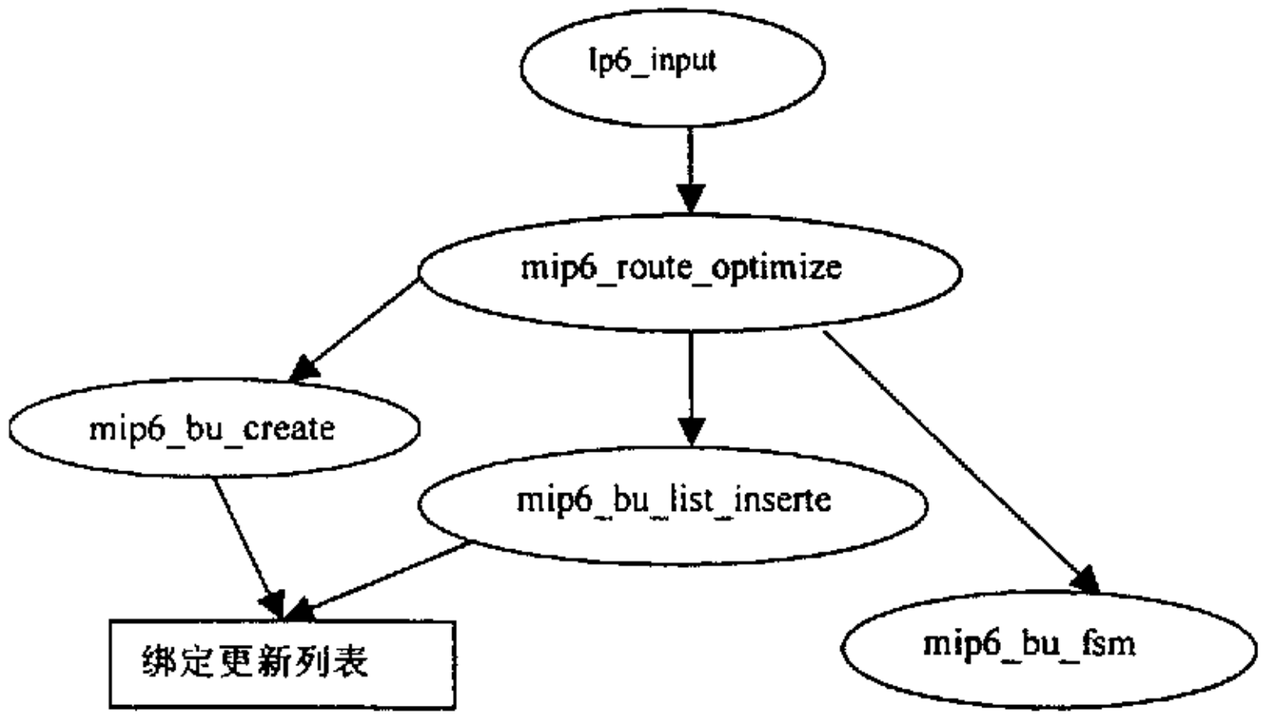


图 2-21 路由优化模块

2.3.6 移动 IP 邻机发现子模块

移动 IP 的工作机制是建立在邻机发现的基础之上的，需要邻机发现模块提供 DAD、proxy ND 等支持。移动 IP 还需要对邻机发现模块进行了一些修改，使其满足移动性实现的需求。移动 IP 使用几个函数对其所需要的 nd 部分的功能进行了进一步的封装，使其更加方便移动 IP 模块的实际调用。这里的移动邻机发现模块主要包括几个对 DAD 和 proxy ND 功能进行封装的函数，将它们单独列为一个模块，主要是基于实现上的考虑。这些函数比较简单和分散，这里不作详细的说明。

2.3.7 移动 IP 隧道机制

移动 IP 的封装机制主要用于移动节点和通信节点进行通信。

移动节点和通信节点进行通信有两者可能的方式。第一种方式就是双向隧道模式。另一种方式是采用路由优化。双向隧道模式对通信对端是否支持移动 IP 不作要求。当移动节点还没有向通信对端进行注册，也就是在通信对端上还没有关于移动节点的绑定信息的时候，可以采用这种方式进行通信。通信对端发往移动节点的数据报被传送给家乡代理，然后由家乡代理通过隧道转交给移动节点。移动节点发往通信对端的数据报先通过隧道（称为反向隧道）传给家乡代理，然后在由家乡代理转发给通信对端。这里的隧道就是通过封装机制来实现的。

2.3.7.1 不需要进行封装的情形

移动节点中通信对端进行通信的过程中，下面情形是不需要进行封装的。移动节点使用转交地址作为数据报源地址的时候数据报是不需要进行封装的。例如，当移动节点向通信对端发送 COT 消息的时候，数据报直接发往通信对端，而没有经过家乡代理进行转发。

如果移动节点已经向通信对端注册了它的转交地址，那么发往这个通信对端的数据报也不需要进行封装。当移动节点与通信对端通信的时候，如果待发送的数据报的目的地址已经存在于绑定更新项中，那么这个数据报的源地址必定是它的转交地址。这是因为，如果存在有关目的地址的合法绑定更新项，mip6_exthdr_create()会在分组中插入一个 HAO，然后 ip6_output()将数据报的源地址与 HAO 中的内容相交换。最后，这个数据报不会通过反向隧道发送。

通信对端与移动节点进行通信，向移动节点发送数据报的时候，通信对端会检查本机的绑定缓存中有没有数据报目的地址所对应的绑定缓存项。如果找到了对应的绑定缓存，那么就使用第二类路由扩展报头，把数据报先发往该绑定缓存中所指示的移动节点转交地址，然后交移动节点处理。在这种情形下，通信对端发往移动节点的数据报不需要把数据报发往家乡代理再进行隧道封装转交。

2.3.7.2 隧道封装过程

当移动节点使用它的家乡地址向通信对端发送数据报的时候，如果在移动节点上没有关于通信对端的可用的绑定更新，那么移动节点必须使用反向隧道；当通信对端向移动节点发送数据报的时候，如果通信对端上不存在关于移动节点的可用的绑定缓存项，那么数据报必须发往移动节点的家乡代理然后再经过隧道转发给移动节点。

隧道封装处理的基本思想是，当在移动节点和通信对端之间发送/接收包含 MIPv6 信令的消息（比如 HOT/HOTI 等）的时候，这些数据报必须进行加密。如果这些数据报不含有任何移动 IP 信令，那么这些数据报没有必要专门进行保护。

2.3.7.3 移动节点中的封装处理

在移动节点中，在数据包从上层向下传递到 ip6_ouput 后，该函数首先根据数据包的扩展选项构造各种扩展头，包括 hop-by-hop 扩展头、routing 扩展头、

destination 扩展头以及和 MIPv6 相关的各种扩展头。然后就查询安全策略，察看对数据包的处理策略。安全策略规定，如果数据报中含有移动 IP 信令（比如 HOT 等），那么这个数据报必须进行加密，因此需要进行 IPSec 处理，KAME/MIP 的 IPSec 处理主要是进行 ESP 封装。如果数据报中不含有任何移动 IP 信令，而只是通常的数据报，那么这些数据报没有必要专门进行保护，也就是说，不需要专门安排 ESP 封装处理。这里，IPSec 提供的查询函数为 IPSec6_getpolicybyaddr()，提供的隧道处理函数为 IPSec6_output_tunnel()。

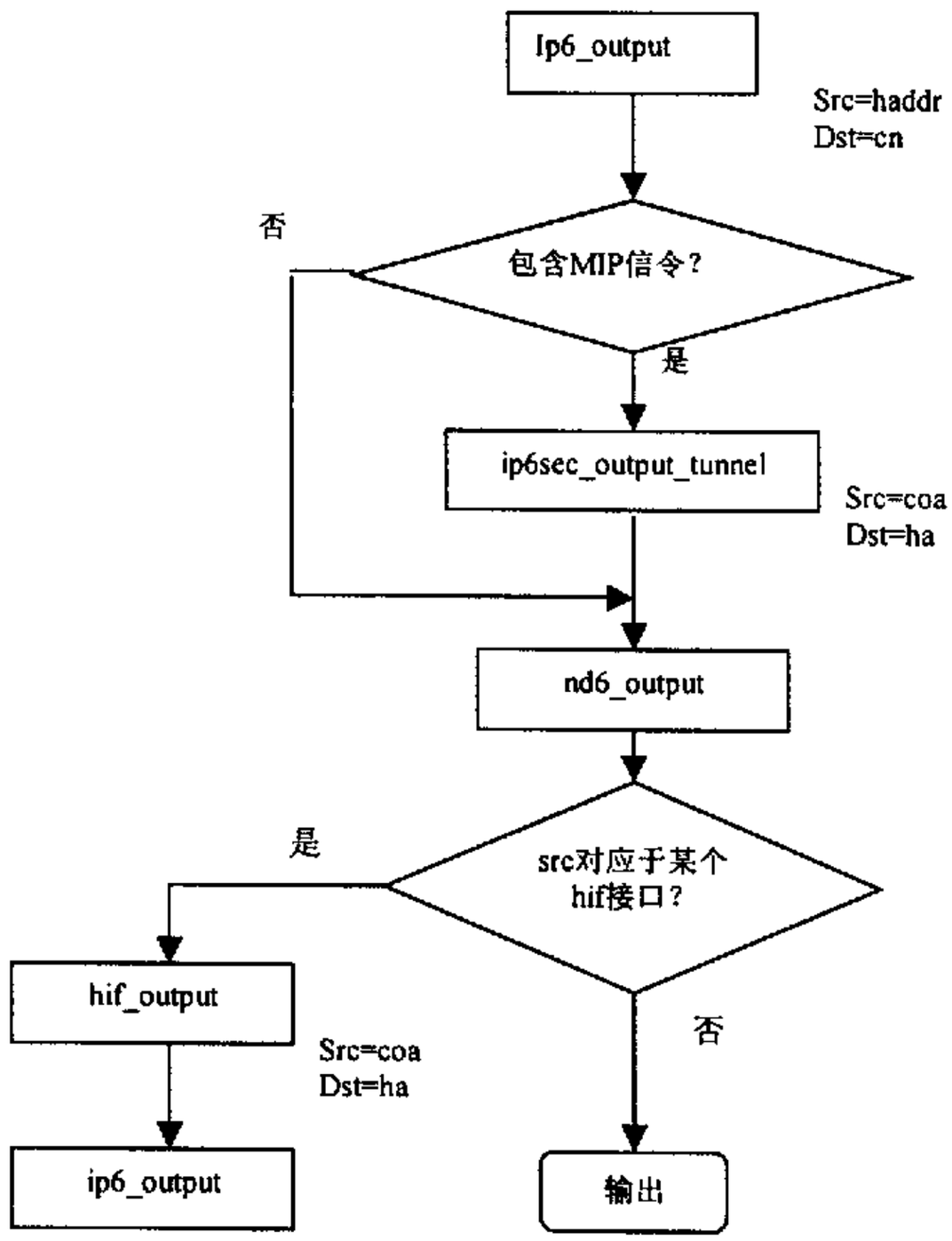


图 2-22 移动节点上的 MIP 封装机制简图

在 ip6_ouput()完成对 IP 层对数据报的处理后，将数据报向下传递给 nd6_output()继续处理。在 nd6_output 中，如果一个数据报的源地址是家乡地址，并且这个家乡地址已经经过注册，也就是说，程序中这个地址对应这移动节点上的某个 hif 接口，那么这个数据报将被传送给 hif_output()处理，hif_output 给数据报添加上一个 IPv6 报头，再次传递给 ip6_output()，重新进行输出处理；否则，nd6_output 直接调用接口输出函数输出数据包。

移动节点通过反向隧道向通信对端发送数据报的时候，数据报的处理流程如

图 2-22 所示：移动节点调用 ip6_output()处理待发送的分组，分组的源地址为移动节点的家乡地址，目的地址为通信对端的地址。如果分组包含移动 IP 信令，安全策略数据库会要求对这个分组进行 ESP 封装，分组提交给 ipsec6_output_tunnel()处理。处理后的分组源地址变成移动节点的转交地址，目的地址变成移动节点的家乡代理的地址。由于分组的源地址变成了转交地址，所以在 nd6_output 中不会再将这个分组传递给 hif_output()处理，而是直接调用接口输出函数输出分组。如果分组不含任何移动 IP 信令，那么不会对分组专门进行 ESP 封装，在这个分组传递给 nd6_output()处理的时候，分组的源地址仍然是移动节点的家乡地址。这样，便可以在移动节点上找到一个对应的 hif 接口，并将分组递交给该 hif 接口的接口处理函数 hif_output()进行处理。处理后的分组源地址为移动节点转交地址，目的地址为移动节点的家乡代理地址。这个分组被提交给 ip6_output()继续处理的时候，由于分组的源地址已经变成了转交地址，所以不会再对分组进行任何与移动 IP 有关的封装处理。可以看出，当移动节点发往通信对端的数据报中含有 MIPv6 信令的时候，我们使用函数 ipsec6_output_tunnel()进行 ESP 封装，否则，我们使用函数 hif_output()进行简单封装。

2.3.7.4 家乡代理上的封装处理

通信对端向移动节点发送数据报的时候，如果通信对端上面不存在关于该移动节点绑定缓存，那么把数据报发往移动节点的家乡代理来封装转发。家乡代理上调用 ip6_forward()来转发这些数据报。ip6_forward()在转发数据报的时候，首先查询安全策略，察看对数据包的处理策略。这里的安全策略与移动节点上发送数据报的安全策略相同：如果数据报中含有移动 IP 信令（比如 HOT 等），那么这个数据报必须进行加密，KAME/MIP 中是进行 ESP 封装。如果数据报中不含有任何移动 IP 信令，而只是通常的数据报，那么这些数据报没有必要专门进行保护。这里，IPSec 提供的查询函数为 IPSec6_getpolicybyaddr()，提供的隧道处理函数同样是 IPSec6_output_tunnel()。

在完成 IPSec 相关处理后，ip6_forward()调 mip6_bc_list_find_withphaddr()检查家乡代理上是否存在一条绑定缓存，它的对端地址等于这个分组的目的地址，也就是检查分组的目的地址是不是某一个已经在该家乡代理注册过的移动节点

的家乡地址。如果存在这样的绑定缓存项，分组提交 mip6_tunnel_output()处理。mip6_tunnel_output()给数据报添加上一个 IPv6 报头，再次传递给 ip6_output()，重新进行处理；否则，ip6_forward 将分组传递给 nd6_output()继续处理。

在家乡代理上，由于不存在 hif 接口，或者说 hif 接口都为空，那么 nd6_output()不会再对分组进行封装，而是在完成相关处理后直接调用接口输出函数输出分组。

家乡代理转发通信节点发往移动节点的数据报的时候，数据报处理流程如图 2-23 所示：

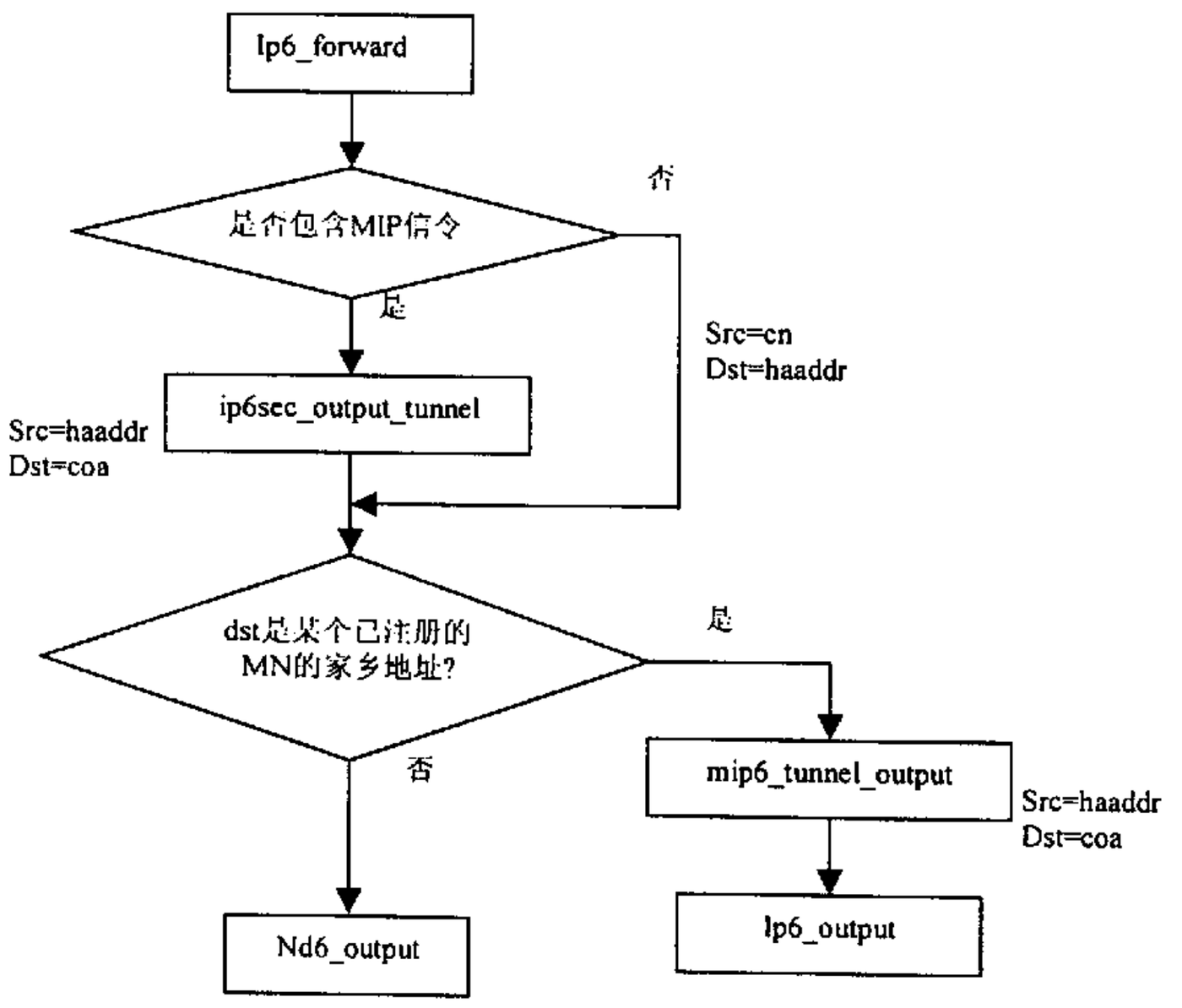


图 2-23 家乡代理上的 MIP 封装机制简图

移动节点调用 ip6_forward()处理待转发的分组，分组的源地址为通信节点的地址，目的地址为移动节点的家乡地址。如果分组包含移动 IP 信令，安全策略数据库会要求对这个分组进行 ESP 封装，分组提交给 IPSec6_output_tunnel()进行 ESP 封装，封装后分组的源地址变成家乡代理的地址，目的地址变成移动节点的转交地址。由于分组的目的地址变成了转交地址，所以 mip6_bc_list_find_withphadd()不会再找到对应的绑定缓存项，从而分组将绕过 mip6_tunnel_output()的处理。相反，如果分组不含任何移动 IP 信令，那么不会对分组专门进行 ESP 封装。在这种情况下，由于分组的目的地址为移动节点家

乡地址，那么 mip6_bc_list_find_withphadd()将必定能够从 mip6_bc_list 中发现一条对应的绑定缓存，从而调 mip6_tunnel_output()进行简单封装，封装后分组的源地址为家乡代理的地址，目的地址为移动节点的转交地址，这个分组被传递给 ip6_output()的时候，ip6_output()把它当作通常的分组直接发送给移动节点。 这样，当通信对端发往移动节点的数据报中含有 MIPv6 信令的时候，我们使用函数 IPSec6_output_tunnel()进行 ESP 封装，否则，我们使用函数 mip6_tunnel_output()进行简单封装。

2.3.7.5 移动 IP 封装后数据报的格式

当在移动节点和家乡代理之间建立 IPSec 隧道的时候，我们必须为它们准备好安全关联。安全关联两端所使用的地址必须是移动节点的家乡地址和家乡代理的地址。这里，在安全关联中不可以使用移动节点的转交地址，因为当移动节点位置改变后，它的转交地址便可能会发生变化。因此，根据上面讨论可以小节如下：

1. 当移动节点向通信对端发送移动 IP 信令的时候，移动节点发出的数据报格式如下：

   IP(SRC=careof_addr, DST=HA)

   DSTOPT(HAO=home_addr)

   ESP

   IP(SRC=home_addr, DST=CN) -> encrypted

   MH -> encrypted

2. 当通信对端向移动节点发送移动 IP 信令的时候，数据报经过家乡代理转发给移动节点。家乡代理处理后发出的数据报格式如下：

   IP(SRC=HA, DST=careof_addr)

   RTHDR2(home_addr)

   ESP

   IP(SRC=CN, DST=home_addr) -> encrypted

   MH -> encrypted

3. 当移动节点向通信对端发送普通的数据报的时候，移动节点发出的数据报格式如下：

   IP(SRC=careof_addr, DST=HA)

IP(SRC=home_addr, DST=CN)

PAYLOAD

4. 当通信对端向移动节点发送普通的数据报的时候，家乡代理处理后转发的数据报格式如：

IP(SRC=HA, DST=careof_addr)

IP(SRC=CN, DST=home_addr)

PAYLOAD

在上述的四种情形中，1 和 2 由 IPSec6_output_tunnel()处理（需要合适的 SA 对和 SPD），3 由 hif_output()完成，4 由 mip6_tunnel_output()完成。1 和 2 为 ESP 封装，3 和 4 为简单封装。

2.3.7.6 解封装过程

在隧道的出口处，需要对数据报进行解封装。所有输入的数据报都会由下层交给 ip6_input 进行处理。它对到达分组进行处理，主要任务包括分组的合法性验证、转发处理、扩展选项处理及上层递送等。如果头标为 IPPROTO_MOBILITY（图中标注为 MOBILITY ），则调用函数 mobility6_input，启动移动 IP 输入处理。如果头标为 IPPROTO_IPV6，则调用函数 encap6_input。如果头标为 IPPROTO_ESP，那么交给 esp6_input 进行处理。

简单封装数据包的解封装工作在 encap6_input 中进行，ESP 封装数据报的解封装工作在 esp6_input 中进行。

在 encap6_input 对数据报进行处理的时候，因为存在各种不同类型的隧道，而各种隧道所用的都是各自不同的封装机制，这样带给我们的问题是我们在实现过程中如何去查找正确的上层隧道处理，单凭 IP 包的协议类型项来判断到底由那一条隧道来处理已经不能满足目前的要求，为此，协议栈增加了结构体 encaptab 构成封装信息表，查找时通过查看和隧道的起始终止节点地址对匹配的表项来确定隧道的处理。hif 对简单封装数据报的解封装过程与 gif 类似。先在封装表中查找到对应的表项 match（encap6_input 中调用 encap6_lookup 进行），然后提出出该封装项的协议项（psw = (const struct ip6protosw *)match->psw），在这里，hif 接口的所用的协议为 mip6_tunnel_protosw。最后把数据报传递给 psw->pr_input 进行解封装处理，这里 hif 所用的解封装函数为 mip6_tunnel_input。

mip6_tunnel_input 把数据报剥去一个 IPv6 首部。

2.3.8 报文输入输出与转发机制

移动 IP 模块是建立在 IP 模块的基础之上，对移动 IP 模块中 IP 数据报的处理作了一定的变动，使其符合移动性对数据报格式的要求。本模块所涉及的 ip6_input 函数、ip6_output 函数和 ip6_forward 函数以及所调用的其它一些相关函数，都要按照移动 IP 协议的要求进行修改。函数 ip6_input 在将数据报向上层传送前，必须调用函数 dest_mip_hao，如果存在家乡地址目标选项，就将家乡地址域中的地址（移动节点家乡地址）与报头中的源地址（移动节点转交地址）相交换，将数据报的家乡地址传给上层，从而使移动 IP 对于上层协议来说是透明的。各节点在调用 ip6_output 向移动主机发送数据报时，必须首先查找绑定缓存表，把发往移动节点家乡地址的数据报改为发往移动节点的转交地址。

2.3.9 网管命令接口

配置和管理有关MIP6相关信息，该模块需要完成的功能包括：配置单个移动节点及移动功能开启、配置家乡代理及使能、显示移动节点的家乡地址、显示家乡代理列表、显示绑定更新列表、显示绑定高速缓冲条目。

系统初始化时，要对网络进行配置，之外，网络管理员也可以通过配置命令进行配置。处理过程主要是通过 socket 与内核进行通信，调用一些系统函数如 ioctl、sysctl、write 和 read 对内核中的接口状态和一些表进行操作。图 2-24 显示了网络管理模块中的基本架构。

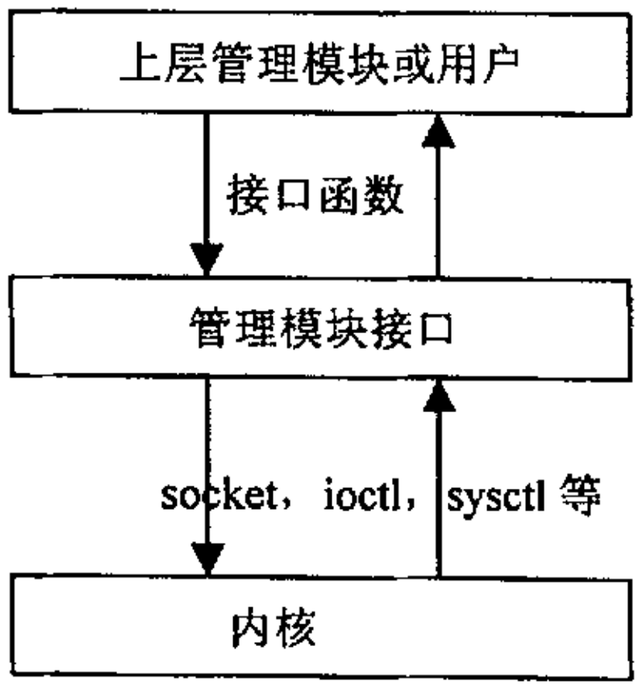


图 2-24 网络管理接口函数

移动 IP 通过函数 mip6_ioctl 和 hif_ioctl 提供给用户控制和配置的接口，用户

通过命令 mip6control 和 mip6config 控制移动 IP 模块的运行和维护，该命令通过 ioctl 将命令参数传递给本模块中的控制函数。

2.3.10 全局变量

本模块为实现方便首先定义了如下全局变量：

- hif_softc_list（struct hif_softc_list hif_softc_list）：指向移动节点的虚拟接口链表表头的指针；
- hif_coa（struct sockaddr_in6 hif_coa）：移动节点的外地转交地址；
- mip6_subnet_list （extern struct mip6_subnet_list mip6_subnet_list）：指向移动节点子网链表表头的指针；
- mip6_prefix_list （extern struct mip6_prefix_list mip6_prefix_list）：指向移动节点子网前缀链表表头的指针；
- mip6_ha_list （struct mip6_ha_list mip6_ha_list）：指向移动节点子网家乡代理链表表头的指针；
- mip6_bc_list （extern struct mip6_bc_list mip6_bc_list）：指向节点绑定缓存链表表头的指针；
- haifinfo_tab （extern struct hagent_ifinfo *haifinfo_tab）：指向家乡代理上家乡代理列表数据的第一个元素的指针；
- halist_expire_head （struct hagent_entry halist_expire_head）：指向家乡代理上所有家乡代理表项按有效时间大小排序的链表的表头；
- gaddr_expire_head （struct hagent_gaddr gaddr_expire_head）：指向家乡代理上所有家乡代理表项的全局地址按有效时间大小排序的链表的表头；
- mip6_config (struct mip6_config mip6_config)：网管配置参数全局变量；
- mip6stat (struct mip6stat mip6stat)：移动 IPv6 统计参数全局变量；
- mip6_tunnel_protosw （extern struct protosw mip6_tunnel_protosw）。

2.4 移动 IPv6 协议软件的实现

2.4.1 开发及运行平台的选择

与中兴通讯公司现有的 IPv6 协议栈保持一致，移动 IPv6 也采用商用的实时操作系统 Vxworks 作为开发和运行环境[23]。Vxworks 是目前应用最为广泛的一

种实时操作系统，它的集成开发环境叫 Tornado，这个集成开发环境提供了高效明晰的图形化的实时应用开发平台，包括一套完整的面向嵌入式系统的开发和调测工具。Tornado 环境采用主机－目标机交叉开发模型，应用程序在主机的 Windows 环境下编译链接生成可执行文件，下载到目标机，通过主机上的目标服务器与目标机上的目标代理的通信完成对应用程序的调试、分析。由于 Vxworks 所具有的开发方便的特点，而且还提供了非常完备的 TCP/IP 协议栈网络支持以及众多的可选模块。

开发本协议栈软件系统需要的硬件环境如图 2-25 所示：

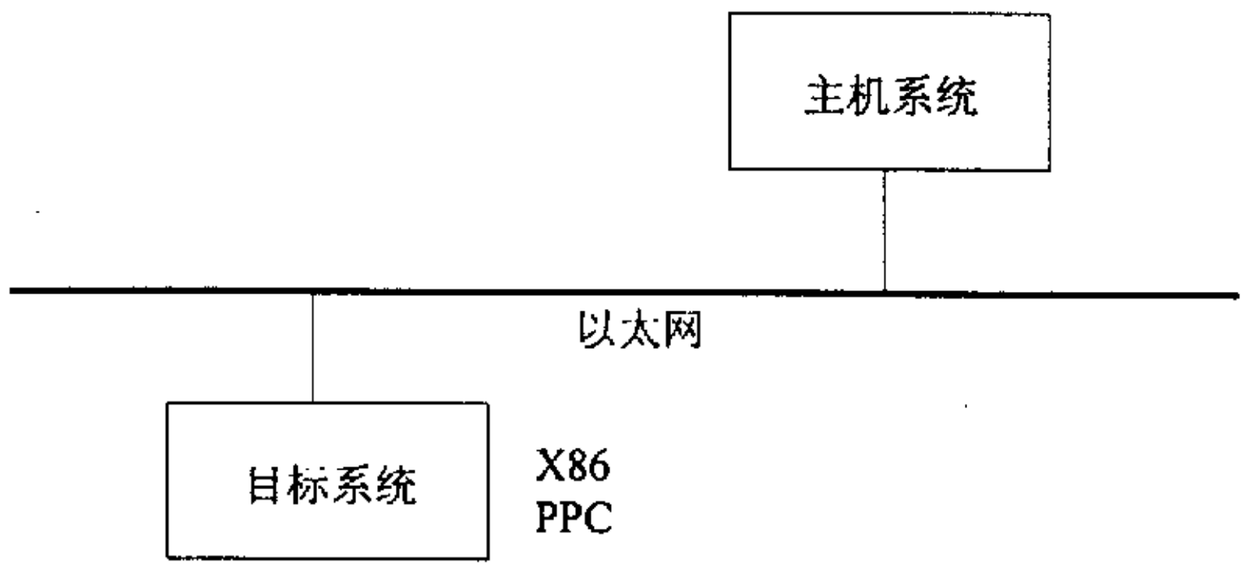


图2-25 协议栈软件硬件开发环境

图 2-25 中主机系统采用普通的 PC 机，通过以太网对相应的目标系统进行开发、调试，目标系统可以采用 x86、PPC 等处理系统，协议栈软件将首先在 x86 系统上实现，因此该目标系统可以采用通用 PC 机仿真。

协议栈系统目前运行在 x86 系统中，按照设计原 IPv6 协议栈软件能够适用于多种 CPU 体系中，经过简单的移植，新协议栈也将能够运行在 PPC、ARM、MIPS 等硬件环境中。

原来协议栈软件系统采用与操作系统无关的设计，通过操作系统封装层屏蔽底层操作系统的差别，能够在不同的实时操作系统环境下运行。实际运行的操作系统选择在 Vxworks 下运行，采用标准 C 语言实现。移动 IPv6 模块将继续沿用和保持这些风格。

2.4.2 开发方法和技术路线

移动 IPv6 协议是比较庞杂和复杂的协议，需要进行大量的研究和开发工作，

工作量比较大，因此选择一个合适的技术路线和开发方法是非常重要的。由于该协议的内容比较多，在现有的人力和财力下完全从零开始设计、实现移动 IPv6 模块是不可能按时完成的，因此必须另找途径进行开发。

目前 IPv6 的应用已经提上日程，许多相关的机构已经建立 IPv6 的实验床测试 IPv6 的运行情况，同时有许多的组织和机构在 IPv6 协议实现方面作了大量的工作，其中有一些 IPv6 的协议实现包含了移动 IPv6 的实现，并且代码是完全公开的,同时其包括移动性在内的各种功能都得到了不断的丰富完善。

考虑到以上情况并结合论文工作的实际，我们认为充分利用已有的成果，选择一套结构清晰、具备移动性功能、运行稳定的开放代码协议栈，对其移动性功能部分的实现代码进行研究、消化吸收，并根据最新的协议规范进行改造，是一个省时省力的高效方法，因此本课题的开发主要采用这种方式。采取该方法能够大大降低整个软件协议栈的开发成本，缩短协议开发周期，是一个省时省力的方法，同时这种思路使我们完全能够掌握移动 IPv6 协议栈的技术细节，避免采用商用协议栈时的一些弊端，如技术支持不够到位，以及受到其他一些非技术因素的影响。

目前，国际上有很多组织和研究机构都在实现 IPv6 的协议栈，因此本课题可以利用的资源较多，经过调研我们初步选择采用 KAME[26]协议栈软件作为我们研究、改进和扩充的蓝本。KAME 协议栈可以通过互联网免费获取，能够较好的支持 IPv6 的移动性功能。它是日本几大公司联合支持的 IPv6 项目，更新速度比较快，同时在 6Bone 等实验床上进行了广泛的测试，比较稳定。这个协议栈软件是源代码开放的，针对的操作系统平台是 FreeBSD，我们要在消化吸收的基础上将它的移动性代码移植到到 Vxworks 系统下面来，同时对其不完善的地方进行扩充和优化。在进行实际的移植工作之前，需要在调研、协议理解和代码阅读以及技术交流方面做许多工作，作好充分准备。

2.4.3 开发过程

出于 FreeBSD 下面移动 IPv6 协议的实现比较复杂，所以本文花大量的时间在协议的分析和实现代码的分析上，这些工作已经部分体现在了前文的叙述当中。首先将移动 IPv6 的代码从 IPv6 协议栈源代码中剥离出来，然后把程序代码和协议规范的各个部分对应起来分析比较，再进行具体的协议移植工作并自行编

写部分代码，将移植过来的代码软件模块与自行开发的软件模块结合起来进行调试，最终完成移动 IPv6 协议在 Vxworks 系统下的实现。

第三章 移动 IPv6 协议栈软件的功能测试

在开发过程当中和开发结束之后，都需要对协议栈软件进行测试。协议的一致性和互操作性将直接影响到网络运行的效率，但是由于在 IPv6 架构下的测试手段和测试设备比较欠缺，也由于移动 IPv6 还没有形成正式的标准，严格的一致性测试和互操作性测试失去了意义，因此，本文测试工作的重点是测试协议栈能否正常工作，移动 IPv6 的基本功能是否已经实现。本章主要进行功能测试，通过对七种基本情形下移动 IPv6 协议栈软件的工作情形进行观察，来验证是否已经实现了的移动 IPv6 协议所定义的基本功能。本章的功能测试工作在 Intel X86 硬件调试平台上进行。

3.1 测试内容

测试内容包括：

- 移动节点移动性检测功能；
- 无 IPSec 保护时移动节点向家乡代理注册外地转交地址功能；
- 无 IPSec 保护时移动节点回到家乡后注销外地转交地址功能；
- IPSec 保护时移动节点向家乡代理注册外地转交地址功能；
- IPSec 保护时移动节点回到家乡后注销外地转交地址功能；
- 移动节点向通信节点注册外地转交地址功能；
- 移动节点向通信节点注销外地转交地址功能；

3.2 测试环境

- 硬件环境：四台运行 IPv6 协议的目标机，运行微软 IPv6 软件包的两台主机，基本网络设备（如网线、交叉线、HUB 等）。
- 软件环境：Vxworks 实时操作系统，IPv6 协议软件影像文件,Sniffer 抓包工具等。

为测试移动 IPv6 相关功能，搭建如图 3-1 所示的测试环境：

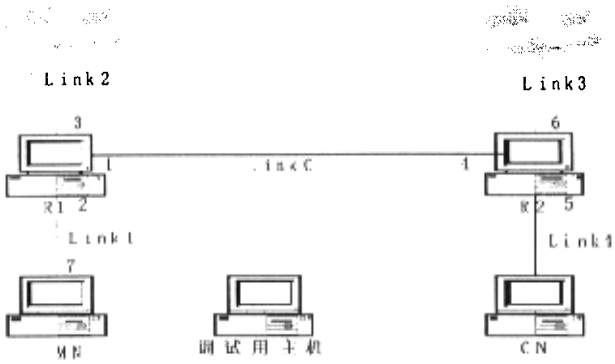


图 3-1 功能测试环境示意图

其中 HA 运行 IPv6 协议栈软件并配置成移动节点家乡代理的路由器，与其相连的有三个子网：Link0、Link1、Link2。三个子网配置的 IPv6 网络前缀是：

Link0：3ffe:0：0:cd31::/64

Link1：3ffe:0：0:cd32::/64

Link2：3ffe:0：0:cd33::/64

根据相连子网的前缀和接口的ID号，为HA的三个网口配置如下的全局IPv6地址：

接口 1：3ffe:0：0:cd31：200：4bff：feb1：86e0

接口 2：3ffe:0：0:cd32：260：8ff：fec6：2c4c

接口 3：3ffe:0：0:cd33：204：76ff：fe71：3ae4。

Router 运行 IPv6 协议栈软件的普通路由器，与其相连的有三个子网：Link0、Link3、Link4。三个子网配置的 IPv6 网络前缀是：

Link0：3ffe:0：0:cd31::/64

Link3：3ffe:0：0:cd34::/64

Link4：3ffe:0：0:cd35::/64

根据相连子网的前缀和接口的 ID 号，为 Router 的三个网口配置如下的全局 IPv6 地址：

接口 4：3ffe:0：0:cd31：200：b4ff：fe8b：2f41

接口 5：3ffe:0：0:cd34：200：b4ff：fe84：7272

接口 6：3ffe:0：0:cd35：210：5aff：fe9a：84f1。

MN 为运行 IPv6 协议栈软件的主机，通过网管命令将其配置成移动节点。

MN 的全局 IPv6 家乡地址和转交地址都通过自动配置方法得到。在调试中，MN 自动配置得到的家乡地址为：3ffe:0：0:cd35：210：4bff：fe23：311f。

CN 为运行 IPv6 协议栈软件的普通主机，其全局 IPv6 地址也通过自动配置方法得到。MN 和 CN 之间要做路由优化功能的测试。

调试用主机是一台运行 WIN2000 操作系统的计算机，其上安装了微软的 IPv6 软件包、Tornado2.0、Sniffer 等工具。HA、Router、MN、CN 上的运行的 IPv6 软件影像是从调试用主机上下载的；Tornado 工具用来调试跟踪 HA、Router、MN、CN 上运行的协议栈代码。

3.3 测试方法与步骤

3.3.1 移动节点移动性检测功能：

如图 1 所示的组网示意图连接 HA 和 Router；当 HA 从主机下载工程映象后按如下网管命令配置好 HA：

（1）接口 1 的配置：

IPv6 nd prefix-advertisement 3ffe:0:0:cd31::1/64 300 300 onlink autoconfig

IPv6 address 3ffe::cd31:200:b4ff:feb1:86e0/64

（2）接口 2 的配置：

IPv6 nd prefix-advertisement 3ffe:0:0:cd32::1/64 300 300 onlink autoconfig

IPv6 nd home-agent-config-flag

IPv6 nd ra-interval 1

IPv6 address 3ffe::cd32:260:8ff:fec6:2c4c/64

IPv6 nd send-ra

（3）接口 3 的配置：

IPv6 nd prefix-advertisement 3ffe:0:0:cd33::1/64 300 300 onlink autoconfig

IPv6 nd ra-interval 1

IPv6 address 3ffe::cd33:204:76ff:fe71:3ae4/64

IPv6 nd send-ra

（4）HA 上静态 IPv6 路由条目配置成：

IPv6 route 3ffe:0:0:cd34::0/64 3ffe::cd31:200:b4ff:fe8b:2f41

IPv6 route 3ffe:0:0:cd35::0/64 3ffe::cd31:200:b4ff:fe8b:2f41

（5）HA 上家乡代理功能使能：

enable homeagent

当 Router 从主机下载工程映象后按如下网管命令配置好 Router：

（1）接口 3 的配置：

IPv6 nd prefix-advertisement3ffe:0:0:cd31::1/64 300 300 onlink autoconfig

IPv6 address 3ffe::cd31:200:b4ff:fe8b:2f41/64

（2）接口 4 的配置：

IPv6 nd prefix-advertisement 3ffe:0:0:cd34::1/64 300 300 onlink autoconfig

IPv6 nd ra-interval 1

IPv6 address 3ffe::cd34:200:b4ff:fe84:7272/64

IPv6 nd send-ra

（3）接口 5 的配置：

IPv6 nd prefix-advertisement 3ffe:0:0:cd35::1/64 300 300 onlink autoconfig

IPv6 nd ra-interval 1

IPv6 address3ffe::cd35:210:5aff:fe9a:84f1/64

IPv6 nd send-ra

（4）Router 上静态 IPv6 路由条目配置成：

IPv6 route 3ffe:0:0:cd32::0/64　3ffe::cd31:200:b4ff:feb1:86e0

IPv6 route 3ffe:0:0:cd33::0/64　3ffe::cd31:200:b4ff:feb1:86e0

当 MN 从主机下载工程映象后按如下网管命令配置好 MN：

home_prefix 3ffe:0:0:cd32::1 64

homeagent 3ffe::cd32:260:8ff:fec6:2c4c fe80::260:8ff:fec6:2c4c

enable mobilenode

配置好以上节点后，把 MN 接入到 HA 和 Router 的各个子网中，并在各个子网间切换，通过 MN 上打印的调试信息可以判断移动节点在不同子网间切换时能否正确的执行移动性检测。

3.3.2 无 IPSec 保护时移动节点向家乡代理注册外地转交地址功能

如图 3-1 所示的组网示意图连接 HA 和 Router；

HA、Router、MN 从主机下载各自对应的工程映象后，按 3.3.1 所述网管命令分别配置好 HA、Router、MN；

把 MN 连到家乡链路上 Link1，用连在 Link1 和 Link3 上的运行微软 IPv6 软件包的主机 Ping 移动节点 MN：ping6 3ffe::cd32:210:4bff:fe23:311f –t.；

把 MN 从家乡链路上拔下，连到外地链路上，并在外地链路之间切换；MN 正确执行移动性检测后，向家乡代理发送 BU 报文，注册其在外地自动配置的转交地址；

用在 HA 上显示的调试信息查看 BU 是否正确注册、HA 和 MN 之间的双向隧道是否成功建立、MN 的家乡地址的 DAD 检测是否成功、HA 是否已经作为 MN 的家乡地址的邻机发现功能的 Proxy；

主机上的 ping6 3ffe::cd32:210:4bff:fe23:311f –t 在 MN 向 HA 成功注册后能继续 ping 通在外地子网上的移动节点 MN。

3.3.3 无 IPSec 保护时移动节点向家乡代理注销外地转交地址功能

调试步骤如下：

如图 3-1 所示的组网示意图连接 HA 和 Router；

HA、Router、MN 从主机下载各自对应的工程映象后，按 3.3.1 所述网管命令分别配置好 HA、Router、MN；

把 MN 连到外地链路上 Link3，MN 向 HA 成功注册外地转交地址后，用连在 Link1 和 Link3 上的运行微软 IPv6 软件包的主机 Ping 移动节点 MN 的家乡地址：ping6 3ffe::cd32:210:4bff:fe23:311f –t；

把 MN 从外地链路上拔下，连到家乡链路上，MN 正确执行移动性检测后，向家乡代理发送 BU 报文，注销其在外地自动配置的转交地址；

用在 HA 上显示的调试信息查看 BU 是否正确注销、HA 和 MN 之间的双向隧道是否成功删除、HA 是否已经不作为 MN 的家乡地址的邻机发现功能的 Proxy；

主机上的 ping6 3ffe::cd32:210:4bff:fe23:311f –t 在 MN 向 HA 成功注销后能继续 ping 通在家乡子网上的移动节点 MN。

3.3.4 IPSec 保护下移动节点向家乡代理注册外地转交地址功能

如图所示的组网示意图连接 HA 和 Router；当 HA 从主机下载工程映象后，

按 3.3.1 所述网管命令配置好 HA，同时用网管命令在 HA 上配置 IPSec 的安全策略库和相应的安全关联库如下：

安全策略库

spd add 3ffe::cd32:260:8ff:fec6:2c4c/128 3ffe::cd32:210:4bff:fe23:311f/128 6 out IPSec esp transport require

spd add 3ffe::cd32:210:4bff:fe23:311f/128 3ffe::cd32:260:8ff:fec6:2c4c/128 5 in IPSec esp transport use

安全关联库

sad add 3ffe::cd32:260:8ff:fec6:2c4c 3ffe::cd32:210:4bff:fe23:311f esp 65543 trans 200000 150000 0 des-cbc "wangzh" keyed-md5 "tiandongxubook" 0

sad add 3ffe::cd32:210:4bff:fe23:311f 3ffe::cd32:260:8ff:fec6:2c4c esp 65542 trans 200000 150000 0 des-cbc "wangzh" keyed-md5 "wangzhongabook" 0

当 Router 从主机下载工程映象后，按 3.3.1 所述网管命令配置好 Router；

当 MN 从主机下载工程映象后按 3.3.1 所述网管命令配置好 MN,同时用网管命令在 MN 上配置 IPSec 的安全策略库和相应的安全关联库如下：

安全策略库

spd add 3ffe::cd32:210:4bff:fe23:311f/128 3ffe::cd32:260:8ff:fec6:2c4c/128 6 out IPSec esp transport require

spd add 3ffe::cd32:260:8ff:fec6:2c4c/128 3ffe::cd32:210:4bff:fe23:311f/128 5 in IPSec esp transport use

安全关联库

sad add 3ffe::cd32:210:4bff:fe23:311f 3ffe::cd32:260:8ff:fec6:2c4c esp 65542 trans 200000 150000 0 des-cbc "wangzh" keyed-md5 "wangzhongabook" 0

sad add 3ffe::cd32:260:8ff:fec6:2c4c 3ffe::cd32:210:4bff:fe23:311f esp 65543 trans 200000 150000 0 des-cbc "wangzh" keyed-md5 "tiandongxubook" 0

把 MN 连到家乡链路上 Link1，用连在 Link1 和 Link3 上的运行微软 IPv6 软件包的主机 ping 移动节点 MN：ping6 3ffe::cd32:210:4bff:fe23:311f –t.

把 MN 从家乡链路上拔下，连到外地链路上，并在外地链路之间切换；MN 正确执行移动性检测后，向家乡代理发送 BU 报文，注册其在外地自动配置的转

交地址。

用在 HA 上显示的调试信息查看 BU 是否正确注册、HA 和 MN 之间的双向隧道是否成功建立、MN 的家乡地址的 DAD 检测是否成功、HA 是否已经作为 MN 的家乡地址的邻机发现功能的 Proxy。此外，查看主机上的 ping6 3ffe::cd32:210:4bff:fe23:311f –t 在 MN 向 HA 成功注册后能继续 ping 通在外地子网上的移动节点 MN.，如果能够继续使用它原来的地址与对段保持通信，说明成功。

3.3.5 IPSec 保护下移动节点向家乡代理注效外地转交地址功能

如图所示的组网示意图连接 HA 和 Router:

当 HA 从主机下载工程映象后，按 3.3.1 所述网管命令配置好 HA

同时用网管命令在 HA 上配置 IPSec 的安全策略库和相应的安全关联库如下：

安全策略库

spd add 3ffe::cd32:260:8ff:fec6:2c4c/128 3ffe::cd32:210:4bff:fe23:311f/128 6 out IPSec esp transport require

spd add 3ffe::cd32:210:4bff:fe23:311f/128 3ffe::cd32:260:8ff:fec6:2c4c/128 5 in IPSec esp transport use

安全关联库

sad add 3ffe::cd32:260:8ff:fec6:2c4c 3ffe::cd32:210:4bff:fe23:311f esp 65543 trans 200000 150000 0 des-cbc "wangzh" keyed-md5 "tiandongxubook" 0

sad add 3ffe::cd32:210:4bff:fe23:311f 3ffe::cd32:260:8ff:fec6:2c4c esp 65542 trans 200000 150000 0 des-cbc "wangzh" keyed-md5 "wangzhongabook" 0

当 Router 从主机下载工程映象后，按 3.3.1 所述网管命令配置好 Router.

当 MN 从主机下载工程映象后按 3.3.1 所述网管命令配置好 MN,同时用网管命令在 MN 上配置 IPSec 的安全策略库和相应的安全关联库如下：

安全策略库

spd add 3ffe::cd32:210:4bff:fe23:311f/128 3ffe::cd32:260:8ff:fec6:2c4c/128 6 out IPSec esp transport require

spd add 3ffe::cd32:260:8ff:fec6:2c4c/128 3ffe::cd32:210:4bff:fe23:311f/128 5 in IPSec esp transport use

安全关联库

sad add 3ffe::cd32:210:4bff:fe23:311f 3ffe::cd32:260:8ff:fec6:2c4c esp 65542 trans 200000 150000 0 des-cbc "wangzh" keyed-md5 "wangzhongabook" 0

sad add 3ffe::cd32:260:8ff:fec6:2c4c 3ffe::cd32:210:4bff:fe23:311f esp 65543 trans 200000 150000 0 des-cbc "wangzh" keyed-md5 "tiandongxubook" 0

把 MN 连到家乡链路上 Link1，用连在 Link1 和 Link3 上的运行微软 IPv6 软件包的主机 Ping 移动节点 MN：ping6 3ffe::cd32:210:4bff:fe23:311f –t；

把 MN 从家乡链路上拔下，连到外地链路上，并在外地链路之间切换；MN 正确执行移动性检测后，向家乡代理发送 BU 报文，注册其在外地自动配置的转交地址；

用在 HA 上显示的调试信息查看 BU 是否正确注册、HA 和 MN 之间的双向隧道是否成功建立、MN 的家乡地址的 DAD 检测是否成功、HA 是否已经作为 MN 的家乡地址的邻机发现功能的 Proxy；

主机上的 ping6 3ffe::cd32:210:4bff:fe23:311f –t 在 MN 向 HA 成功注册后能继续 ping 通在外地子网上的移动节点 MN。

3.3.6 移动节点向通信节点注册外地转交地址功能

如图 3-1 所示的组网示意图连接 HA 和 Router；

HA、Router、MN 从主机下载各自对应的工程映象后，按 3.3.1 所述网管命令分别配置好 HA、Router、MN。

把 MN 连到家乡链路上 Link1；

当 CN 从主机下载工程映象，将其连到外地链路 Link4 上；

把 MN 从家乡链路上拔下，连到外地链路 Link3 上，MN 正确执行移动性检测后，向家乡代理发送 BU 报文，注册其在外地自动配置的转交地址。

在 CN 上用 ping6 命令 ping 移动节点家乡地址 3ffe::cd32:210:4bff:fe23:311f，MN 收到 HA 通过隧道转发过来的 ICMP 报文后，向 CN 注册其在外地的转交地址，进行路由优化。

用在 CN 上显示的调试信息查看 MN 发来 BU 是否正确注册。

断开 HA 和 Router 的连接 Link0，连在 Link4 上 CN 的 ping6 3ffe::cd32:210:4bff:fe23:311f 命令 Ping 连在 Link3 上的 MN 的家乡地址。

3.3.7 移动节点向通信节点注销外地转交地址功能

如图 3-1 所示的组网示意图连接 HA 和 Router；

HA、Router、MN 从主机下载各自对应的工程映象后，按 3.3.1 所述网管命令分别配置好 HA、Router、MN。

把 MN 连到外地链路 Link3 上，MN 正确执行移动性检测后，向家乡代理发送 BU 报文，注册其在外地自动配置的转交地址。

当 CN 从主机下载工程映象，将其连到外地链路 Link4 上；

在 CN 节点上使用 ping6 命令来 ping 移动节点的家乡地址：ping6 3ffe::cd32:210:4bff:fe23:311f，MN 收到 HA 通过隧道转发过来的 ICMP 报文后，向 CN 注册其在外地的转交地址，进行路由优化。

把 MN 从外地链路 Link3 上拔下，连到其家乡链路 Link0 上，MN 正确执行移动性检测后，向 CN 发送 BU 报文，注效其在外地自动配置的转交地址。

用 CN 上的 ping6 命令来 ping 移动节点的家乡地址：ping6 3ffe::cd32:210:4bff:fe23:311f，MN 收到 HA 转发过来的 ICMP 报文后，不再向 CN 发送 BU 报文进行路由优化。

3.4 测试结果

1. 移动节点 MN 移动性检测功能：

a) MN 上能正确显示在子网间切换的信息。

b) MN 在各子网间切换时，其判断已经移动了的快慢与路由器发送 RA 通告的时间间隔密切相关，当 RA 以每秒一次的频率发送时，切换时间在两秒钟左右，而当 RA 以十秒钟一次的频率发送时，切换时间在十秒钟左右。

c) 移动 IPv6 模块能正确地实现 MN 的移动性检测功能。

2. 无 IPSec 保护时移动节点向家乡代理注册外地转交地址功能：

a) MN 从一个外地子网移到另一个子网后，向 HA 发送了新的注册报文，通过 Tornado 的调试工具查看 HA 上的绑定缓存，能看到关于移动节点的新的转交地址的绑定缓存项。

b) MN 在各外地子网间切换时，主机能通过 MN 的家乡地址 ping 继续通 MN。

c) 移动 IPv6 模块能正确地实现无 IPSEC 保护时移动节点向家乡代理注册外地转交地址功能。

3. 无 IPSEC 保护时移动节点回到家乡后注销外地转交地址功能：

a) MN 从一个外地子网移回到家乡子网后，向 HA 发送注销报文，通过 Tornado 的调试工具查看 HA 上的绑定缓存，能看到关于移动节点的绑定缓存项已不存在。

b) 从外地子网回到家乡子网时，主机能继续通过 MN 的家乡地址 ping 通 MN。

c) 移动 IPv6 模块能正确地实现无 IPSec 保护时移动节点向家乡代理注削外地转交地址功能。

4. IPSec 保护时移动节点向家乡代理注册外地转交地址功能：

a) MN 从一个外地子网移到另一个子网后，向 H A 发送了新的注册报文，通过 Tornado 的调试工具查看 HA 上的绑定缓存，能看到关于移动节点的新的转交地址的绑定缓存项。

b) MN 在各外地子网间切换时，主机能通过 MN 的家乡地址 ping 继续通 MN。

c) 移动 IPv6 模块能正确地实现无 IPSec 保护时移动节点向家乡代理注册外地转交地址功能。

5. IPSEC 保护时移动节点回到家乡后注销外地转交地址功能：

a) MN 从一个外地子网移回到家乡子网后，向 HA 发送注销册报文，通过 Tornado 的调试工具查看 HA 上的绑定缓存，能看到关于移动节点的绑定缓存项已不存在。

b) 从外地子网回到家乡子网时，主机能继续通过 MN 的家乡地址 ping 通 MN。

c) 移动 IPv6 模块能正确地实现无 IPSEec 保护时移动节点向家乡代理注削外地转交地址功能。

6. 移动节点向通信节点注册外地转交地址功能：

a) MN 从向 CN 发送注册报文，通过 Tornado 的调试工具查看 CN 上的绑定缓存，能看到关于移动节点的绑定缓存项已存在。

b) MN 发来 BU 正确注册后，断开 HA 和 Router 的连接 Link0，连在 Link4 上 CN 的 ping6 3ffe::cd32:210:4bff:fe23:311f 命令 ping 连在 Link3 上的 MN 的家乡地址。

c) 移动 IPv6 模块能正确地实现移动节点向 CN 注册外地转交地址，实现路由优化功能。

7. **移动节点向通信节点注销外地转交地址功能：**

a) MN 从向 CN 发送注销报文，通过 Tornado 的调试工具查看 CN 上的绑定缓存，能看到关于移动节点的绑定缓存项已不存在。

b) MN 发来 BU 正确注销后，用 CN 上的 ping6 命令来 Ping 移动节点的家乡地址，MN 收到 HA 转发过来的 ICMP 报文后，不再向 CN 发送 BU 报文进行路由优化。

c) 移动 IPv6 模块能正确地实现移动节点向 CN 注削外地转交地址的功能。

3.5 结论

在观察测试结果的时候，必须注意的是，并不是能够 ping 通就说明移动 IP 完成切换，需要根据观测到的实际结果，对照协议进行分析。在试验过程中，要注意观察 ping 报文的源地址，这个地址应该是移动节点的家乡地址。如果 ping 报文的源地址是转交地址，那么不能说明移动 IP 切换成功。

本章针对协议的工作特点，设计了七种基本情形来对移动 IPv6 的工作进行验证。测试结果表明，移动 IPv6 的基本功能正常，能够正确地进行移动检测、切换、注册和连续通信。这说明本文开发的协议软件已经实现了一个移动 IPv6 的基本功能集。以这个移动 IP 的调试环境为框架和基础，进行适当的扩展，添加必要的其他节点，引入更有吸引力的业务，就可以构建一个丰富多彩的多媒体业务环境。

第四章 移动 IPv6 协议栈软件的性能分析

性能测试是协议测试的另一个重要方面，路由器是 IP 网络的核心设备，协议栈软件性能的好坏直接影响到其所在 IP 网络的规模、稳定性和可扩展性。本章对移动 IPv6 协议栈软件的性能进行了一些测试与分析，主要工作分为两个部分，第一部分测试移动 IPv6 模块的引入对路由器性能指标所带来的影响。软件协议栈功能的增加不可避免的会带来性能上的一些下降，一个完善的软件需要将这些不利因素控制在能够接受的范围之内，使其不会对路由器的整体性能造成明显的影响。第二部分则是对移动 IPv6 工作性能的一个关键指标即切换延迟时间进行了测量，切换延迟时间综合反应了协议设计与实现方案的优劣。本章的性能测试工作采用 ZTE GAR 硬件平台，它的 CPU 为 Motorola PPC。

4.1 移动 IPv6 模块的引入对路由器性能指标的影响

移动 IPv6 的引入，会给路由器性能带来负面的影响。这主要是由于路由协议在处理数据报的时候，必须额外花费一定的时间和资源来进行移动 IPv6 的软件处理，这将使得路由器的性能下降。比如，函数 ip6_input 在将数据报向上层传送前，必须调用函数 dest_mip_hao，如果存在家乡地址目标选项，就将家乡地址域中的地址（移动节点家乡地址）与报头中的源地址（移动节点转交地址）相交换，将数据报的家乡地址传给上层，从而使移动 IP 对于上层协议透明。各节点在调用 ip6_output 向移动主机发送数据报时，必须首先查找绑定缓存表，把发往移动节点家乡地址的数据报改为发往移动节点的转交地址。这些软件处理过程延缓了数据报的处理过程，造成路由器性能的下降。

为了测试移动 IPv6 功能的引入对原来协议栈的性能的影响，需要在相同的路由器硬件平台和网络环境下，分别在移动性功能引入的前后时间，对软件协议栈的一些关键性能指标进行测试，再将前后测试的结果进行对照和分析。下文首先分析了移动 IPv6 软件可能会影响到的一些主要的路由器性能参数，然后在中兴通讯 ZTE GAR 路由器硬件平台上，分别运行包含移动 IPv6 功能的协议栈软件和不包含移动 IPv6 功能的协议栈软件，对这些参数进行测量和比较。

4.1.1 测试内容

一般情况下，路由器性能测试包括以下指标中的一项或者几项：

- 吞吐量：测试路由器包转发的能力。通常指路由器在不丢包条件下每秒转发包的极限。
- 时延：测试路由器在吞吐量范围内从收到包到转发出该包的时间间隔。
- 丢包率：测试路由器在不同负荷下丢弃包占收到包的比例。不同负荷通常指从吞吐量测试到线速(线路上传输包的最高速率)，步长一般使用线速的 10%。
- 背靠背帧数：测试路由器在接收到以最小包间隔传输时不丢包条件下所能处理的最大包数。该测试实际考验路由器缓存能力，如果路由器具备线速能力（吞吐量=接口媒体线速），则该测试没有意义。
- 系统恢复时间：测试路由器在过载后恢复正常工作的时间。测试方法可以采用向路由器端口发送吞吐量 110%和线速间的较小值，持续 60 秒后将速率下降到 50%的时刻到最后一个丢包的时间间隔。如果路由器具备线速能力，则该测试没有意义。
- 系统复位：测试路由器从软件复位或关电重启到正常工作的时间间隔。正常工作指能以吞吐量转发数据。
- 路由表容量。
- 路由协议收敛时间等指标。

移动 IPv6 功能主要是通过软件实现的，所以它对系统硬件工作的状况影响不大，在上述性能指标当中，增加新的功能模块后可能会影响到的指标主要包括吞吐量、时延和丢包率等。测试工作主要针对这几个参数进行。

本节所有项目都进行两次测试，第一次测试的时候，软件系统中不编译移动 IPv6 模块，第二次则将移动 IPv6 模块编译进协议栈，从而将两次测试的结果进行量化比较。

4.1.2 测试环境

对系统性能的测试主要基于图 4-1 所示的环境进行。Router 为待测试设备，其硬件系统采用 ZTE GAR 路由器硬件平台，软件系统主要包括 IPv6 待测试协议栈软件，并配置两个网卡 port1 和 port2，由于待测协议栈目前仅提供了 NE2000 和 3C905 的软件驱动支持，目前建议选用这两种网卡；AX4000 为性能测试仪，配置了两个带 10B/100B 物理接口卡的测试功能块，分别标注为 port A 和 port B，

port A 功能块用于产生测试中所需要的各种流，而 port B 功能块用于收集和分析测试数据，当然在某些测试项目中，port B 也将发送部分流，同时 port A 则将兼顾部分收集和分析测试数据的任务。

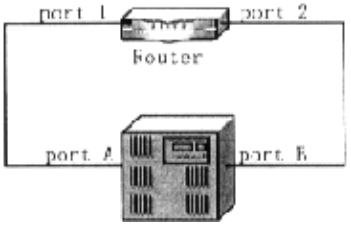


图 4-1　测试环境

4.1.3　测试方法与步骤

➢ 单端口吞吐量测试

路由器的两个网卡接口分别与测试仪的两个 100M 接口卡相连，然后正确配置路由器和测试仪的两个端口，使测试仪从一个端口发出的数据包可以经过路由器，回到另一个端口；测试仪向被测设备发送流量，指定帧长为 78。在测试仪分析软件的 GUI 界面上观测在指定的帧长大小情况下能达到的无丢包最大速率，并进录。

➢ 时延，也就是包处理时间的测试

路由器的两个网卡接口分别与测试仪的两个 100M 接口卡相连，然后正确配置路由器和测试仪的两个端口，使测试仪从一个端口发出的数据包可以经过路由器，回到另一个端口；在固定帧长情况下，测试仪以理论上的满速率向路由器发送流量，发送速率分别以理论上满速率的 100%，90%，80%，70%依次递减直到无丢包，观察分析软件界面显示的结果，并纪录。简单起见，包长指定了 78 字节。

4.1.4　测试结果

软件系统不编译移动 IPv6 模块时，在 ZTE GAR 路由器硬件平台上运行软件系统。对这个路由器的单端口最大吞吐量进行测试，测试过程中，统一采用长度为 78 字节的数据包。测量结果如图 4-2 所示。

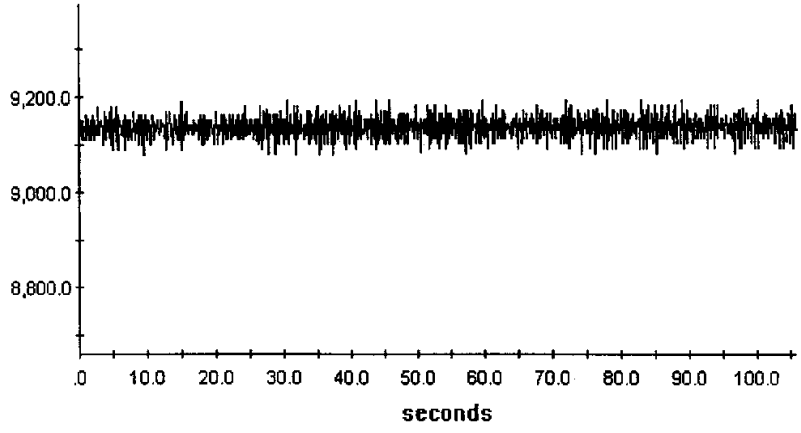


图 4-2　不编译移动 IPv6 时的单端口最大吞吐量

将移动 IPv6 模块编译进软件系统，在 ZTE GAR 路由器硬件平台上运行软件系统。对这个路由器的单端口最大吞吐量进行测试，测试过程中，同样采用长度为 78 字节的数据包。测量结果如图 4-3 所示。

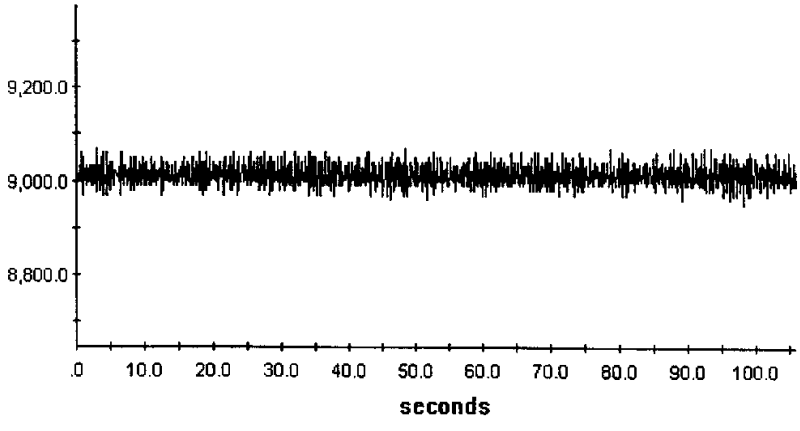


图 4-3　编译移动 IPv6 时的单端口最大吞吐量

从图 4-2 和图 4-3 可以看出，在原来的 IPv6 路由器上，包长 78 字节的情况下，单端口最大吞吐量约 9135p/s，而在系统引入了移动 IPv6 模块后，单端口最大吞吐量下降为 9013p/s，说明移动功能的引入导致路由器的转发性能下降。

$$
\begin{aligned}
P &= (\text{throughput}(t1) - \text{throughput}(t2)) / \text{thruput}(t1) \\
&= (9013 - 9135) / 9135 \\
&= -0.013355 \\
&\approx -1.34\%
\end{aligned}
$$

不编译移动IPv6模块的时候，在ZTE GAR路由器硬件平台上运行软件系统。

对这个路由器的时延进行测试，测试结果如图 4-4 所示。

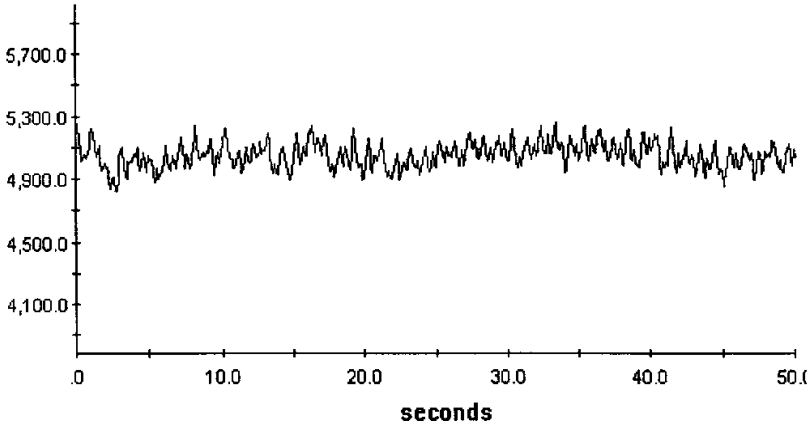


图 4-4 不编译移动 IPv6 的时延

将移动 IPv6 模块编译进软件系统后，在 ZTE GAR 路由器硬件平台上运行软件系统。对路由器的时延进行测试，测试结果如图 4-5 所示。

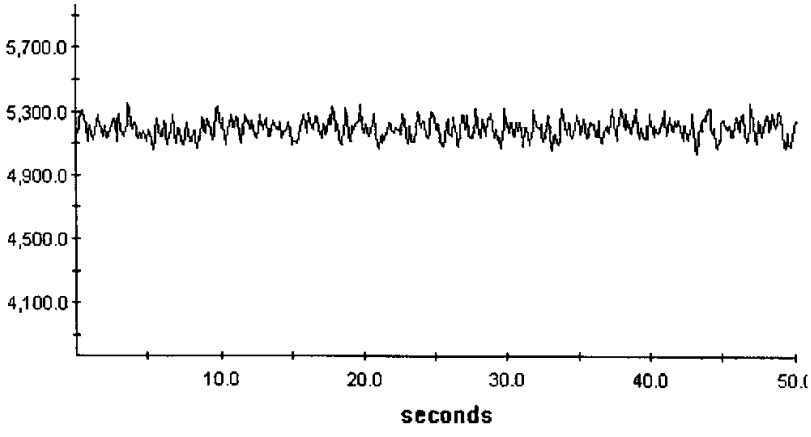


图 4-5 编译移动 IPv6 后的时延

从图 4-4 和图 4-5 可以看出，在运行原来的协议栈的时候，数据报平均延迟为 5055.82 微秒，引入移动 IPv6 功能后，数据报延迟变成了 5748 微秒。移动 IPv6 功能的引入，使得路由器在转发数据包的时候延迟加大，系统性能下降。

■ P =(delay(t1)－delay(t2))/delay(t1)

=（5748.00－5055.82）/5748.00

=0.1204210

≈12%

4.1.5 结论

经过对路由器性能测试的结果可以知道，将移动 IPv6 引入已有的 IPv6 协议栈软件后，转发速率降低了 1.3355%，包处理时间延长了 12%，系统性能下降。但是这对本文所要开发的一个演示系统来说还是能够接受的。

4.2 移动 IPv6 切换时延的测量

移动 IPv6 的切换延时是本章测试另外一个重点。移动 IPv6 是一种简单有效的网络层移动性解决方案，但它同时带来了切换的问题，随着用户移动性的增强，切换就成为影响性能的一个关键的因素。切换延迟时间也就成为反映移动 IPv6 协议工作性能的一个关键指标。本节致力于移动 IPv6 系统中切换时延的测量，通过测量切换过程所需要的时间，从一个侧面来考察移动 IPv6 软件协议栈的系统性能。

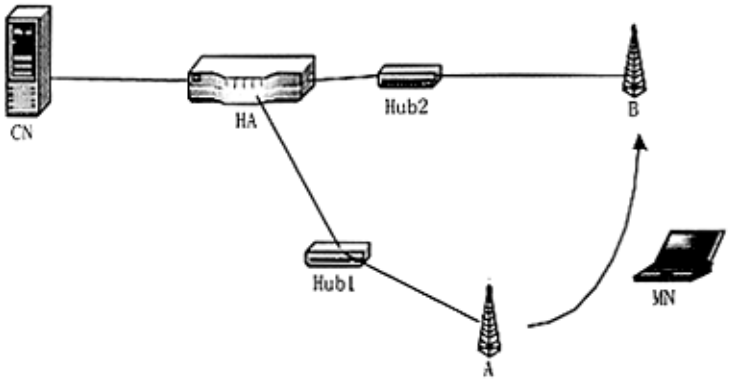


图 4-6 切换时延的测量

在上图所示的测试环境中，A 和 B 为无线接入点，HA 为支持移动 IPv6 家乡代理功能的 GAR 路由器。A 和 B 分别通过 HUB 与 HA 的不同端口相连。移动节点 MN 通过 A 或者 B 连接 HA 并和 CN 保持通信。假设 MN 首先与 A 相连，然后因为某些原因，断开与 MN 与 A 的无线连接，迫使其通过 B 接入网络，在这样的情形下，如果 A 和 B 分别位于不同的子网中，而移动节点在不同的接入点之间切换的时候，就相当于移动节点在不同的子网之间进行了移动。A 和 B 与路由器的连接方式决定了移动节点 MN 的在两个接入点之间进行切换的性质。根据 A 和 B 的不同位置，当移动节点在两个接入点之间的切换的时候，它们切换的实质分别如表 4-1 所示：

CASE	A 的位置	B 的位置	切换实质
1	家乡网络	外地网络	MN 从家乡网络移动到外地网络
2	外地网络	外地网络	MN 从一个外地子网移动到另一个外地子网
3	外地网络	家乡网络	MN 从外地子网移动到家乡子网

表 4-1　移动切换的三种情形

所谓切换延迟，也就是从移动节点通过原来的网络从通讯节点或者家乡代理接受到最后一个数据报到通过新的网络接受到第一个数据报之间的这段时间。在这个延迟时间里，数据报将无法传送。根据这个定义，可以设计相应的测试环境来对移动 IPv6 不同子网间的切换时间进行测量。

在一般情况下，如果要让一个节点的接入点从 A 变成 B，那么需要将这个节点进行移动，至少使其移动到 A 的覆盖边缘，这样需要频繁移动实验设备，增加了实验的复杂性。简便起见，我们首先将节点通过 A 接入网络，然后将接入点 A 断电，迫使移动节点另外寻找无线信道，在寻找到 B 以后再通过 B 接入网络。通过这种方法虽然也能够实现从接入点 A 到 B 的切换，但是也带来一个问题，那就是移动节点被动从 A 切换到 B 与主动从 A 切换到 B 的延迟时间是不同的。与移动节点先搜集备用接入点信息然后主动切换到那个接入点的切换过程相比，被动切换在切换发生时还没有完整的关于 B 的信息，在切换发生过程中，它还需要花费一定时间来寻找可用的无线信道，使得切换延迟时间大大增加。因此，在测量过程中需要考虑 AP 之间的切换所耗费的时间。

移动 IPv6 总的切换时间 T_{total} 可以分成两个部分：基本的 AP 切换所需要的时间 T_{AP} 和移动 IPv6 三层信令交互所需要的时间 T_{L3}。那么有

$$T_{total}=T_{AP}+T_{L3}$$

T_{total} 可以用上文描述方法方便地进行测量。采用这种方法虽然可以测量出总的切换时间，但是这个时间里面的 T_{AP} 是用来完成二层切换的，这个时间主要由移动节点的无线网卡和无线接入点的工作性能决定，不受移动 IPv6 模块性能的影响。所以 T_{total} 并不能很好的反映移动 IPv6 的软件性能。移动 IPv6 协议是网络层方案，移动 IPv6 软件主要工作在 IP 协议的第三层，与 T_{total} 相比，三层切换所用的时延 T_{L3} 更加准确地体现了软件的切换性能。因此，本节考察的重点是 T_{L3}。

$$T_{L3}=T_{total}-T_{AP}$$

T_{AP} 是单纯在两个 AP 之间进行二层切换所耗费的时间。当两个接入点 A 和

B 通过相同的 HUB 都连接到路由器的同一接口时，移动节点在位于同一个子网两个无线接入点之间进行切换所耗费的时间就是 T_{AP}。从而，T_{L3} 可以方便地估算出来。简明起见，实际实验过程中将 A 和 B 与位于家乡链路的 HUB 相连，从而统一置于路由器家乡链路上面，得到表 4-2 所示情形。

CASE	A 的位置	B 的位置	性质
4	家乡网络	家乡网络	单纯的二层切换

表4-2　移动切换的第四种情形情形

经过上文分析，本节的主要工作成为在所描述的四种情况下测试移动节点从A切换到B所需要的耗费的时间。在每一种基本情况下，切换所耗费的时间通过如下方法可以测量。

在通信节点上使用ping命令每隔一个固定的时间△T向移动节点发送一个数据包，在移动节点上使用tcpdump命令对来自通信节点的所有数据报进行监测并记录。

记录下移动节点接收到的每连续两个数据报之间的时间差$\triangle T_k$，这个差值可以由移动节点上的tcpdump程序计算并打印。在通信节点上利用ping发送报文和移动节点上tcpdump捕获报文的过程中，采取相关措施让移动节点从A切换到B。在对tcpdump记录进行分析的时候，切换发生的标志通常是某一个时间差突然变长，远超出它周围的数值。下文是从tcpdump产生的某个文件中节选出来的一段。

001716 3051::240:5ff:feae:b3d6 > 3052::204:76ff:fe71:3ae4: icmp6: echo request　(n-3)

004509 3051::240:5ff:feae:b3d6 > 3052::204:76ff:fe71:3ae4: icmp6: echo request　(n-2)

000911 3051::240:5ff:feae:b3d6 > 3052::204:76ff:fe71:3ae4: icmp6: echo request　(n-1)

4. 014133 3051::240:5ff:feae:b3d6 > 3052::204:76ff:fe71:3ae4: icmp6: echo request　(n)

000766 3051::240:5ff:feae:b3d6 > 3052::204:76ff:fe71:3ae4: icmp6: echo request　(n+1)

002365 3051::240:5ff:feae:b3d6 > 3052::204:76ff:fe71:3ae4: icmp6: echo request　(n+2)

上文中的每一行记录都是移动节点捕获到的一个数据包的部分信息。试对第n-2行进行分析：004509是移动节点接收到第n-2个数据包的时间与接收到第n-1个数据包的时间差，这个差值是0.004509秒。3051::240:5ff:feae:b3d6是报文源地址，3052::204:76ff:fe71:3ae4是报文目的地址。Icmp6: echo request是报文类型。通过观测可以发现，时间差值4.014133明显超出了它周围的数值，也就是说，在

接收到这个数据包之前相当长的时间里面没有接收报文，可以判断出在这段时间里面极有可能发生了网络切换。通过这种方法，再结合其他因素，便可最后判断出切换发生的时间。

假设在切换发生之前，时间差$\triangle T_i$的平均值为$E\triangle T_i$，那么在理想情况下，如果不发生切换的话，在接受到n－1的数据报之后的$E\triangle T_i$时间，应该接受到第n个数据报。而实际上移动节点在接收到第n－1个数据包之后的$\triangle T_n$才收到，相差的时间就是切换所带来的总的延迟T_{total}。

$$T_{total}=\triangle T_n-E\triangle T_k \qquad (k\neq n)$$

在实际测量过程中，对每一次延迟测试10次，记录每一次切换发生前后的接收到的相邻 报文的时间间隔。记$\triangle Tk$在第I次实验中的测量值为$\triangle T_{ki}$。

1. 当两个AP都位于家乡网络，移动节点在这两个AP之间切换的时候，测试数据如表4-3所示。

	$\triangle T_{n-4}$	$\triangle T_{n-3}$	$\triangle T_{n-2}$	$\triangle T_{n-1}$	$\triangle T_n$	$\triangle T_{n+1}$	$\triangle T_{n+1}$	$\triangle T_{n+1}$
h2h1	003568	002196	002276	003321	4.035591	000297	002812	000634
h2h2	003993	003633	005130	004168	4.141627	000770	002433	004043
h2h3	002660	000470	002180	000643	4.107991	002512	000477	003066
h2h4	003790	003681	004313	003957	3.956048	000478	003143	003067
h2h5	000457	001548	002178	001321	4.058772	000484	002648	002223
h2h6	000192	004312	000553	001946	5.357676	005367	005582	005834
h2h7	003634	004050	004005	005028	5.077855	004605	001946	003732
h2h8	001147	001716	004509	000911	4.014133	000766	002365	003958
h2h9	004821	004547	004231	004373	4.159669	000475	003280	000476
h2h10	001820	000465	001003	003441	4.160194	000506	002639	000555

表 4-3　同一子网内部的切换时延

这时候有

$$T_{total}(h2h)=E(\triangle T_{ni}-E(\triangle T_{ki})) \qquad (k\neq n, I=1,\cdots\cdots 10)$$
$$=E(\triangle T_{ni})-E(E(T_{ki}))$$
$$=E(\triangle T_{ni})-(E(T_{ki})$$
$$\approx E(\triangle T_{ni})$$

=4.3069556

在这种情形下测量出来的 T_{total} 就是 T_{AP}，所以有

T_{AP}=4.3069556

2. 断开家乡链路，mn 被迫接入与外地链路相连的 AP。测量结果如表 4-4。

	$\triangle T_{n-4}$	$\triangle T_{n-3}$	$\triangle T_{n-2}$	$\triangle T_{n-1}$	$\triangle T_n$	$\triangle T_{n+1}$	$\triangle T_{n+1}$	$\triangle T_{n+1}$
h2f1	003679	004342	004553	004184	7.883043	004938	001786	005948
h2f2	002663	002319	002156	002066	7.978839	004795	002426	005236
h2f3	002278	001931	002852	002360	6.893191	019898	020178	019756
h2f4	004913	004451	003986	004076	7.580645	005588	002083	005556
h2f5	004486	003762	003941	003822	7.845706	016763	020421	020050
h2f6	003833	004207	003953	003730	6.776443	015613	020300	020154
h2f7	004347	004003	003798	004397	6.739466	016167	019712	020149
h2f8	003984	004188	003897	003918	6.519942	018770	020090	019882
h2f9	000906	003768	002470	002240	6.730791	016825	020132	020104
h2f10	003823	003852	004284	004113	7.036051	019173	019974	020025

表 4-4 从家乡网络切换到外地网络的时延

计算移动节点从家乡切换到外地网络时候总的延迟，这时有：

$$T_{total}(h2f)=E(\triangle T_{ni}-E(\triangle T_{ki}) \qquad (k\neq n, I=1,\cdots\cdots 10)$$

$$=E(\triangle T_{ni})-E(E(T_{ki}))$$

$$=E(\triangle T_{ni})-(E(T_{ki})$$

$$\approx E(\triangle T_{ni})$$

$$=7.198412$$

完成从家乡到外地的三层切换所用的时间：

$$T_{L3}(h2f)=T_{total}(h2f)-T_{AP}$$

$$=7.198412-4.3069556$$

$$=2.86771664$$

$$\approx 2.87$$

3. 断开外地链路，移动节点被迫接入家乡接口所在的 AP，实验结果如表 4-5。

	$\triangle T_{n-4}$	$\triangle T_{n-3}$	$\triangle T_{n-2}$	$\triangle T_{n-1}$	$\triangle T_n$	$\triangle T_{n+1}$	$\triangle T_{n+1}$	$\triangle T_{n+1}$

f2h1	000898	003472	000687	002455	5.526725	008693	004402	004330
f2h2	004164	004187	004010	004115	5.292190	009208	002521	004305
f2h3	001542	002371	002391	000803	4.793622	008727	002516	004511
f2h4	000292	004107	000424	003242	5.240332	009103	004374	004543
f2h5	004869	004608	004554	004331	4.971143	008244	003996	004058
f2h6	004707	004337	004259	004679	5.242267	008598	004262	004269
f2h7	004384	005532	004572	004887	4.907013	008339	003504	004396
f2h8	004375	004768	004771	004835	5.096803	008224	002699	004527
f2h9	004547	004333	005254	004676	4.089533	015772	020035	008884
f2h10	004337	004353	004618	004254	4.815688	009364	004367	004466

表 4-5 从外地网络切换到家乡网络的时延

计算移动节点从外地切换到家乡所花费的总的时间，这时有：

$$T_{total}(f2h)=E(\triangle T_{ni}-E(\triangle T_{ki})) \qquad (k\neq n,\ I=1,\cdots\cdots\ 10)$$

$$=E(\triangle T_{ni})-E(E(T_{ki}))$$

$$=E(\triangle T_{ni})-(E(T_{ki})$$

$$\approx E(\triangle T_{ni})$$

$$=4.9978304$$

完成从外地到家乡的三层切换所用的时间：

$$T_{L3}(f2h)=T_{total}(f2h)-T_{AP}$$

$$=4.9978304-4.3069556$$

$$=0.6908748$$

$$\approx 0.69$$

4. 从一个外地到另一个外地的情形

	$\triangle T_{n-4}$	$\triangle T_{n-3}$	$\triangle T_{n-2}$	$\triangle T_{n-1}$	$\triangle T_{n}$	$\triangle T_{n+1}$	$\triangle T_{n+1}$	$\triangle T_{n+1}$
f2f1	004915	005054	004638	004969	6.658164	019752	020065	020111
f2f2	004477	004983	004538	004685	7.820326	019276	020093	019935
f2f4	000567	003355	002928	000958	7.862725	004987	001443	005879
f2f5	004449	004294	004431	005025	7.021223	019430	020222	019596
f2f6	004418	004573	004806	004613	7.620042	018947	019971	020010

f2f7	001387	002544	002554	000967	6.900880	019446	020002	019726
f2f8	004630	004754	004898	004875	7.382839	004903	001863	005402
f2f9	005272	004758	004714	005189	6.533582	019337	020129	019620
f2f10	004636	004671	004721	005045	6.393586	019385	019968	019998
f2f17	004713	004549	005022	005139	7.048454	019307	019953	020362

表 4-4 从一个外地网络切换到另一个外地网络的时延

计算当移动节点从外地移动到另一个外地所花费的总的时间：

$$T_{total}(f2f)=E(\triangle T_{ni}-E(\triangle T_{ki})) \qquad (k\neq n, I=1,\cdots\cdots 10)$$

$$=E(\triangle T_{ni})-E(E(T_{ki}))$$

$$=E(\triangle T_{ni})-(E(T_{ki})$$

$$\approx E(\triangle T_{ni})$$

$$=7.1241821$$

完成从外地到家乡的三层切换所用的时间：

$$T_{L3}(f2f)=T_{total}(f2h)-T_{AP}$$

$$=7.1241821-4.3069556$$

$$=2.8172265$$

$$\approx 2.82$$

综合上文分析，在本文所组建的实验环境当中，当移动节点从家乡网络移动到外地的时候，网络层的切换时延是 2.87 秒，当从一个外地移动到另外一个外地的时候，网络层的切换时延为 2.82 秒，而当从外地回到家乡的时候，网络层的切换时延很短，只需要 0.69 秒。在采用移动 IPv6 单一切换机制进行切换的时候，这样的切换性能已经是比较理想的。

第五章 基于移动IPv6技术的无线多媒体演示系统

2003 年 5 月 21 日，中兴通讯利用本文工作的成果，结合其网络事业部的 WLAN 接入设备，成功实现移动 IPv6 技术在 WLAN 环境下的多媒体业务演示。据悉，这是国内设备商用自主开发的设备第一次演示移动 IPv6 技术。演示系统通过无线局域网接入点设备 WAS－W100A，将具有 WLAN 功能的移动终端接入到由 IPv4/v6 双栈路由器组成的 IPv6 试验网络。移动终端通过试验网络的视频服务器在线播放多媒体影像。当移动终端在属于两个子网的无线接入点设备之间切换时，其与视频服务器的通信不中断，多媒体影像仍然能够连续地播放。通过演示系统，更好地展示和宣传移动 IPv6 技术。本章着重阐述这个演示系统的拓扑结构，并详细介绍了系统中各主要部件的选型。

5.1 移动 IPv6 无线多媒体演示系统

搭建无线局域网环境，可以演示各种 IPv6 的关键技术，并进一步演示 IPv6 的移动性，展示 IPv6 的优越性，还可以在无线环境下评测移动 IPv6 的功能、性能，并为下一步深入地研究 IPv6 的移动性提供试验平台。下图为无线环境下的移动 IPv6 多媒体演示系统的网络拓扑图。

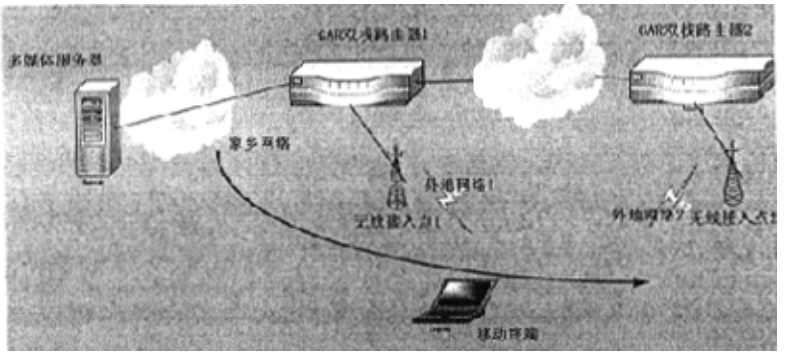


图 5-1 演示环境示意图

在这个示意图中，主要包括以下设备：两个路由器是开发成功的 GAR 双栈路由器；两个无线接入点是中兴通讯的 WLAN 设备：WAS-W100A；多媒体服务器上运行支持 IPv6 的操作系统 FreeBSD 和支持 IPv6 的在线视频服务器程序；移动终端为装有无线局域网的网卡笔记本电脑，其上运行支持 IPv6 的操作系统 FreeBSD、支持移动 IPv6 的 KAME 协议栈及支持 IPv6 的在线视频播放客户端程

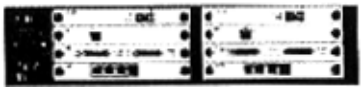
序。

5.2 主要部件的选型

1. GAR 双栈接入路由器

演示环境中所使用的两个路由器是中兴通信技术中心研究部成功开发的双栈接入 GAR 路由器。它能够支持地址自动配置机制、完善的过渡机制、多种 QoS 策略、IPSec、多种路由协议等，标志着中兴通讯的路由器产品开始全面的支持未来互联网的升级和换代。本文的工作在 GAR 路由器原来协议栈的基础上，添加了新的功能特性，在 IP 层提供了对网络层移动性的支持。支持移动 IPv6 功能的 GAR 双栈接入路由器是移动 IPv6 多媒体演示系统的核心和主要构建部件。

GAR 侧视图



GAR 后视图

图 5-2 GAR 视图

2. FreeBSD[27]和 KAME[26]

虽然 GAR 双栈接入路由器也能配置成移动节点，完成移动节点的各种功能。但是移动节点需要大量的上层软件来提供对各种应用和业务的支持，而 GAR 的开发和运行都是基于 Vxworks 实时嵌入式操作系统进行的，在这种操作系统下能够用于音频和视频演示的应用程序很少，同时也没有提供演示所需要的图形化的用户界面。因此，移动节点上使用的操作系统和协议栈需要另外考虑。

目前世界上有很多组织或者机构在进行 IPv6 的研究，并且在不同操作系统上开发出一些的实验系统，而 KAME 协议栈是这些系统中比较著名的一个。它是由日本 WIDE(Widely Integrated Distributed Environment)组织从事 IPv6/IPSec 协议栈开发的项目组开发的，其移动 IPv6 功能正在开发和完善当中。KAME 协议栈支持的操作系统包括 FreeBSD，NetBSD，OpenBSD 等。而在 BSD 系列操作系统下有着大量的应用程序可供选择和使用。这些操作系统和 KAME 协议栈都是开放源代码软件，可以从互联网免费下载。

综合各种因素，演示系统选择 FreeBSD 作为移动节点使用的操作系统。FreeBSD[27]是一个同时支持 Intel 处理器体系结构 X86 和 DEC Alpha 处理器体系结构的 免费的操作系统，主要特点在于其高性能和高可靠性，用户可以从互联网上下载获得。FreeBSD 中使用了另一个著名的自由软件 XFree86，来提供工业标准的 X 视窗系统 X11R6,在 X 上可以运行多种图形界面软件提供方便用户使用的图形界面和应用软件。FreeBSD 捆绑了很多系统工具作为基本系统的一部分，此外 FreeBSD 网站中还提供了大量的开放软代码的第三方应用程序共选择使用，基本上满足了构建一个演示系统的需要。

3. MPlayer[27]

Mplayer 是一个 Linux 操作系统下运行的媒体播放工具，它也可以运行在其它的许多 UXIX 操作系统下，包括 FreeBSD。它也可以运行在非 X86 平台上面。它能够播放的媒体类型有 MPEG, VOB, AVI, OGG/OGM, VIVO, ASF/WMA/WMV, QT/MOV/MP4, FLI, RM, NuppelVideo, yuv4mpeg, FILM, RoQ, PVA。也可以利用它来观看 VideoCD, SVCD, DVD, 3ivx, RealMedia 和 DivX 格式的电影。MPlayer 的另外一个优点是支持大量的驱动。

4. 服务器的配置

服务器需要升级，成为能够支持 IPv6 服务器，并需要开启 FTP6 服务。

5. 无线 AP

无线 AP 用在无线局域网热点地区，用以覆盖无线局域网的接入终端，为用户提供无线局域网接入功能，并完成简单的无线用户管理和对无线信道的动态分配。

中兴通信无线局域网的系列产品，涵盖了从无线接入网关到用于终端设备的无线网卡。我们的演示系统中上连到 IPv6 路由器的无线接入点设备就选用中兴通信的设备 W100A。W100A 系统硬件包括无线网口、一个以太网口和一个 RS232 配置口。系统软件主要由支撑子系统、基本业务子系统、扩展业务子系统和配置子系统组成。W100A 具备公共无线局域网运营所要求的各项功能，可以满足运营商部署无线局域网的要求。

6. 无线网卡

移动节点上使用支持 802.11b 无线网卡接入 IPv6 网络。现在市面上有众多无

线网卡可供选择，但其大多都只有 Windows 的驱动程序。因我们的移动节点上要运行支持移动 IPv6 的软件，而 Windows 操作系统目前无此功能。因 FreeBSD 和 NetBSD 的开放源码性，最新的 FreeBSD 和 NetBSD 操作系统有较多支持无线网卡的驱动程序，及不少 IPv6 的应用程序。经调研选择使用 dlink[31]无线网卡。所以在移动节点上的软件主要构成是：FreeBSD＋无线网卡驱动程序＋KAME 协议栈＋IPv6 应用程序。

5.3 演示环境的构建

构建演示系统所需的软硬件资源

1. 硬件资源

至少两台 IPv6 路由器，用于搭建两个不同的 IPv6 子网；一台笔记本电脑；两个支持 802.11b 的无线局域网接入点设备（AP），用于对移动节点的无线接入；两个支持 802.11b 的 PCMIA 无线接入网卡，用于移动节点通过无线信道接入 IPv6 网络。

2. 软件资源

支持移动功能的 IPv6 协议栈软件，运行在 IPv6 子网的路由器上；支持无线局域网网卡的 FreeBSD 或 NetBSD 操作系统，KAME 协议栈源码，这些软件运行在移动节点上；无线接入点的相关软件。

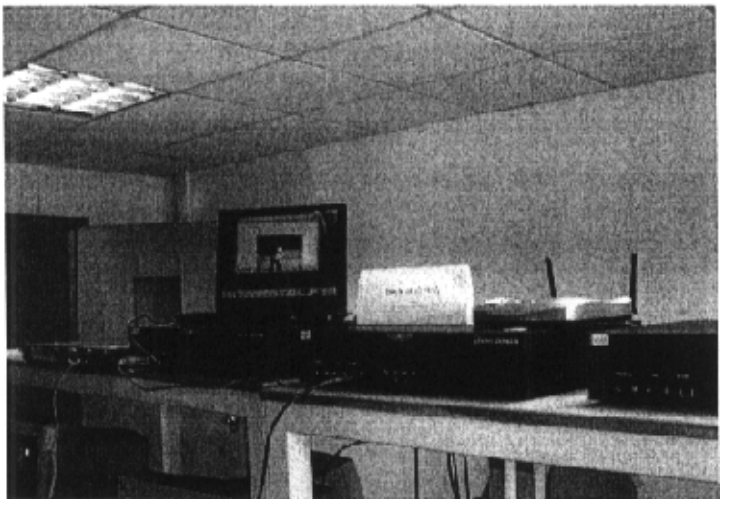


图 5-3 实际演示系统环境

实际演习系统如图 5-3 所示。如果所示，在两个 GAR 双栈路由器上配置好

各种参数，使整个演示系统为一纯 IPv6 网络。GAR1 路由器有三个接口，也就对应着三个不同 IPv6 子网网络，一个作为移动终端的家乡网络，另外两个作为移动终端的外地网络,其中一个接口作为 WAS-W100A 的上连接口。GAR2 的两个接口分别位于不同 IPv6 子网网络，都为移动终端的外地网络,其中一个接口作为 WAS-W100A 的上连接口。多媒体服务器通过 IPv6 的地址自动配置机制得到全局的 IPv6 地址。移动终端配置好有关移动 IPv6 的参数后，通过视频服务器的全局的 IPv6 地址用在线视频播放客户端程序在线播放视频，当移动终端通过 WAS-W100A 在不同的 IPv6 子网间漫游切换时，其与视频服务器的通信不中断，图 5-4 中小女孩载歌载舞的多媒体影像保持连续播放且抖动极小，形象地演示了 IPv6 移动性技术。

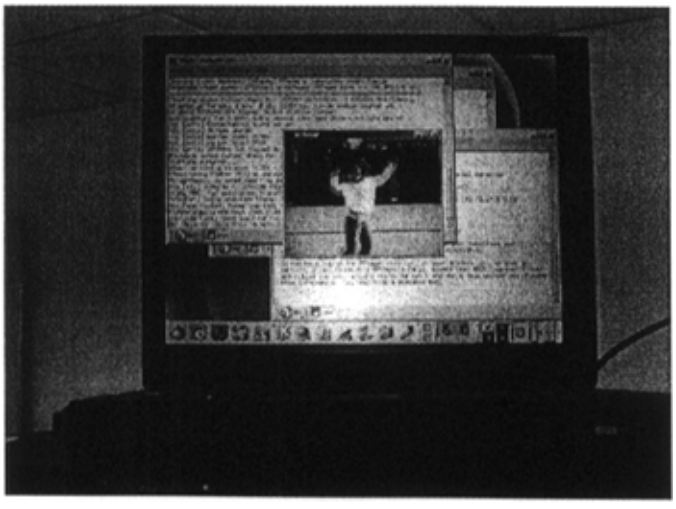


图 5-4 移动终端上的演示画面

移动 IPv6 技术作为 IPv6 网络的移动性解决方案，使得移动终端能够以固定的全局 IP 地址接入互联网，当终端在不同的子网、不同的接入技术之间漫游切换时，仍能用该 IP 地址和网络上的其个网络他设备进行通信。该技术和 IPv6 庞大的地址空间相结合，能实现终端设备与整的网络随时随地在线连接，从而保证了移动终端对于 IP 呼叫、多媒体、移动 QQ、互动游戏、电子商务等业务的支持，并将对互联网实现运营产生深远影响。

5.4 小节

本章讲述了我们所构建的无线环境下的移动 IPv6 演示系统，介绍了它的各个组成部件，包括硬件的选型与软件的配置。移动 IPv6 技术作为 IPv6 网络的移动性解决方案，使得移动终端能够以固定的 IP 地址接入互联网，当终端在不同的子网、不同的接入技术之间漫游切换时，仍能用该 IP 地址和网络上的其它设备进行通信。该技术和 IPv6 庞大的地址空间相结合，能实现终端设备与整个网络的随时随地在线连接，具有广泛的应用前景。

结 束 语

随着网络技术与便携式终端的不断发展，在 IP 网络中实现对移动性的支持变得越来越重要，移动 IP 为移动主机提供了基于 IP 网的透明业务传输，而移动 IPv6 则是未来移动通信社会发展的必然趋势。本论文旨在研究 IPv6 的移动性，建设网络平台，并进行一些实验，主要包括以下工作：

1. 确定了移动 IPv6 要实现的功能集，完成了移动 IPv6 功能模块的划分，并给出了系统设计方案。
2. 在对协议和 KAME 协议栈进行对照分析的基础上，采用代码移植的方法，完成了移动 IP 协议实时嵌入式操作系统 Vxworks 下的实现工作。这也是本文最主要的工作。
3. 立足于现有的条件，对协议实现的基本功能进行了测试，并具体给出测试环境、测试方法和测试结果。测试表明，我们已经实现了一个移动 IPv6 的基本功能集。
4. 对移动 IPv6 协议栈的性能进行了测试，对测试数据进行了详细分析和讨论。
5. 在无线环境下实际构建了一个移动 IPv6 多媒体演示环境，进行技术演示。

本文的后继工作主要包括以下几个方面：

1. 跟踪对移动 IPv6 技术的发展，跟踪最新草案进展情况。
2. 对现已完成的移动 IPv6 软件实现进行维护和更新，继续完善软件功能。
3. 对移动 IPv6 的安全、QoS、快速切换以及实际应用推广等问题继续进行较为深入的研究。

致 谢

在三年的硕士研究期间，导师杨庚教授对我的学业进行了精心指导，他渊博的专业知识、深厚的理论功底、孜孜不倦的治学精神、亲切待人的态度，都给我留下了深刻的印象。也正是在他的指引之下，我开始关注和研究移动 IP 技术。在此向导师表示衷心的敬意与谢意。

本论文的开发工作主要在中兴通讯技术中心研究部 IPv6 项目组展开，项目组为本文的工作提供了良好实验条件和工作环境。感谢项目组张洪渊博士和周庆标博士对我的悉心指导，也感谢项目组其他成员对我的关心和帮助，他们是项目经理彭海清、王忠博士、王雅琳博士、魏国富博士、李明正、田东旭、罗志丹、王芳和童进博士。

感谢已经毕业的师兄师姐，他们出色的工作给了我很多启发。感谢同门以及师弟师妹们所给予的帮助，跟他们的讨论和交谈使我受到很多启发。

我的家人，从来都是我的坚实后盾，愿与他们分享我的一切成果与快乐。

作者已发表论文

[1] 张连生，张纪金，朱健，杨庚：局域网环境中移动 IP 平台的实现与分析，计算机应用研究，2003.03：150-152

[2] 周佩聆，张连生，杨庚：蜂窝移动 IP 网络越区切换问题解决方案的分析研究，计算机应用研究，2003.03：66-68

[3] 朱健，杨庚，张连生，张纪金：AAA 技术在移动 IP 中的应用，微型机与应用，2002.10：36-37

参 考 文 献

[1] Dave Johnson, Charles Perkins, Jari Arkko, Mobility Support in IPv6, draft-ietf-mobileip-ipv6-2, July, 2003

[2] J. Arkko, V. Devarapalli, ENST Bretagne, Using IPSec to Protect Mobile IPv6 Signaling between Mobile Nodes and Home Agents, draft-ietf-mobileip-mipv6-ha-IPSec-06, June 30, 2003

[3] Samita Chakrabarti, Erik Nordmark, Extension to Sockets API for Mobile IPv6, draft-ietf-mip6-mipext-advapi-00.txt, February, 2004

[4] S. Deering, R. Hinden, Internet Protocol, Version 6 (IPv6) Specification, RFC2460, December 1998

[5] T. Narten, E. Nordmark, W. Simpson, Neighbor Discovery for IP Version 6 (IPv6), RFC2461, December 1998

[6] R. Hinden, S. Deering, IP Version 6 Addressing Architecture, RFC2373, July 1998

[7] R. Hinden, M. O'Dell, S. Deering, An IPv6 Aggregatable Global Unicast Address Format, RFC2374, July 1998

[8] S. Thomson, T. Narten, IPv6 Stateless Address Autoconfiguration, RFC2462, December 1998

[9] A. Conta, S. Deering, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification, RFC2463, December 1998

[10] J. McCann, S. Deering, J. Mogul, Path MTU Discovery for IP version 6, RFC1981, August 1996

[11] S. Kent, R. Atkinson, Security Architecture for the Internet Protocol, RFC2401, November 1998

[12] S. Kent, R. Atkinson, IP Authentication Header, RFC2402, November 1998

[13] S. Kent, R. Atkinson, IP Encapsulating Security Payload (ESP), RFC2406, November 1998

[14] M. Crawford, Router Renumbering for IPv6, RFC2894, August 2000

[15] R. Gilligan, S. Thomson, J. Bound, W. Stevens, Basic Socket Interface Extensions for IPv6, March 1999

[16] W.Richard Stevens, TCP/IP Illustrated Vollume1:The Protocols, Addison Wesley, 1994

[17]Gary R.Wright, W.Richard Stevens, TCP/IP Illustrated Vollume1:The Implementation, Addison Wesley, 1994

[18]M. Crawford, Transmission of IPv6 Packets over Ethernet Networks, RFC2464, December 1998

[19]Andrew S. Tanenbaum: Computer Networks , 3rd Ed Prentice Hall , Inc. 1996

[20]Hesham Soliman, Claude Castelluccia etc: Hierarchical Mobile IPv6 mobility management (HMIPv6), draft-ietf-mobileip-hmIPv6-07.txt, October, 2002

[21]谢希仁,计算机网络，电子工业出版社，1994

[22]沈金龙：计算机通信网，东南大学出版社，1995

[23]孔祥营，柏桂枝,嵌入式实时操作系统 Vxworks 及其开发环境 Tornado，中国电力出版社, 2002

[24]阚志刚，黄晖，马建：移动 IPv6 概述，中兴通讯技术，2002.6

[25]阚志刚等：移动通信的基石----移动 IP，移动通信，2001.1

[26]http://www.kame.net

[27]http://www.freebsd.org

[28]http://www.ietf.org

[29]http://www.ipv6.net.edu.cn

[30]http://www.ncn.cn

[31]http://www.dlink.com