

基于 SDL 语言的移动 Ad-hoc 网路由协议的设计和验证

潘红艳^{1,2} 于全²

(中国科学技术大学研究生院 北京100039)¹ (中国电子设备系统工程公司研究所 北京100039)²

摘要 论文简要阐述了 SDL 的基本概念及使用它开发通信协议软件的优点,并给出在移动 Ad-hoc 网路由协议开发中采用 SDL 语言对路由协议软件进行设计和验证的实例,为通信协议的设计和验证提供了一种有效的方法。

关键词 SDL,移动 Ad-hoc 网,设计,验证

Design and Verification of a Routing Protocol for Mobile Ad-hoc Network Based on SDL

PAN Hong-Yan^{1,2} YU Quan²

(Graduate School, University of Science & Technology of China, Beijing 100039)¹

(Institute of China Electronic System Engineering Corporation, Beijing 100039)²

Abstract This paper briefly reveals the basic concept of SDL and advantages to develop communication protocol softwares using SDL, and gives an application instance of designing and verifying routing protocols based on SDL in developing mobile Ad-hoc network routing protocol. It provides an effective method for communication protocol designing and verifying.

Keywords SDL, Mobile Ad-hoc network, Design, Verification

1 引言

移动 Ad-hoc 网(Mobile Ad-hoc Network; MANET)是一种新型的无线通信网络,它采用先进的网络通信协议、动态的网络拓扑结构、多跳转接技术和自组织、自恢复网络技术^[1],具有响应时间短、组建方便、覆盖面广、可以进行动态网络控制和路由选择的特点和优点,可广泛应用于各种军事战术数据通信和民用无线数据通信领域。

SDL 语言主要用来描述实时系统行为方面的规格,特别适合开发一些具有实时和并发特点的通信系统。使用 SDL 语言描述、分析通信协议软件,可以大大提高软件开发效率,提高软件的可靠性。

文章介绍了 SDL 语言在移动 Ad-hoc 网路由协议设计和验证中的具体应用过程,为通信协议的设计和验证提供了一种有效的方法。

2 SDL 语言简述

SDL (Specification and Description Language)^[2,3]是1976年至1992年由 ITU-T 发展和标准化的一种 FDT。至今已颁布几个更新版本(SDL-2000是最后一版),主要顺应了近几年软件工程和编程语言发展的面向对象方法的采用。SDL 的数学模型是扩展有限状态机(EFSM),是一种高层泛用型的用于事件驱动、实时和通信系统的描述语言;电信系统和协议是它主要应用领域之一。

SDL 有图形和文本两种表示形式,SDL/GR 和 SDL/PR。它既可以用来对系统的功能进行说明,也可以描述系统的内部结构和行为,因而应用范围很广,可以用于协议工程的各个阶段。SDL 采用一种层次结构来对系统进行描述和说明,结构和功能被清楚地划分。系统可以分为多个功能块,功能块表示系统内部功能相对独立的几个模块,它们之间通过

信道相连。根据系统规模的大小,每个功能块还可以分为子功能块或进程。与传统 FSM 类似,SDL 进程具有有限数量的被有限数量跃迁连接着的宏控制状态。通过一个输入信号(也可称使能条件)的接收、事先设置的定时器的定时或当前状态变量(连续信号)值所指定条件的确认,跃迁能够被触发。

在 SDL 中,定义了一些基本的数据类型和操作,以及构造新类型的机制,使得 SDL 拥有和 C/C++ 语言一样的数据操作能力,因此它可以用于系统的设计和实现。

SDL 的使用和相关的支持工具是分不开的。比较出色和流行的是瑞典 Telelogic 公司开发的 Telelogic Tau^[4]工具软件,可用于 SDL 系统的设计、说明、分析、验证、代码生成和系统测试,作者在设计和验证中采用了该工具。在 Telelogic Tau 中,通过对由控制状态和数据值构成的相应的有限状态机的穷举或局部模块的检查,进行系统静态和动态特性的验证。

3 移动 Ad-hoc 网介绍

我们研究的移动 Ad-hoc 网是由一组带有无线收发装置的移动 Ad-hoc 单元(Packet Radio Unit, PRU)组成的一种多跳自组织的通信网络。PRU 由无线电台、天线和分组数据终端协议模块组成。

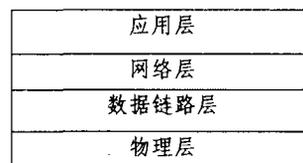


图1 Ad-hoc 网的网络体系结构

为了更好地说明移动 Ad-hoc 网路由协议的通信流程和设计原则,我们先对移动 Ad-hoc 网的协议层次结构进行描述,如图1所示。移动 Ad-hoc 网的协议体系结构和 ISO OSI 七

层协议体系结构相似,不过它只包含了 OSI 的其中四层,即应用层、网络层、数据链路层和物理层。其中数据链路层还分为 RLC(radio link control)和 MAC(media access control)两个子层。体系结构中各层所要完成功能分别为:

物理层的功能:提供数据传输设备之间的物理接口;封装数据链路层的协议数据单元;对数据链路层的传输头和协议数据单元计数;对数据提供前向纠错(FEC),交织;形成物理层帧同步。

数据链路层的功能:封装网络层的协议数据单元成帧;确保网络层数据在无线信道的可靠传输,提供帧检错;对链路层的协议数据单元实施监控;对链路层的协议数据单元实施优先级流量控制;实施网络接入控制。

网络层的功能:封装上一层数据报;为传输的数据提供合适的路径路由;通过路由信息的交换来反应网络拓扑的动态变化;传输无线网络管理信息。

应用层的功能:提供面向用户的协同应用服务。

在移动 Ad-hoc 网各层协议中,路由协议是整个网络体系结构的核心。我们研究的移动 Ad-hoc 网路由协议是一种基于按需方式进行的路由算法,即在保证路由的基础上尽可

能地减少路由控制信息的广播,从而提高实际数据传输效率;同时它又引入一定的主动路由机制来提高节点对网络拓扑的反应速度。路由协议主要由四部分组成:路由请求、路由维护、分组转发、路由公告。

4 路由协议设计和验证

4.1 协议设计

通过对移动 Ad-hoc 网协议的分析,作者认为该协议实体功能相对独立,协议实体可以由若干个状态机组成,其功能由外部和内部离散事件驱动,采用 SDL 语言对该协议进行设计和建模是非常适合的。SDL 提供了一种简化复杂系统结构的机制,将系统划分为不同的层次,同时通过对系统或构件的接口进行形式化描述,直观地确定所描述的性质的接口的行为,帮助设计人员构造一个比较简洁的系统概念模型,建立并保持系统概念的完整性。整个移动 Ad-hoc 网协议系统的设计如图2所示,系统内部由一个功能模块 MANET(protocol)构成。功能模块和系统环境之间通过信号相互连接,通过接收外部信号,激发相应的操作。在模块内部,还有更进一步的详细描述。

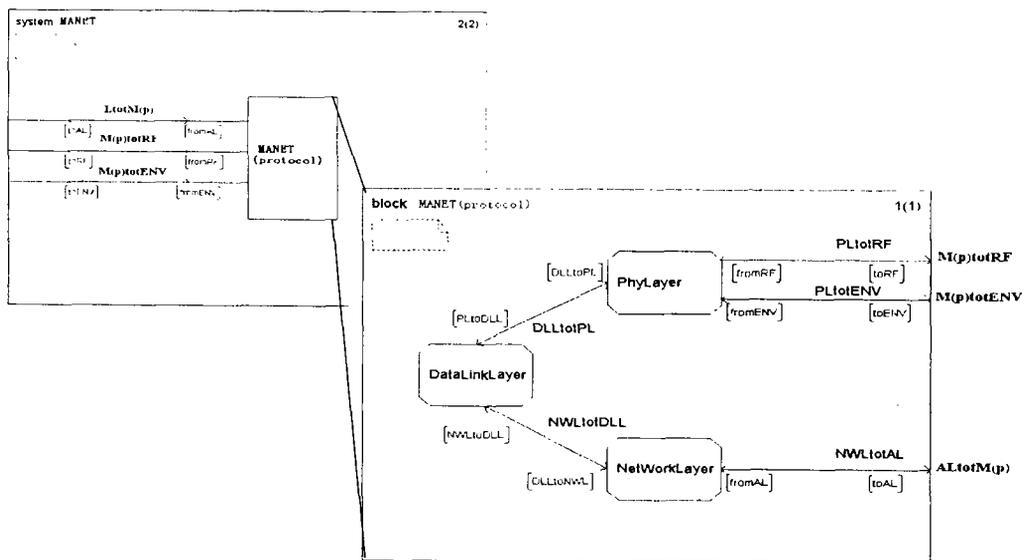


图2 系统规格及功能模块

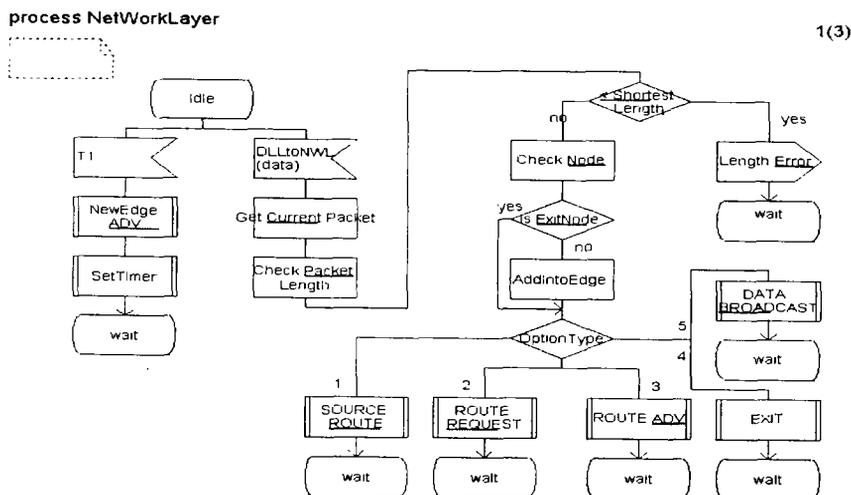


图3 网络层接收进程规格描述

从图2可以看出功能模块中各个处理进程之间的信号连接和相互作用关系。在 MANET(protocol)模块中包含三个处

理进程:NetWorkLayer 进程、DataLinkLayer 进程和 PhyLayer 进程。本文仅重点讨论 NetWorkLayer 进程。NetWorkLayer 进程的主要工作流程如下:

(1)进行网络层初始化。网络有关的参数配置。包括:路由公告间隔、等待确认时间、路由泛洪请求次数、重传次数、编码模式等;网络层定时器、队列、缓存空间、路由表的初始化等。

(2)进行数据链路层数据分组接收。首先取出当前分组,进行分组长度检查,如果小于最短分组,提示出错;如果正确,维护节点路由信息,接着如果是源路由类型,则进行源路由分组处理;如果是路由请求类型,则进行路由请求分组处理;如果是路由公告类型,则进行路由公告处理;如果是退出类型,则进行退出节点分组处理;如果是数据广播类型,则进行数据广播处理。网络层接收进程规格描述见图3所示。

(3)进行应用层数据分组发送。首先取出当前分组,如果是广播分组,按照广播接力传送;如果是有目的节点的分组,如果是自己,提示出错并删除此数据分组;接着如果是退出节点,则删除此数据分组。如果该目的节点在本地网络,则构造数据分组传送;如果不是,则构造网关数据分组传送。

图3形式化地描述了 NetWorkLayer 进程中收进程在不同状态下,受到信号激励后所产生的状态跃迁,以及相应的操作,直观地表现了进程中可执行的条件、过程、结果输出,即系统的行为。

4.2 协议验证

用 SDL 语言描述协议系统就象使用某种编程语言编程

一样,它需要对系统的描述进行正确性验证。协议的正确性验证试图在协议开发的前期最大限度地检测和纠正协议错误和缺陷,包括死锁、活锁、数据操作错、不可执行的行动、协议外部性能不符合服务要求等。

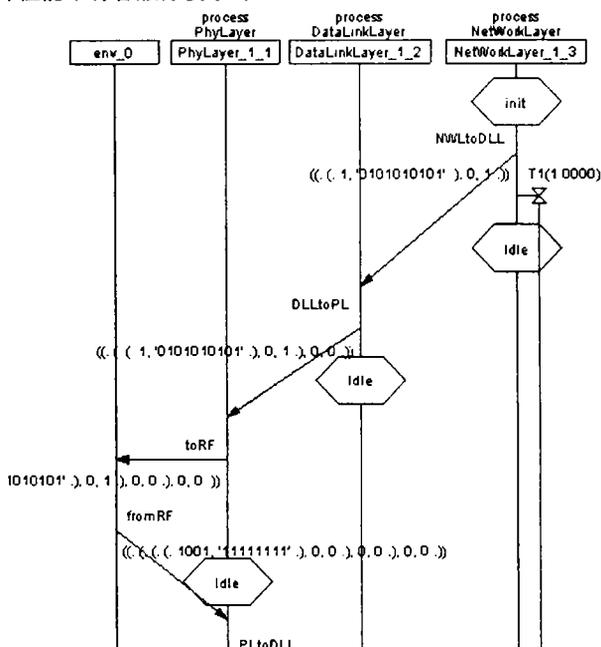


图4 MSC 记录系统运行轨迹

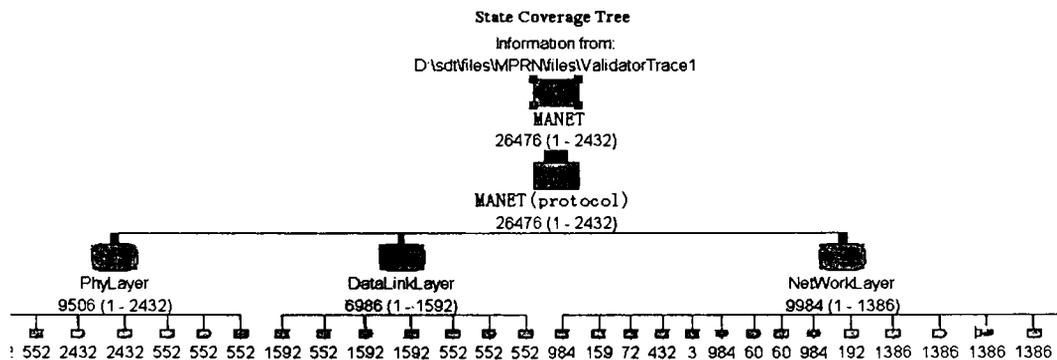


图5 系统状态覆盖树

因此,在对协议实现形式化的、抽象的描述说明,即设计建模完成后,需要对协议进行仿真验证。首先对 Tau SDL/GR 编辑器中设计好的协议进行语法和语义分析,如果分析过程正确结束就可以产生仿真代码,然后把仿真代码载入 Tau 提供的 Simulator 中进行整个系统的仿真。仿真是在用户控制下,通过送信号、跟踪执行系统,检查系统、进程和变量,还可以采用 SDL/GR 形式,或使用 MSC(Message Sequence Chars)记录系统仿真的轨迹,如图4所示。MSC 提供了一种理解动态行为和期望行为来验证系统的方法。最后,使用 Tau 提供的 Validator 进行系统验证,检查系统运行时的情况,看存不存在死锁等设计错误。Validator 也可以采用 SDL/GR 形式,使用 Navigator 工具探测系统行为树,或使用 MSC 记录系统仿真的轨迹。图5是 Validator 通过对系统进行自动状态空间探测后,产生的系统状态覆盖树。覆盖树显示了系统的框图结构,表明系统中各构件的执行情况。在每个进程框图(NetWorkLayer、DataLinkLayer 和 PhyLayer)分支中,可以看到所有可能的状态、信号输入和输出及相应操作构件,还可

得到系统各构件的执行百分比。图5中系统各构件均已执行过,执行百分比是100%。

结论 通过对一种移动 Ad-hoc 网路由协议的设计和验证,表明 SDL 特别适合开发一些具有实时和并发特点的通信系统和网络协议。使用 SDL 在不同抽象层次上描述、分析通信系统和网络协议软件,可以有效地分解和降低系统的复杂性,设计出功能上正确可靠,逻辑上完整一致和易于有效实现的通信协议软件,并且使整个设计和实现的过程更加规范化和自动化,从而提高通信协议软件的生产率、可靠性和可维护性。

参考文献

- 1 Tanenbaum A S. 计算机网络. 清华大学出版社,2000
- 2 ITU-T, Recommendation Z.100 (11/99) Specification and description language (SDL)
- 3 Ellsberger J, et al. SDL Formal Object-oriented Language for Communicating Systems. Great Britain:Prentice Hall,1997
- 4 Telelogic Tau4. 3 Users Guide, Telelogic AB,2001