基于 NuSMV 的 AADL 行为模型验证的探究

刘 博,李蜀瑜
(陕西师范大学 计算机科学学院,陕西 西安 710062)

摘 要:鉴于模型在软件系统开发中日趋重要的地位和 AADL 模型在嵌入式软件建模中的良好应用前景,为了在嵌入式软件系统开发前期保证 AADL 模型的质量,提出了一种基于模型测试的 AADL 架构和 NuSMV 模型的验证方法。文中首先对当前的 AADL 发展情况作简单介绍,然后对 NuSMV 验证模型的结构作大致分析,在随后的文章中对 NuSMV 的验证过程作详细的介绍。与此同时,使用具体的汽车巡航控制系统作为实例进行具体分析。文中通过测试用例执行输出进行验证来判断该方法的正确性。

关键词:嵌入式构件分析与设计语言;AADL 集成开发环境;行为模型;NuSMV 验证方法
中图分类号:TP31 **文献标识码**:A **文章编号**:1673-629X(2012)02-0110-04

AADL Behavior Based on NuSMV Model Validation of Inquiry

LIU Bo, LI Shu-yu
(Dept. of Computer Science, Shaanxi Normal University, Xi'an 710062, China)

Abstract: As models take important part in software development and AADL can describe embedded software effectively, a method for AADL model and NuSMV model testing is put forward to insure the quality of the models in early step of software developing. Proposed a test based on the AADL architecture model and NuSMV model validation methods. The development of the current AADL is introduced briefly, then the structure of the NuSMV model validation is analysed, in the subsequent article in the verification process of the NuSMV is described in detail. At the same time, the use of specific automobile cruise control system as the example is analysed. At last, verify the correctness of the method through an example.

Key words: AADL; OSATE; behavior model; NuSMV validation methods

0 引 言

近年来,随着嵌入式软件得不断进步与发展,原有的开发方法已经不能满足新系统的开发需求。由此,工程师们开发出许多基于软件开发方式的工程方法。例如模型驱动体系结构(MDA),MDA 是由 OMG(对象管理组织)提出并倡导的软件开发方法[1]。在 MDA 中,模型大致被分成三个层次:PIM(Platform Independent Model),也就是平台无关模型,这种模型主要用在分析和设计的初级阶段,所以先不对平台细节做考虑;PDM(Platform Description Model),平台描述模型,顾名思义,就是用来描述实施平台所搭建得模;PSM(The Platform Specific Model),平台相关模型,它包含了所有在 PIM 中表示的功能,并且把有关平台的设计细节也添加进去[2]。MDA 的设计模型框架如图 1 所示。而 MDA 的开发流程如图 2 所示。

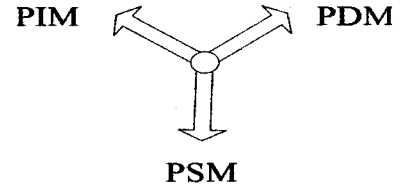

图 1 MDA 设计模型框架图

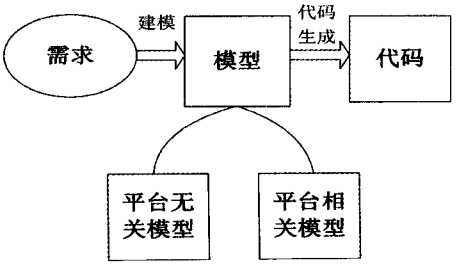

图 2 MDA 的开发流程图

OSATE(Open Source AADL Tool Environment)[3]是一个在 Eclipse 基础上建立的一套开源的集成开发环境,它是一个支持对 AADL 语言进行编辑、语法检查、安全检查等功能的建模和验证工具,在 OSATE 上

收稿日期:2011-07-15;**修回日期**:2011-10-21
基金项目:中央高校基本科研业务费专项资金(GK201002011);教育部科学教育重点项目(107106)
作者简介:刘 博(1988-),男(回族),河南郑州人,硕士研究生,主要研究领域嵌入式系统;李蜀瑜,副教授,博士,主要研究领域为嵌入式系统、Web 服务。

开发了多种分析插件,进行可调度性分析、安全性分析、时间延迟性分析等[4]。

AADL 的一般应用在实时的、安全的、面向任务的嵌入式软件设计当中。然而,在对这类模型进行验证的时候,往往要浪费大量的精力进行手动的验证。自动机技术的引入使得原本枯燥乏味的验证过程变得简单明了。当前版本的 AADL 分析验证工具主要是对静态的性能进行检测和评估,主要有以下验证工具:UPPAAL、Spin、NuSMV,不同的验证工具所针对的验证方向也不同,文中将主要使用 NuSMV 对 AADL 的行为模型进行形式化验证。

1　AADL 与 NuSMV

1.1　AADL 概述

AADL 可以详细描述嵌入式系统性能相关的属性,如可靠性、有效性、时间性、响应性、吞吐量、安全性。并且可以建立相应的图形化的模型。这样使得各个组件的性质、组件之间的联系清晰地展现在开发者眼前[5]。

在 AADL 中,把组件主要分为两大部分:一部分是软件组件,另一部分是执行平台组件。软件组件由数据、进程、线程、子程序等组成;执行平台组件由处理器、总线、存储器、外设等。AADL 的图形化表示方法把这两个部分用不同的图形清晰地区别开来[6]。

另外这两大部分之间由不同的项目和端口链接起来,使得软件组件和执行平台组件构成一个整体。更重要的是,使用图形化的模型表示方法可以清楚地表示组件的行为[7]。

1.2　验证工具 NuSMV

1987 年,在卡内基梅隆就读博士的 McMillan 开发出一种新的模型验证器 SMV(Symbolic Model Verifier),这个验证器使用 BDD 来缓解状态爆炸问题,实现了符号模型检测技术。2001 年,在 SMV2.4.4 版本的基础上进行实现和扩展,对软件构架加入了一些新的特征[8],图 3 为 NuSMV 的目录结构。在 SMV 的基础上,NuSMV 进行了三个方面的扩展:

首先是功能上,不仅支持 CTL 描述规范,同时也支持 LTL 描述规范,方便用户使用。然后是对系统构架的扩展,NuSMV 定义了一个良好的软件体系结构,使得用户操作更加简答。最后是对 NuSMV 文档和源码进行了进一步的扩充,使得 NuSMV 更容易读写。

NuSMV 的引入,大大简化了实时系统验证的工作量,并且增加了验证系统的可靠性。由于它高效地实现了符号模型检测技术,源代码开放,并且有很好的软件体系结构,易于定制和扩展,这就给研究者提供了更广阔的研究空间[9]。

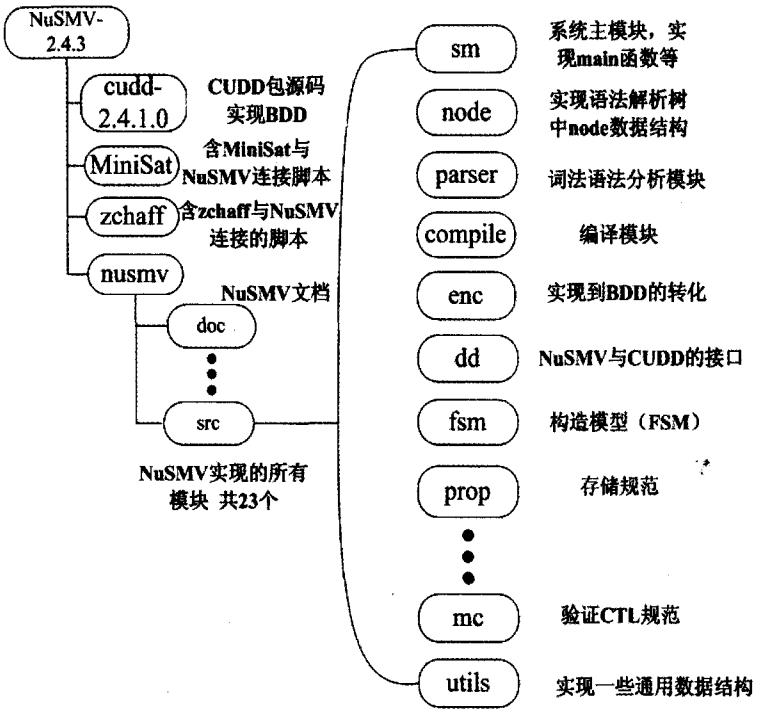

图 3　NuSMV 的目录结构图

2　使用 NuSMV 的验证过程

文中验证过程通过一个实例来引入,这样可以更加清晰地看到验证的过程。文中通过对自适应巡航控制与碰撞报警系统(Adaptive Cooperative Cruise Control and Collision Warning Systems)的验证来展示使用 NuSMV 的验证过程。

2.1　实例描述

系统描述:CACC 是一个复杂的巡航控制系统。它包括了两个监控系统:一个是碰撞检测雷达(FCW),另一个是无线通信系统。整个系统通过通信传递来控制车辆的行驶速度。除了一个手动的驾驶控制模式,CCAC 有三个以上的自动巡航速度控制机制[10]。而汽车想要达到的目标速度是由司机来设定的。当危险发生时,CACC 中的 FCW 模块会对驾驶司机发出报警,来提醒司机进行必要操作。CCAC 是通过对当前速度和目标速度的分析,经过 FCW 模块的计算和调节,来实现巡航的功能。

CACC 是由 4 个子系统(使用单层线描绘的矩形表示)和 9 个设备(用双层线表示)组成的。控制系统可以通过驾驶员输入、FCW 模块还有无线控制命令来实现 MANU、CC、ACC、CACC 系统之间的切换。当驾驶员想要自行控制车辆时,它的优先级是最高的,也就

是说当司机发出控制汽车的命令时，系统将无条件转到 CC 控制系统。在 CC 子系统中是通过分析当前速度和目标速度来判断是刹车还是加速。在此系统中，驾驶员可以按自己的意愿输入想要达到的速度。然而，在 ACC 子系统中，FCW 输入的目标速度是根据分析当前距离目的地的距离和汽车的当前速度来确定的。通过 FCW 的分析，FCW 把目标速度传递给 CC 系统来控制车辆。

图 4 是 CC 子系统的 AADL 图[5]。

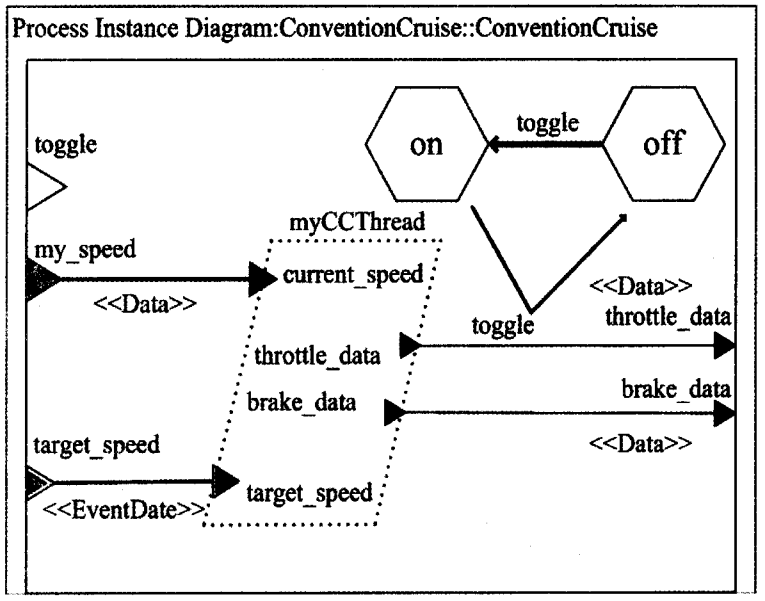

图 4 CC 子系统的 AADL 图

2.2 建立相应的 NuSMV 模型

基于 Eclipse 平台开发了一个模型转换器来实现从 AADL 模型到 UPPAAL 可识别的时间自动机模型的转换，并把它以插件形式集成在由美国卡内基梅隆大学软件工程研究所开发的 AADL 的开源开发工具环境（Open Source AADL Tool Environment，OSATE）中[11]。在 OSATE 环境下可以进行 AADL 模型代码的编辑、语法检查及模型系统实例化等操作[12]。

NuSMV 用其输入语言来描述 Kripke 结构和特征的规范。在 NSMV 中 Kripke 结构经常被称作为 Finite State Machine（FSM）。

NuSMV 输入语言，同 C 语言中的表达式和语句的概念有些相似。

AADL 支持层次型的结构框架，而 NuSMV 也支持层次型的结构框架。正是基于此种情况，AADL 模型可以简单直接的转换成为 NuSMV 模型。在这个过程中需要做的是忽略系统组件和状态转换组件，而组件间的联系通过专门功能的链接模块来控制。

2.3 局限性

文中，从 AADL 模型到 NuSMV 模型的转换过程忽略了一些系统组件，只是在层级构架的基础上实现的直接转化。

NuSMV 模型是由 4 个主要的模块组成的，它们分别是 main，CC，FCW，CFCW。这些模块和 5 个安全构件还有 5 个 CLT 的步骤在 CACC 模型中都被证明过。如程序中 safety properties 2 和 3，它们的错误是因为它们离实现安全手动控制只是差了一小步，这一小步是语义上的差别。然而让我们感兴趣的是 safety query # 4 的结果是 true。

同 safety #3 相比而言，CACC 模块的系统反应要比 ACC 模块的快很多。对于 progress properties 而言，#2、#4、#5 的结果为 true，而#1 和#3 的结果是 false。在没有没有公平性约束条件下，CACC 模块的控制优先级是高于 CC、ACC 模块的。但是在当前型号的 NuSMV 之中，没有提供这种约束条件。然而，在不借助工具的前提下，来实现这种转化是非常困难的。

CACC 系统模型程序如下：

```
MODULE main()
    VAR
    cruise:{manu, cc, acc, cacc};
    alarm:boolean;
    events:{none, toCruise, toOverride, toAdaptive,toCooperative, OutRange, LostCom};
        --override is to set cruise off
    current_speed   :{0, 1, 2, 3};
    target_speed    :{0, 1, 2, 3};
    throttle_data :{0, 1, 2, 3};
    brake_data :{0, 1, 2, 3};
cruiser:  cc(cc_toggle, current_speed, target_speed,throttle_data, brake_data);
    forward_collision_warning :  fcw(fcw_toggle, alarm);
cooperative_fcw:  cfcw(cfcw_toggle, alarm);
DEFINE cc_toggle := (cruise = cc | cruise = acc | cruise = cacc);
    DEFINE fcw_toggle := (cruise = acc);
    DEFINE cfcw_toggle := (cruise = cacc);
```

在此忽略了初始化、切换和定义三个子模块

```
SPEC  AG(alarm -> AF(cruise = manu))--Safety #1: true
    SPEC  AG(cruiser.action = brake -> AX cruise = manu) -- Safety #2, false, but false alarm
    SPEC  AG(events = OutRange -> AX (cruise = manu)) --Safety #3, false, but false alarm
```

```
SPEC  AG(events = LostCom -> AX (cruise = manu)) --Safety #4, true,
SPEC  AG(events = toOverride -> AX(cruise = manu)) --Safety #5, true,
SPEC  AG(events = toAdaptive -> AF(cruise = acc ) ) --progress #1, false
SPEC  AG ( events = toCooperative -> AF (cruise = cacc) )--progress #2, true
SPEC  AG(events = toCruise -> AF(cruise = cc)) --progress #3, false
SPEC  AF(current_speed < target_speed ->
    AF(current_speed = target_speed) ) --progress #4, true
    SPEC  AF(current_speed > target_speed ->
    AF(current_speed = target_speed) ) --progress #5, true
```

3　结束语

文中探讨了使用形式化验证工具 NuSMV 来验证 AADL 模型行为的过程,总结了 AADL 模型到 NuSMV 模型转化的优势。同时也通过实例证明了 NuSMV 作为 AADL 验证工具的不足。并且也指出了这些不足的地方。NuSMV 是一种能够很好的支持层次型构架的验证模型,但是,用它作为 AADL 的验证工具来说,还是有很多不足的。文中只是对 NuSMV 验证 AADL 模型的过程进行了初步的探讨,在今后的研究中,将进一步深入学习研究 NuSMV 验证方法。

参考文献:

[1] 胡　军,李宣东,郑国梁. 构件化嵌入式软件设计的分析与验证[D]. 南京:南京大学,2005.

[2] 刘　倩,桂盛霖,李　允,等. 基于 UPPAAL 的 AADL 模型可调度性验证[J]. 计算机应用,2009(7):1820-1824.

[3] 王瀚博,周兴社,董云卫,等. 结构分析和设计语言 AADL 研究[J]. 计算机工程与应用,2009,45(16):1-4.

[4] 杨志斌,皮　磊,胡　凯,等. 复杂嵌入式实时系统体系结构设计与分析语言:AADL[J]. 软件学报,2010(5):899-915.

[5] 陶　勇,桂盛霖,马　亮,等. AADL 模型的代码自动生成及集成技术[J]. 计算机工程,2009(8):59-61.

[6] 王　庚,周兴社,张　凡,等. AADL 模型的测试方法研究[J]. 计算机科学,2009(11):127-130.

[7] 马春燕,董云卫,朱宇峰,等. AADL 测试模型的构造研究[J]. 西北工业大学学报,2010(6):969-973.

[8] 张军林,张治国. NuSMV 模型验证器实现与分析[D]. 广州:中山大学,2010.

[9] Liu Hong, Gluch D P. Formal Verification of AADL Behavior Models: A Feasibility Investigation[C]//Proceedings of the 47th Annual Southeast Regional Conference. [s. l.]:[s. n.], 2009:19-21.

[10] The SAE Architecture Analysis & Design Language (AADL)[M]. Reading, MA: Addison-Wesley, 1975.

[11] Liu H, Gluch D P. Conceptual Modeling with the Object-process Methodology in Software Architecture[J]. Journal of Computing Education in Colleges, 2004(19):10-21.

[12] 汤小明,苏罗辉,宋科璞. 飞行管理系统 AADL 建模与分析[J]. 计算机技术与发展,2010,20(3):191-194.

(上接第 109 页)

参考文献:

[1] Akyildiz I, Su W, Sankarasubramaniam Y, et al. A Survey on Sensor Networks[J]. IEEE Communications Magazine, 2002, 40(8):102-114.

[2] 王中生,曹　琦. 基于 ZigBee 技术的无线定位研究与实现[J]. 计算机技术与发展,2010,20(12):189-190.

[3] He T, Huang C D, Blum B M, et al. Range-Free Localization Schemes in Large Scale Sensor Networks[C]//Proc of the 9th Annual Int'l Conf on Mobile Computing and Networking. San Diego: ACM Press, 2003:81-95.

[4] Grid L, Estrin D. Robust Range Estimation Using Acoustic and Multimodal Sensing[C]//Proc IEEE/RSJ Int Conf Intelligent Robots and Systems (IROS). Maui, Hawaii, USA: [s. n.], 2001:1312-1320.

[5] Doherty L, Pister K S J, Ghaoui L E. Convex position estimation in wireless sensor networks[J]. IEEE Computer and Communications Societies, 2001, 3(5):1655-1663.

[6] Niculeseu D, Nath B. DV based positioning in ad hoc networks[J]. Journal of Telecommunication Systems, 2003, 22(1-4): 267-280.

[7] Hightower J, Boriello G, Want R. SpotON: An Indoor 3D Location Sensing Technology Based on RF Signal Strength[R]. Washington: Univ of Washington, 2000.

[8] 朱　浩,顾宗海,苏　金. 一种基于交点质心求解的 RSSI 定位算法及其优化[J]. 郑州大学学报(工学版),2010,31(6):43-46.

[9] 蒋峥峥,王汝传,孙力娟. 基于移动 Agent 无线传感器网络节点自定位算法[J]. 计算机技术与发展,2007,17(6):206-209.

[10] 文　举,金建勋,袁　海. 一种无线传感器网络四边测距定位算法[J]. 传感器与微系统,2008,27(5):108-110.

[11] 王书聪. 无线传感器网络分布式定位算法研究[J]. 计算机技术与发展,2008,18(11):62-63.

[12] 任维政,徐连明,邓中亮. 基于 RSSI 的测距差分修正定位算法[J]. 传感技术学报,2008,21(7):1247-1250.