

SysML：一种新的系统建模语言

蒋彩云，王维平，李群

(国防科学技术大学信息系统与管理学院，长沙 410073)



摘要：SysML(Systems Modeling Language)是国外正在研究的一种新的系统建模语言，是为系统工程应用开发的标准建模语言，它的内容包括SysML语义和SysML表示法两个部分。介绍了SysML的产生背景、元模型理论、语言体系结构和语言形式，重点介绍了SysML的九种基本图形：类图、装配图、用例图、需求图、参数图、顺序图、活动图、状态机图和时间图，总结了SysML的主要特点和应用领域。

关键词：SysML；SysML语义；SysML表示法；系统工程

中图分类号：TP311.51 **文献标识码：**A **文章编号：**1004-731X(2006)06-1483-05

SysML: A New Systems Modeling Language

JIANG Cai-yun, WANG Wei-ping, LI Qun

(School of Information System & Management, National University of Defense Technology, Changsha 410073, China)

Abstract: SysML is a new kind of systems modeling language being studied abroad. It is a standard modeling language for systems engineering application, which includes SysML semantics and SysML representation. The background, metamodel theory, language architecture and formalism of SysML were introduced. More attention was paid to nine basic diagrams of SysML which are class diagram, assembly diagram, use case diagram, requirement diagram, parametric diagram, sequence diagram, activity diagram, state machine diagram and timing diagram. Finally, main characteristics and application scope of SysML were summarized.

Key words: SysML; SysML semantics; SysML representation; systems engineering

引言

当前，系统工程师使用的建模语言、工具和技术种类很多，如行为图、IDEF0、N2图等，这些建模方法使用的符号和语义不同，彼此之间不能互操作和重用。系统工程正是由于缺乏一种强壮的“标准”的建模语言，从而限制了系统工程师和其他学科之间关于系统需求和设计的有效通信，影响了系统工程过程的质量和效率。

统一建模语言 UML (United Modeling Language)是面向对象的标准建模语言，自 1997 年 11 月 17 日被 OMG(Object Management Group)批准为标准以来，UML 已经获得工业界、科技界和应用界的广泛支持。同时 UML 自身也在不断地发展和完善，目前的最新版本是 UML2.0。虽然 UML 的设计初衷是为软件开发提供一种标准建模语言，但 OMG 标准化过程也支持为特殊领域定制 UML，如系统工程领域。早在 2000 年，Ingmar Ogren^[1]就探讨了定制 UML 满足系统工程需要的可能性，提出把 UML 的子集(主要是组件图)和编程语言 Ada95 的伪代码子集结合起来创建一种系统工程建模语言 SEML(Systems Engineering Modeling Language)。随后，Jakob Axelsson^[2]提出扩展 UML 使之能对具有连续时间行为的物理组件建模。Terry Bahill 和 Jesse Daniels^[3]提出

在非软件领域如系统、硬件和算法的设计中应用 UML 工具。Conrad Bock^[4]把 UML2.0 活动图与系统工程中广泛使用的增强功能流块图 EFFBD(Enhanced Functional Flow Block Diagram)进行了比较，提出修改或扩展 UML2.0 满足系统工程的功能流建模需求。

为了满足系统工程的实际需要，国际系统工程学会 INCOSE(International Council on Systems Engineering)和对象管理组织 OMG 决定在对 UML2.0 的子集进行重用和扩展的基础上，提出一种新的系统建模语言——SysML(Systems Modeling Language)，作为系统工程的标准建模语言。和 UML 用来统一软件工程中使用的建模语言一样，SysML 的目的是统一系统工程中使用的建模语言。2003 年 3 月，OMG 公布了 UML for SE RFP^[5](UML for Systems Engineering Request for Proposal)，5 月召开了首次会议，并成立了由用户、开发商和政府机构组成的支持 SysML 的非正式组织。2004 年 1 月 12 日，SysML 的非正式组织向 OMG 提交了 SysML 语言的初步草案，2004 年 8 月 2 日向 OMG 提交了修改后的 SysML0.8 版，2004 年 10 月 11 日向 OMG 提交了第二次修改后的 SysML0.85 版，2005 年 1 月 10 日向 OMG 提交了第三次修改后的 SysML0.9^[6]版。SysML0.9 版是一个重要的里程碑，确定了核心的系统工程图形。预计 SysML1.0 正式版将在 2005 年的第二季度被 OMG 作为标准采纳，2005 年年内工具开发商将推出 SysML 的集成支持环境。

SysML 是一种多用途的标准建模语言，能够支持各种

收稿日期：2005-04-19

修回日期：2005-11-28

作者简介：蒋彩云(1974-)，女，湖北人，博士生，研究方向为系统论证与仿真评估；王维平(1962-)，男，辽宁人，教授，博导，研究方向为系统工程学科、系统论证与仿真评估。

复杂系统的详细说明、分析、设计、验证和确认，这些系统可能包括硬件、软件、信息、过程、人员和设备等。SysML的定义包括 SysML 语义和 SysML 表示法两个部分。

1 SysML 语义

SysML 为系统的结构模型、行为模型、需求模型和参数模型定义了语义。结构模型强调系统的层次以及对象之间的相互连接关系，包括类和装配。行为模型强调系统中对象的行为，包括它们的活动、交互和状态历史。需求模型强调需求之间的追溯关系以及设计对需求的满足关系。参数模型强调系统或部件的属性之间的约束关系。SysML 为模型表示法提供了完整的语义。

1.1 元模型理论

和 UML 一样，SysML 语言的结构也是基于四层元模型结构：元元模型、元模型、模型和用户对象。

元-元模型层具有最高抽象层次，是定义元模型描述语言的模型，为定义元模型的元素和各种机制提供最基本的概念和机制。

元模型是元-元模型的实例，定义模型描述语言的模型。元模型提供了表达系统的各种包、模型元素的定义类型、标记值和约束等。

模型是元模型的实例，定义特定领域描述语言的模型。

用户对象是模型的实例。任何复杂系统在用户看来都是相互通信的具体对象，目的是实现复杂系统的功能和性能。

1.2 语言组织结构(包结构)

SysML 语言重用和扩展了 UML 的很多包(如图 1 所示)，使用的扩展机制包括模型元素的定义类型(stereotype)、元类(metaclass)和模型库(model library)。SysML 的用户模型是通过实例化模型元素的定义类型 stereotype 和元类 metaclass 以及构造模型库中类的子类来创建的。

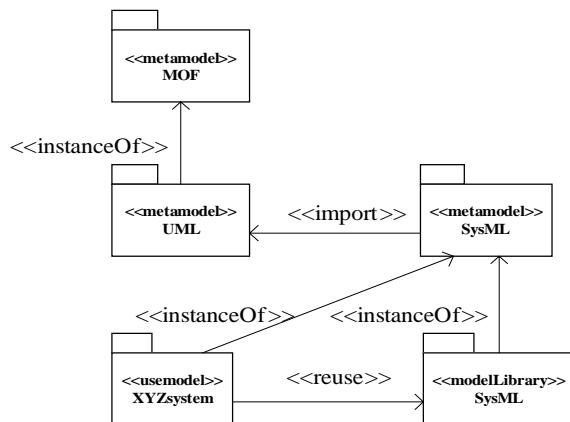


图 1 SysML 对 UML 的扩展

SysML 包结构(如图 2 所示)和 UML 的包结构在很大程度上是一致的。一些 UML 包没有被重用，因为它们在系统工程中不再需要；UML 中的状态机包、交互包和用例包在

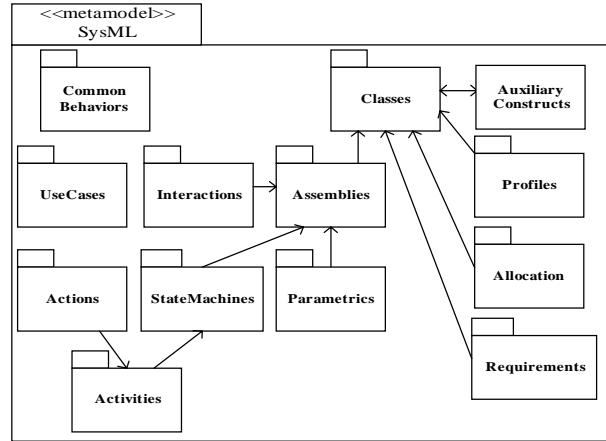


图 2 SysML 的包结构

SysML 中被重用；在 SysML 的活动包、类包和辅助包中增加了一些新的扩展；另外还增加了一些 UML 中没有的新包，如需求包、参数包和分配包。

1.3 语言形式

形式化表示方法的好处有：(1) 提高了描述的正确性；(2) 减少了描述的二义性和不一致性；(3) 增强了描述的可读性。但语言的完全形式化是极为复杂的，因此，为了保持描述的清晰易懂，SysML 和 UML 一样在给出自身的语义说明时采用了半形式化的描述方法。SysML 用自然语言(英语)描述约束和详细语义，力求实现形式严格和易于理解之间的平衡。通过 ISO-AP233 数据交换标准和 XMI 模型交换标准，SysML 语法支持各种系统工程工具之间的互操作。

2 SysML 的图形表示

SysML 的图形表示是 SysML 的可视化表示，是用来为系统建模的工具。SysML 定义了九种基本图形来表示模型的各个方面。从模型的不同描述角度来划分，这九种基本图形分成四类：结构图(Structure Diagram)、参数图(Parametric Diagram)、需求图(Requirement Diagram)和行为图(Behavior Diagram)。结构图包括类图(Class Diagram)和装配图(Assembly Diagram)，行为图包括活动图(Activity Diagram)、顺序图(Sequence Diagram)、时间图(Timing Diagram)、状态机图(State Machine Diagram)和用例图(Use Case Diagram)。

2.1 类图

类图定义了类以及类之间的关系。类包括属性和操作，类关系包括关联关系(association)、泛化关系(generalization)和依赖关系(dependency)。

SysML 类图重用并扩展了 UML 类图。SysML 的 DependencySet 包是对 UML::Kernel::Package 包的扩展，用来对具有共同依赖端或被依赖端的依赖关系分组。SysML 扩展了类和对象的符号，能够表示类的多层继承关系。

图 3 是车辆系统的操作上下文类图。其中《context》、《system》和《external》是用户定义的 stereotype，不是 SysML

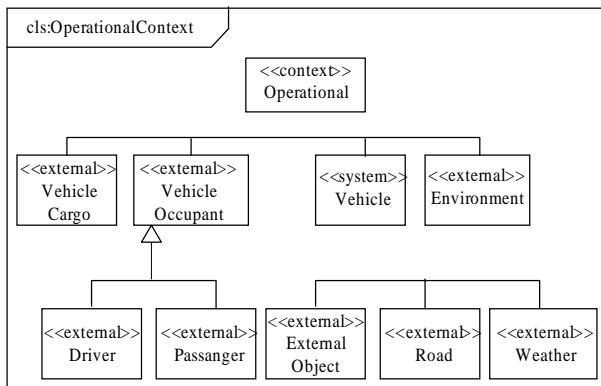


图3 车辆系统的操作上下文类图

固有的,但有助于建模者确定系统及其环境。类图左上角标题中的cls表示类图,冒号后面的OperationalContext是类图的名称。其他基本图形标题的格式与类图标题格式一样,例如,活动图用act表示,装配图用asm表示,参数图用par表示,需求图用req表示,用例图用u-c表示,顺序图用seq表示,状态机图用s-m表示,时间图用tim表示。

2.2 需求图

需求图(图4)是一种新的SysML图形,能够描述需求和需求之间以及需求和其他建模元素之间的关系。需求是指系统必须满足的能力或条件,一个需求能够分解成多个子需求。SysML用《requirement》说明需求,它也是一个类,有两个属性: text 和 id,前者是需求的文本描述,后者是需求的标识符。用户可以定义需求的子类,如操作需求、功能需求、接口需求、性能需求等等。使用导出关系《derive》表示一个需求可以从另一个需求产生或导出,使用满足关系《satisfy》表示一个需求能被其他的模型元素实现,使用验证关系《verify》表示一个需求能被测试例子验证。SysML的《derive》、《satisfy》和《verify》都是继承UML的《trace》。SysML用《rationale》表示基本原理注释元素,能够附在任何模型元素上,用来说明建模决策如分析决策或设计决策的原理或原因。

2.3 用例图

用例图(图5)描述了外部参与者对系统的使用,这是通过系统向参与者提供一系列服务来实现的。用例图包括用例、参与者以及它们之间的通讯。参与者可能是用户、外部系统或其他环境实体,它们和系统直接或间接交互。

用例之间的关系有包含关系(include)、扩展关系(extend)和泛化关系(generalization)。包含关系提供了一种分离出多个用例中共享的共同功能作为基本用例执行的机制; 扩展关系提供了可选的功能,在定义的扩展点和指定的条件下扩展基本用例; 泛化关系提供了基本用例的特殊化机制。SysML用例图重用了UML用例图。

2.4 状态机图

状态机图(图6)通过状态以及状态之间的转移对离散行为建模,它把行为表示为对象的状态历史。在状态的转移、进入和退出过程中会调用活动,并指定相关的事件和守卫条件。在状态中调用的活动称为“do”活动,可以是连续的,也可以是离散的。一个组合状态有嵌套的状态,嵌套的状态可以是顺序的,也可以是并发的。SysML状态机图重用了UML状态机图。

2.5 装配图

装配图把系统描述为部件的集合,并显示了部件之间的连接关系。装配图中的元素包括部件(part)、端口(port)和连接器(connector)。部件是装配的内部元素,连接器将部件连接起来,使部件之间的通信成为可能。一些部件显示为系统边界上的端口,表示它们能够从外部连接,提供了装配和外部环境的交互点。每个部件定义为一个类,部件也可以是一个装配,有它自己的部件、端口和内部结构。

SysML装配图是从UML组合结构图扩展而来的,提供了对系统进行模块化建模的能力,促进了建模元素的重用。为了和UML结构化类区别,SysML定义了《assembly》来说明一个装配,《assembly》也是一个类。SysML的嵌套连接器终端NestedConnectorEnd是从UML的ConnectorEnd扩展而来的,用来说明嵌套的连接器。与此相应,在SysML

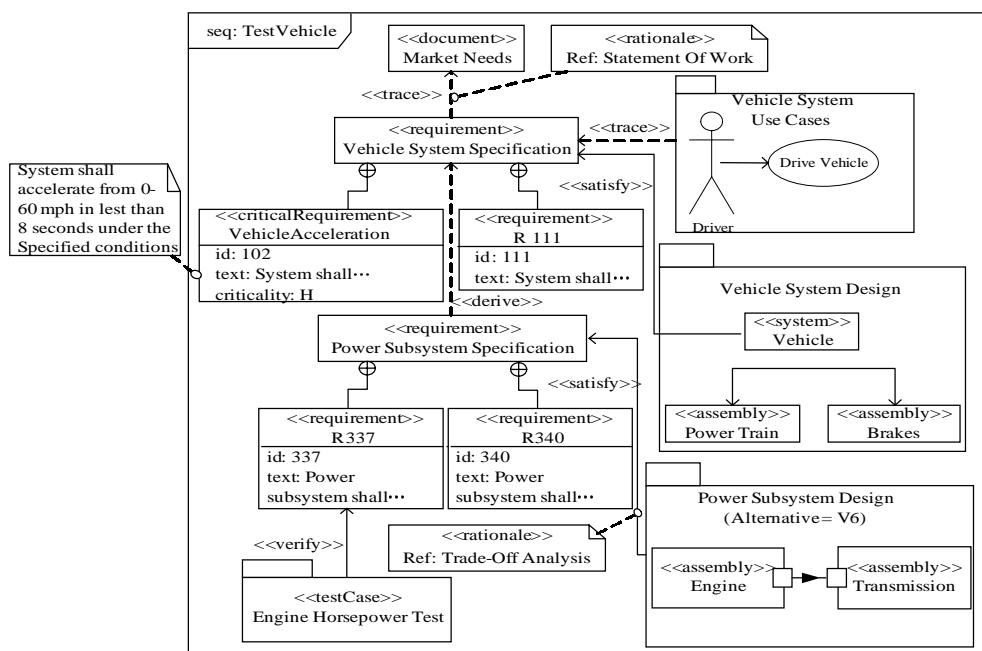


图4 车辆系统的需求图

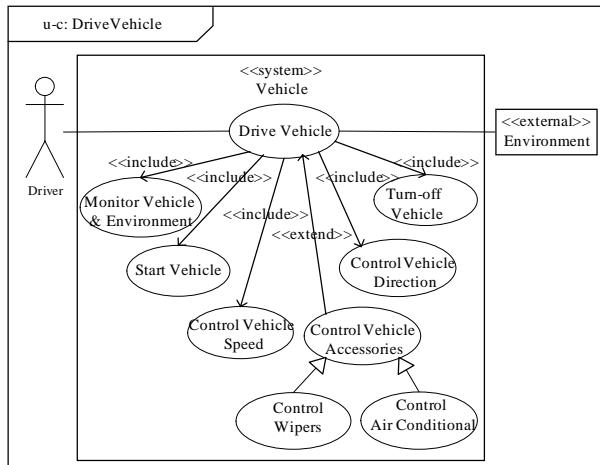


图 5 驱动车辆用例图

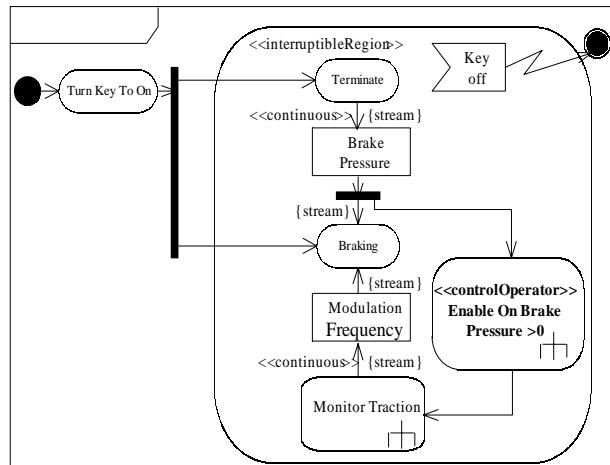


图 8a 操作汽车活动图

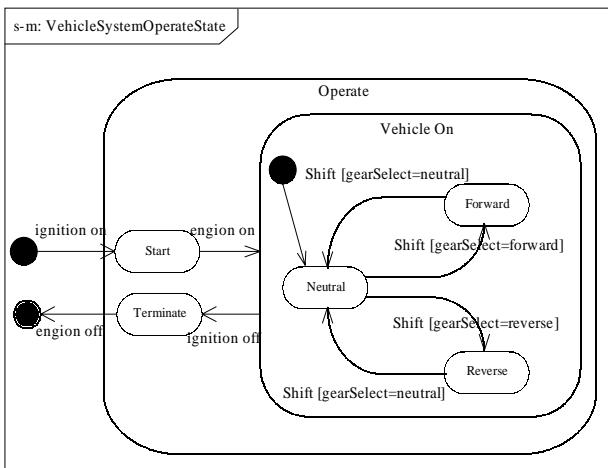


图 6 车辆系统操作状态的状态机图

的图扩展中增加了对嵌套的连接器终端以及属性路径引用的表示方法。

装配图可以是黑箱视图(如图 7a 所示)，也可以是白箱视图(如图 7b 所示)，前者只显示外部可见的元素，后者还显示了部件及其连接的内部细节。

2.6 活动图

活动建模强调活动的输入输出、顺序和条件。SysML 活动图(图 8a)和系统工程领域常用的增强功能流块图 EFFBD(Enhanced Functional Flow Block Diagrams)类似，只是采用的术语和符号不同。

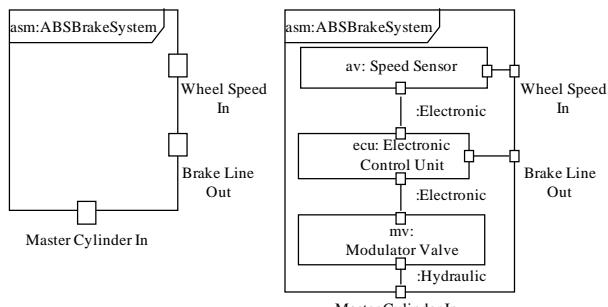


图 7a 装配的黑箱视图

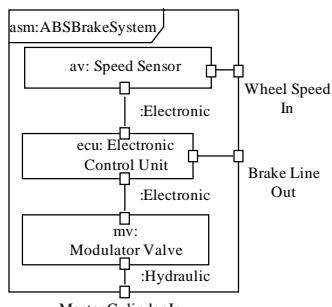


图 7b 装配的白箱视图

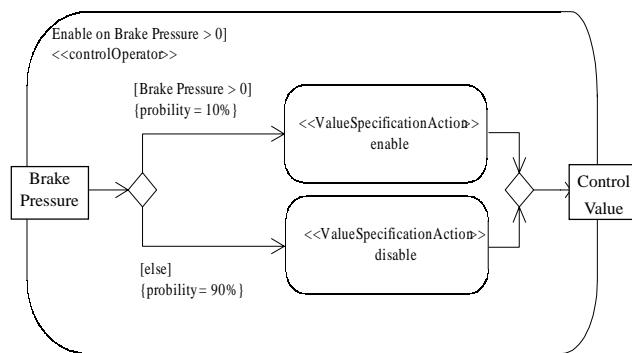


图 8b 控制操作符

SysML 活动图对 UML 活动图进行了扩展，包括把控制作为数据、表示连续的物质流或能量流、引入概率等等。在 UML 活动图中，控制只能使动作开始。在 SysML 活动图中，控制既能使动作开始，又能使正在执行的动作终止。SysML 支持控制操作符(图 8b)，控制操作符是一个逻辑操作符，可以根据输入产生一个控制值输出。SysML 支持对实体流速率的限制，包括物质、能量、信息的连续流和离散流。SysML 扩展了对象节点，包括插脚，使得新到的值可以取代对象节点中已经存在的值；对象节点还可以丢弃没有立即流到下游的值。这两个扩展对于确保动作获得最新信息、避免快速或连续流动的值聚集在对象节点以及对瞬时值如电信号建模是有用的。SysML 在活动中引入了概率，用来表示一个值离开决策点的可能性，输出参数集也可以用概率表示某个输出的可能性。SysML 为活动扩展了类图符号，说明了活动之间的组合关联语义，定义了活动图和类图之间的一致性规则。

图 8a 举例说明了怎样对连续系统建模，这是一个简化的驱动和制动汽车的活动图。打开钥匙起动了两个行为：驱动和制动，它们使用流参数和其他功能通信，一直执行到钥匙关闭。驱动行为对制动行为连续地输出一个制动压力，制动压力信息也流到控制操作符。控制操作符根据输入的制动压力的大小产生一个控制值(enable 或 disable)输出，来控制监控牵引力活动。监控牵引力活动没有控制插脚，因此根据

SysML 语义, 一旦监控牵引力活动开始, 从控制操作符连续到达的 enable 控制值没有效果。监控牵引力活动开始后, 它对制动行为连续地输出一个调制频率。当制动压力等于 0 时, 从控制操作符输出 disable 控制值, 第一个 disable 控制值使监控牵引力活动终止, 其余的 disable 控制值没有效果。控制操作符和监控牵引力活动中的靶形符号表示它们可以被进一步定义。

2.7 参数图

参数图(图 9)是一种新的 SysML 图形, 定义了一组系统属性以及属性之间的参数关系。参数关系说明了一个属性值的变化怎样影响其他的属性值, 参数关系是没有方向的。参数模型是分析模型, 把行为模型和结构模型与工程分析模型如性能模型和可靠性模型等结合在一起, 能用来支持权衡分析, 评价各种备选的解决方案。

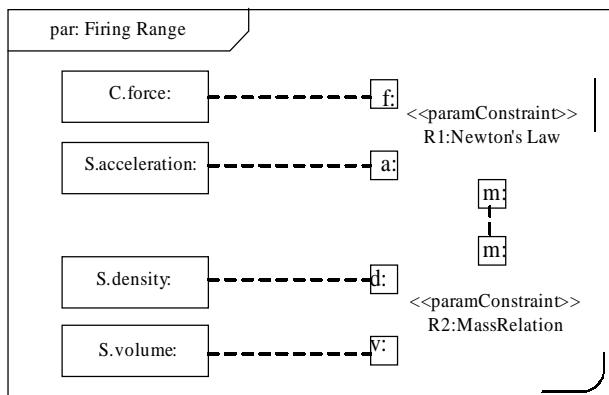


图 9 大炮开火范围参数图

参数约束是一个装配, 用《paramConstraint》表示, 通常和 SysML 装配图结合起来使用。《paramConstraint》声明一个装配用作参数约束, 它的端口定义了参数约束的参数, 《paramConstraint》装配的唯一有效用法是用值绑定约束绑定参数和装配中的其他属性。参数约束内部可以包含其他参数约束, 但不能包含其他部件或属性。参数约束关系用来表示系统的结构模型中属性之间的依赖关系, 可以是基本的数学操作符, 也可以是和物理系统的性质有关的数学表达式如 $F = m * a$ 和 $a = dv/dt$ 等。《valueBindingConstraint》从 UML 的 Connector 扩展而来, 它声明两个相连的属性有相同的值, 用虚线表示。

2.8 时间图

时间图(图 10)表示状态、活动或属性值(离散的或连续的)随时间的变化, 也可以表示事件的时间顺序或表达式为真的时间。时间图是一个二维图, 横轴代表时间, 纵轴代表状态、活动或属性。SysML 时间图重用了 UML 时间图。

2.9 顺序图

顺序图(图 11)根据控制流指定了一连串的交互, 控制流

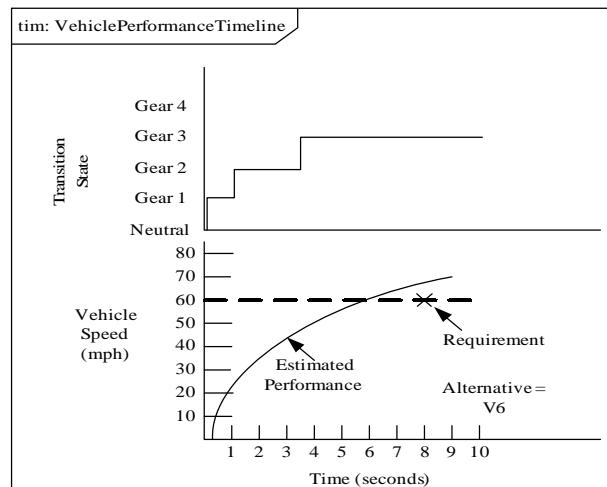


图 10 车辆性能时间图

是通过对对象生命线之间发送和接收消息来定义的。消息包含了控制和数据流, 它发起接收消息的对象的行为, 并把输入传给行为, 消息的时间顺序和它在顺序图上的垂直位置相关。复杂的顺序可以抽象成引用顺序图。顺序图上可以包含条件逻辑, 表示选择、连续流和循环等。对象生命线可以进一步分解为它的组成部分。SysML 顺序图重用了 UML 顺序图。

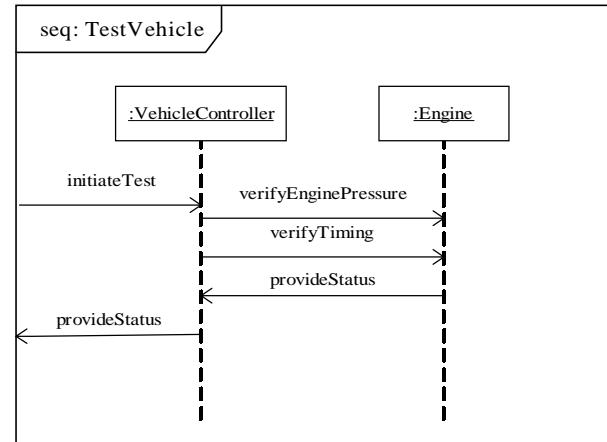


图 11 测试车辆的顺序图

3 SysML 的主要特点及应用领域

SysML 是 UML 在系统工程应用领域的延续和扩展。和其他系统工程建模语言相比, SysML 是一种通用的、功能强大的标准建模语言。它消除了不同方法在表达法和术语上的差异, 避免了符号表示和理解上不必要的混乱。

SysML 独立于任何一种系统工程过程和方法, 但支持过程和方法。SysML 的重点是标准建模语言, 不是标准过程或方法。SysML 的应用必然与过程相关, 不同的系统工程应用领域要求不同的过程。SysML 的开发者提出的开发过程是模型驱动、以体系结构为中心、迭代递增的过程, 这可以说是一个元过程。

(下转第 1492 页)

5 结论

通过对隐通道的仿真，我们得出一些新的隐通道性质，这些性质在静态的隐通道分析中是未知的。

性质1：隐通道的带宽与隐通道工作的硬件环境密切相关，受硬件读写速度的限制，隐通道的实际带宽不可能出现静态分析中的无穷大结果(试验1、试验3)。

性质2：一个实际系统中的隐通道带宽是有上限的。当系统中工作的隐通道带宽已经接近或达到这一极限时，简单增加隐通道的数量会造成隐通道整体带宽的下降(试验4、试验5、试验6)。

性质3：在隐通道中传输的文件，图形文件具有最佳的抗干扰能力(试验7)。

通过这些性质，我们认为，单纯基于源代码或顶级描述的静态隐通道分析，会夸大隐通道的实际威胁(性质1)；隐通道的聚集问题(并行使用多条隐通道)，带宽并不像理论分析的那样简单相加^[6](性质2)；作为隐通道的用户，可能会首选将敏感信息保存在图片中传送(性质3)。

参考文献：

- [1] John McHugh. Covert Channel Analysis: A Chapter of the Handbook for the Computer Security Certification of Trusted Systems [R]. Portland State University, 1995, 16.

(上接第1487页)

SysML能对系统工程的各种问题建模，适用于系统工程的不同阶段，特别是在系统工程的详细说明阶段和设计阶段，使用SysML来说明需求、系统结构、功能行为和分配非常有效。

4 结论

本文简要介绍了SysML语义和表示法，包括SysML元模型理论、语言体系结构和形式，以及九种基本图形。限于篇幅，SysML的分配allocation和辅助结构auxiliary-Constructs以及基本图形的混合用法没有介绍。SysML是一种先进实用、功能强大的标准建模语言，能对系统工程的各种问题建模。在应用SysML解决系统工程问题时，必须有相应的SysML支持环境来支持。目前，作为SysML合作伙伴的各工具开发商正在致力于SysML建模与仿真环境的开发，这必将促进SysML技术的发展，普及SysML技术的应

- [2] Changda Wang, Shiguang Ju. The New Criteria for Covert Channels Auditing [C]//5th IEEE Information Assurance Workshop, 9-11, NY, U.S.A, 2004: 183-186.
[3] Paul A. Karger and John C. Wray, Storage Channels in Disk Arm Optimization [C]//IEEE Computer Society Symposium on 20-22, 1991: 52-61.
[4] Changda Wang, Shiguang Ju, Dianchun Guo, Zhen Yang, Wenyi Zheng. Research on the methods of search and elimination in covert channel [C]//Lecture Notes in Computer Science 3032, 2004: 988-991.
[5] 王昌达, 鞠时光. BL P 安全模型及其发展, BL P 安全模型及其发展[J]. 江苏大学学报(自然科学版), 2004, 25(1): 68-72.
[6] Changda Wang, Shiguang Ju, Searching Covert Channels by Identifying Malicious Subjects in the Time Domain [C]//5th IEEE Information Assurance Workshop, 9-11, NY, U.S.A, 2004: 68-73.
[7] 王昌达, 鞠时光. 隐通道存在的最小条件及其证明[J]. 计算机科学, 2005, 32(1): 77-79.
[8] S B Lipner. A Comment on the Confinement Problem [J]. ACM Operating System Review (S0163-5980). 1975, 9(5): 192-196.
[9] SHIHUH-PYNG SHIEH, Estimating and Measuring Covert Channel Bandwidth in Multilevel Secure Operating Systems [J]. Journal of Information Science and Engineering (S1016-2364). 1999: 91-106.
[10] Nick Ougurtsov, Hilarie Orman, Richard Schroeppel, etc. Experimental Results of Covert Channel Limitation in One-WayCommunication Systems [C]//Network and Distributed System Security, Symposium, 1997: 2-15.

用，从而有力地推动系统工程理论和实践的发展。

参考文献：

- [1] Ingmar Ogren. Possible Tailoring of the UML for Systems Engineering Purposes [J]. Systems Engineering (S1098-1241). 2000, 3(4): 212-224.
[2] Jakob Axelsson. Model Based Systems Engineering Using a Continuous-time Extension of the Unified Modeling Language (UML) [J]. Systems Engineering (S1098-1241). 2002, 5(3): 165-179.
[3] Terry Bahill, Jesse Daniels. Using Object-Oriented and UML Tools for Hardware Design: A Case Study [J]. Systems Engineering (S1098-1241). 2003, 6(1): 28-48.
[4] Conrad Bock. UML 2 Activity Model Support for Systems Engineering Functional Flow Diagrams [J]. Systems Engineering (S1098-1241). 2003, 6(4): 249-265.
[5] OMG. Systems Engineering Domain Special Interest Group (SE-DSIG). UML for systems engineering RFP [EB/OL]. (2003-03-01) [2005-04-18]. <http://www.uml.org/cgi-bin/doc?ad/03-03-41, 2003>
[6] OMG.SysML-v0.9-PDF-050110.pdf [EB/OL]. (2005-01-10) [2005-04-18]. <http://www.sysml.org, 2005>.