

光电干扰武器系统 C3I 可执行建模的研究

陈长喜^{1,2}, 郭立红¹, 于 涌¹, 唐雪松^{1,2}

(1. 中国科学院长春光学精密机械与物理研究所, 长春 130033; 2. 中国科学院研究生院, 北京 100039)

摘 要: UML 已经成为面向对象分析与设计建模事实上的标准。基于 UML 的 C3I 系统模型的设计结果是对系统的静态表述, 而 C3I 系统的本质是动态的。着色 Petri 网(CPN)具有强大的描述能力及严密的数学基础和多种分析手段, 并且是可以仿真运行的。提出了从 UML 产品映射到可用于逻辑、行为和性能方面体系结构评价的 CPN 可执行模型的方法, 对光电干扰武器系统 C3I 进行了 UML 建模的一般描述, 阐述了光电干扰武器系统 C3I 从 UML 产品映射到 CPN 建立可执行模型的过程。通过对可执行模型 CPN 的仿真可对光电干扰武器系统 C3I 进行完整准确的评价。

关键词: C3I; 建模; UML; 着色 Petri 网

Research on Executable Models of C3I in Photoelectric Interferential System

CHEN Changxi^{1,2}, GUO Lihong¹, YU Yong¹, TANG Xuesong^{1,2}

(1. Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033;

2. Graduate School of Chinese Academy of Sciences, Beijing 100039)

【Abstract】 Unified modeling language (UML) has been the facto standard for objected-oriented analysis and modeling design. The modeling result based on UML is static description for the command, control, communication, intelligence (C3I) system, but the essence of C3I system is dynamic. Colored Petri net(CPN) possesses strongly descriptive capability and rigorous mathematics foundation and many analytical method, and it is also able to be emulational. The method that is able to be used for logical, behavioral and performance evaluation of architecture from UML artifact mapping to CPN is put forward. Model of the general description based UML is established. The executable model that is established by mapping from UML to CPN is followed to set forth at length. The C3I models of photoelectric interferential system can be well and truly evaluated by simulation of the executable models of CPN.

【Key words】 C3I; Modeling; Unified modeling language (UML); Colored Petri net (CPN)

1 概述

C3I^[1]泛指支持各类军事机构的设施实时获取信息, 处理信息, 传输信息, 进行态势分析、威胁估计、决策, 然后实施有效指挥和控制, 以夺取军事优势的复杂而宏大的人—机系统。它突破了传统战争的空间和时间观念, 实现指挥自动化, 代替传统人工指挥手段从而大幅提高指挥机关的指挥效率, 被誉为作战效能的“倍增器”。

光电干扰武器系统由跟踪、瞄准系统, 指挥控制系统, 干扰实施系统组成。以指挥控制系统为核心, 指挥控制系统控制跟踪、瞄准系统和干扰实施系统。具体过程是由雷达进行目标捕获与引导, 通过跟踪、瞄准系统的传感器捕获并跟踪目标, 经过预处理获取目标的航迹信息, 并将此信息传送给指挥控制系统。指挥控制系统对目标信息进行如坐标转换、目标的判别、威胁估计等处理操作、呈现态势图并提交作战方案给指挥员, 待指挥员发出命令后干扰实施系统瞄准目标进行光电干扰, 最后评估干扰效果。它智能化、自动化程度高, 而且与其它装备相比, 研制成本较高, 研制周期较长。在开发光电干扰武器系统中的 C3I 系统之前先行建模, 能使系统开发和维护人员增强对系统的理解、保证系统分析、系统设计以及系统维护的一致性, 更重要的是能够降低系统研制风险, 缩短研制周期。UML 已经成为面向对象分析与设计建模事实上的标准。基于 UML 的 C3I 系统模型的设计结果

是对系统的静态表述, 而 C3I 系统的本质是动态的。CPN 具有强大的描述能力及严密的数学基础和多种分析手段, 并且是可以仿真运行的^[2]。本文研究光电干扰武器系统中 C3I 从基于 UML 的静态建模到建立动态可执行模型 CPN 的过程。

2 从 UML 静态模型到动态 CPN 映射方法

在构造 CPN 时须创建从 UML 元素到 CPN 元素的明确映射^[3], 这包括如位置、转移、输入、输出弧等结构元素, 还包括颜色集和在全局声明结点中的变量, 最后还需决定 CPN 的初始化标记。因为类图是体系结构面向对象设计的一般描述, 所以选择类图和活动图来映射 CPN 可执行模型, 具体算法如下:

(1)在转换可执行模型之前, 须纠正 UML 产品的一致性错误, 维护统一数据字典以支持类图和包含规则的行为图等产品。

(2)将类图变换成适宜的形式: 1)将类图中所有的关联关系转换为关联类。根据活动图所定义的消息为类图中的关联关系定义属性, 将其转换为关联类; 2)通过聚合关系将类图

作者简介: 陈长喜(1970 -), 男, 博士、博导、研究员, 主研方向: C3I 系统研究与应用; 郭立红, 博士、博导、研究员; 于 涌, 博士、副研究员; 唐雪松, 博士生、讲师

收稿日期: 2006-02-22 **E-mail:** chenc_xi@163.com

中所有的类变换为只有属性或者只具有操作的形式。即通过聚合关系将既有属性又具有操作的类变换为仅包含操作的父类和一个或多个包含属性且与父类有聚集关系的子类。

(3)确定 CPN 的结构 :1)用类图中所有类的属性构成全局声明结点;2)构造层次化的 CPN :

在类图中为每个交互类创建一个转移;

为每个关联类或聚集类创建一个位置,分配适当的颜色集;

在转移和位置之间用活动图创建弧(注意类图中关联的数量和可执行模型转移之间位置数量应是一一对应的);

为每个转移创建一个子页,为转移中每个操作创建一个子页中的转移,分配输入、输出和 I/O 端口位置;基于活动图创建弧,加入弧标题和监护函数;

为每个代表聚集类的位置指定初始标记。

3 基于 UML 建模的 C3I

基于 UML 的 C3I 系统建模实践过程遵循 Rational 统一过程(Rational Unified Process, RUP)的核心思想和基本原则,即以 Use Case(用例)驱动的、体系构架为核心的迭代化面向对象分析和设计过程^[4]。

开发过程^[5]的起点是充分理解系统业务过程,获得一套业务领域的词汇,形成作战概念和系统初步类图,再开发用例并用交互动态图来描述用例场景执行步骤序列,填充所建初步类图的属性与方法以完善类图。用例图、交互动态图和类图构成了光电干扰武器系统中 C3I 的逻辑模型,再利用构件图和配置图完成物理设计。由于从 UML 静态产品到可执行 CPN 动态产品映射只需类图与活动图,因此略去需求收集和开发用例、交互动态图、构件图与配置图的开发,只列出活动图与类图。图 1 为光电干扰武器系统 C3I 系统的活动图。

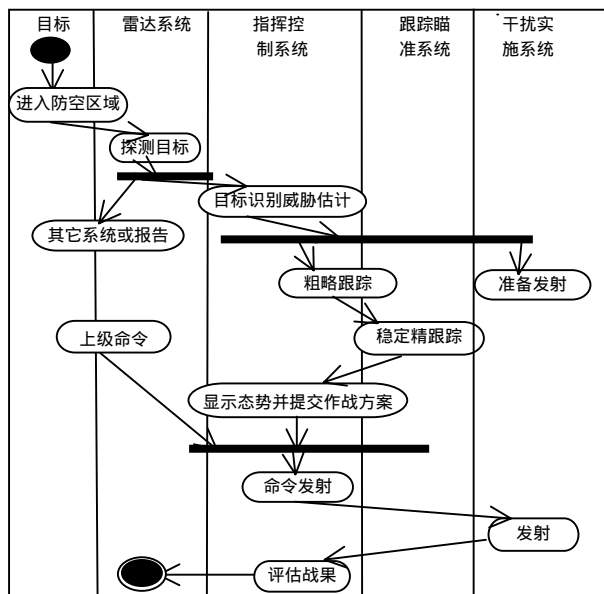


图 1 光电干扰武器系统 C3I 系统的活动图

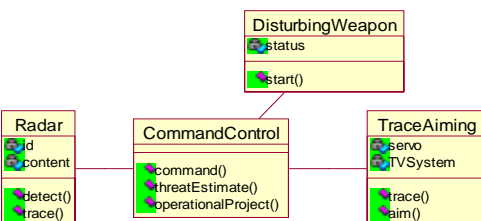


图 2 系统简化类图

雷达、指挥控制、跟踪瞄准、干扰武器系统简化类图如图 2 所示。

4 光电干扰武器系统可执行模型的建立

以指挥、控制系统(CommandControl)和跟踪、瞄准系统(TraceAiming)类图为例按上述映射方法建立可执行模型 CPN 的步骤如下:

(1)将类图进行变换,将 CommandControl 与 TraceAiming 类的关联关系转换成关联类 CommandInformation 与 TraceInformation,将包含属性与操作的 TraceAiming 类变换为由伺服系统(ServoSystem)和电视系统(TVSystem)的聚集。得到只包含指挥控制与跟踪、瞄准类的类图的片段如图 3 所示。

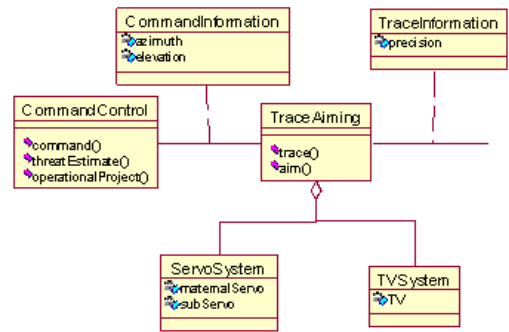


图 3 类图的片段

(2)由类图和数据字典导出 CPN 模型的全局声明节点,此由所有类的属性构成。这些结点构成颜色集和变量集如下:

Global Declaration Node

```
color azimuth=double;    var a:azimuth;
color elevation=double;  var e:elevation;
color maternalServo=int; var ms:maternalServo;
color subServo=int;     var ss:subServo;
color TV=int;           var tv:TV;
color precision=double  var pr:precision;
```

(3)构造 CPN。如图 3 所示,将交互类 CommandControl 和 TraceAiming 变换为转移,关联类与聚集类变为位置,构成顶层 CPN 图如图 4 所示。

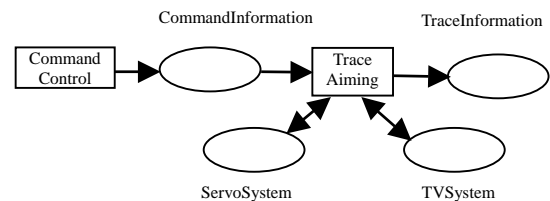


图 4 顶层 CPN 图

(4)构造 CPN 子页。图 4 只构造了最顶层 CPN 图,还要针对顶层转移对应的类图中的每一个操作做下一层的 CPN。如转移 TraceAiming 对应 TraceAiming 类中有 trace 和 aim 两个操作,每个操作都做一个下层视图。以 trace 操作为例,可以做下层转移视图如图 5 所示。

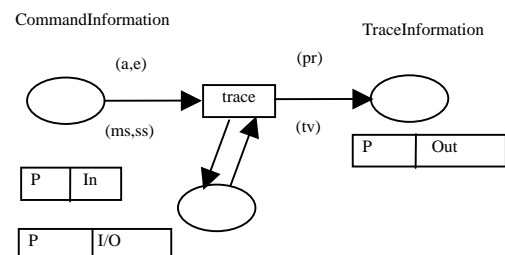


图 5 Trace 转移的 trace 操作转移子页片段

(下转第 31 页)

5 寄存器分配中的生命周期计算

YHFT-DSP/700 的编译器平台中的寄存器分配算法是一种基于着色图(Graph Coloring)的算法^[6],其处理过程如图 4 所示。

```
Register_allocation(Func fn)
{
    R_compute_dataflow_info(); /* 数据流分析*/
    R_compute_live_ranges(); /* 计算寄存器的生命周期*/
    R_register_saving_convention_selection(); /* 选择寄存器的保存方式*/
    R_construct_interference_graph(); /* 构造干涉图*/
    R_graph_coloring(); /* 图着色*/
    R_virtual_to_machine_conversion(); /* 转换成物理寄存器*/
    R_insert_spill_fill_code(); /* 插入溢出及加载代码*/
}
```

图 4 寄存器分配算法

奇异数据类型的存在对于寄存器分配中的数据流分析、生命周期(Live Range, LV)计算以及干涉图的构造都会产生影响。数据流分析时对于 long 型数据的处理,与第 4 节类似,在此不再赘述。计算寄存器的 LV 时,依然将 long 型操作数当作两个操作数(分身)看待,分别计算两个分身的 LV,而 long 型操作数的 LV 取为其两个分身的并集。以图 3 的 long 型操作数 $r3_l$ 为例,则有式(6)。

$$\begin{aligned} LV(r3_l^0) &= \{\eta, \iota, \phi, \kappa\}, \\ LV(r3_l^1) &= \{\eta, \iota, \phi, \kappa, \lambda, \mu\} \\ LV(r3_l) &= LV(r3_l^0) \cup LV(r3_l^1) = \{\eta, \iota, \phi, \kappa, \lambda, \mu\} \end{aligned} \quad (6)$$

寄存器干涉图是供着色用的无向图,图中的节点是符号寄存器,若两个寄存器的生命周期重叠($LV(x) \cap LV(y) \neq \emptyset, x, y \in \text{VirtualRegister}$),则在两个节点之间有一条边。在构造干涉图时,则需要将 long 型数据的两个分身

合二为一,在图中以一个节点表示,其生命周期取两个分身的并集。在这个图的基础上就可以正常地执行寄存器分配的其它步骤了。

6 结论

本文提出了一种实现编译器对奇异数据类型的自动支持的方法,这个方法主要包括对含有奇异数据类型的中间代码的注释、数据流和调度依赖关系的计算、寄存器分配的改进 3 个部分。我们主要以 long 型的奇异数据为例、以 YHFT-DSP/700 编译器平台为背景对算法进行了阐述。实现过程表明这种方法对编译器的改动相对较小、效率较高。而且这种方法对于处理其它类型的奇异数据也是适用的。

参考文献

- 1 陈书明,李振涛,万江华,等. “银河龙腾”高性能数字信号处理器研究进展[J]. 计算机研究与发展, 2006, 43(10).
- 2 胡定磊,陈书明,刘春林. 分簇结构超长指令字 DSP 编译器的设计与实现[J]. 小型微型计算机系统, 2006, 27(2).
- 3 Shannon C J. The IMPACT SC140 Code Generator[D]. University of Illinois, Urbana IL, 2002.
- 4 Aho A V, Sethi R, Ullman J D. Compilers: Principles, Techniques, and Tools[M]. Addison-Wesley Longman Publishing Co., Inc., 1986.
- 5 Lavery D M, Chang P P, Mahlke S A, et al. The Importance of Prepass Code Scheduling for Superscalar and Superpipelined Processors[J]. IEEE Trans. on Comput., 1995, 44(3): 353-370.
- 6 Hank R E. Machine Independent Register Allocation for the IMPACT-I C Compiler[D]. University of Illinois, Urbana IL, 1993.

(上接第 28 页)

(5)列出规则,简略规则模型如下:

if (CommandInformant=跟踪) and (azimuth 与 elevation 已知)
then (maternalServo subServo TVSystem 按 azimuth 与 elevation 进行跟踪)

重复上述过程可以将 UML 图中所有的元素均映射成 CPN 元素,从而构造可执行 CPN 模型。

5 动态可执行建模的分析

对所建的 CPN 可以执行如下步骤^[6,11]:

设 $\Sigma = (P, T; F, K, W, M, 0)$ 为网系统, M 为基网上的一标识(P 为位置的集合, T 为转移的集合, F 为 P 到 T 上的序偶, $K, W, M, 0$ 分别为容量函数、权函数和初始状态标识)。

(1)对 $t \in T, t' \cup t'$ 称为 t 的外延(t' 为前集, t' 为后集);

(2) t 在 M 下有发生权的条件是:对所有

$$p \in P, \{p \in t \Rightarrow M(p) \geq W(p, t)\} \wedge \{p \in t' \Rightarrow M(p) + W(t, p) \leq K(p)\}$$

这时也说 M 授权 t 发生(执行),记作 $M \geq t$ 。

(3)若 t 在 M 下有发生权,那么 t 就可以发生执行,发生的结果是 M 变成如下定义的新标识 M' :对所有 $p \in P$, 有

$$M'(p) = \begin{cases} M(p) - W(p, t) & \text{若 } p \in t - t' \\ M(p) + W(p, t) & \text{若 } p \in t' - t \\ M(p) - W(p, t) & \text{若 } p \in t \cap t' \\ M(p) & \text{若 } p \notin t \cup t' \end{cases}$$

t 的发生使标识变成 M' 的事件记为 $M[t \rightarrow M']$ 。 M' 为 M 的后继标识, t 在 M 下有发生权也称为 t 在 M 下满足点火条件。只要满足点火条件, Petri 就可动态执行。按照此定义可对所建 Petri 网模型进行分析与调试所建模型。

6 结论

利用已成为事实上标准并且符合人们思维习惯的 UML

进行系统建模,开发周期相对短,产品也丰富并且可直接编制程序,但其开发出来的模型是静态的,缺少动态数学分析,须编制程序并实地或同其他程序挂接仿真运行才能判断所建模型的优劣。着色 Petri 网具有强大的描述能力及严密的数学基础和多种分析手段,并且它是可以直接进行仿真的,对于评价所描述系统的功能和性能具有重要意义,但其建模过程相对繁琐且不易直接编制程序。本文所提出的光电干扰武器系统中 C3I 系统先利用 UML 建模,再进行着色 Petri 网的转换的方法,能使两种方法优势互补。该方法将同样地在其它许多领域的系统分析与建模中具有良好的应用前景。下一步将结合光电干扰武器系统 C3I 的工作性能特点、工作环境、工作对象对所建可执行模型进行数学分析与仿真,以评价所建模型。

参考文献

- 1 罗雪山,张维明. C3I 系统理论基础 C3I 系统建模方法与技术[M]. 北京:国防科技大学出版社, 2000-04.
- 2 Jensen K. Colored Petri Nets, Basic Concept, Analysis Method and Practical Use[M]. Springer-Verlag, 1997.
- 3 Wagenhals L W, Haider S, Alexander H. Levis Synthesizing Executable Models of Object Oriented Architectures[J]. Research and Practice in Information Technology, 2002, 12(6).
- 4 尤克滨. UML 应用建模实践过程[M]. 北京:机械工业出版社, 2003: 38-39.
- 5 Rumbaugh J, Jacobson I, Booch G. The Unified Modeling Language User Guide[M]. Addison-Wesley, 2001: 23-28.