

商用仿真软件的 HLA 通用集成技术研究^①

马 成^② 肖田元^③ 孙宏波 李 犁

(清华大学自动化系国家 CIMS 工程研究中心 北京 100084)

摘 要 针对商用仿真软件往往局限于特定领域,一般需要专门开发集成接口将其应用到高层体系结构(HLA)中参与协同仿真的情况,研究了商用仿真软件与 HLA 标准的集成问题,提出一种 HLA 的通用集成框架,研究了商用软件 HLA 集成引擎与动态适配技术,开发出了商用仿真软件的 HLA 通用集成平台,从软件级、模型级和任务级三个级别解决了商用仿真软件与 HLA 标准集成中的通用性问题,实现了多种仿真软件、多类模型和多仿真任务的通用集成。实际应用案例验证了该集成框架的通用性和有效性。

关键词 复杂产品,高层体系结构(HLA),协同仿真,集成引擎,动态适配

0 引 言

复杂产品开发往往涉及电、机、液、压、流等多个领域,传统的单领域仿真已经不能满足多学科耦合的复杂系统的功能和行为分析,多领域协同仿真成为现代仿真技术发展的重要趋势之一^[1]。美国国防建模与仿真办公室(DMSO)1995 年提出的高层体系结构(high level architecture, HLA)作为分布式协同仿真的基本框架,在复杂产品设计和研发中得到了广泛应用,HLA 于 2000 年成为 IEEE 标准(IEEE1516)^[2],成为协同仿真领域最为流行的标准和规范,目前国际上大多数支持分布式协同仿真的复杂系统框架,普遍是参照 HLA 来构建的^[3]。在应用中,存在商用仿真软件与 HLA 标准的集成问题,本文针对商用仿真软件关注于解决某个特定领域的建模和仿真问题,研究了如何将现有的仿真系统与仿真模型转换成符合 HLA 标准的联邦成员,实现商用仿真模型的 HLA 集成的技术,这项研究对 HLA 体系结构以及协同仿真技术在企业中的推广具有现实意义。

1 相关研究

对商用仿真软件的 HLA 集成问题的研究,从技

术实现角度大致可分为以下几类:(1)模型的 HLA 兼容性改造。如文献[4]通过将模型转化为 C 代码再进行 HLA 兼容性修改,实现将 Simulink/Stateflow 模型改造成 HLA 成员;文献[5]提出基于接口的多领域建模实现基于 HLA 的变量映射等。(2)中间件技术。如文献[6]设计了 MATLAB 与 HLA 的中间件,负责与运行支持环境(run-time infrastructure, RTI)的数据交互和时间同步;文献[7]提出了一种基于 HLA 适配器的领域模型封装方法,通过适配器机制实现标准交互接口,屏蔽多种异构仿真应用的内部技术细节。(3)扩展联邦成员架构。如文献[8]参照基本对象模型(base object model, BOM),通过组合具有可重用和互操作能力的仿真模型组件构建功能可定制和大小可伸缩的联邦成员;文献[9]提出了基于模型的联邦开发模式,通过改造现有传统联邦成员结构,实现模型在联邦成员中的即插即用。还有一些文献分析了仿真工具包与 HLA 兼容的方法,如文献[10]将仿真工具包与 HLA 兼容方法划分为 6 类,文献[11]归纳了 4 种仿真工具包与 HLA 集成的方法,并综合参照其中的两种方法实现 simulink 的 HLA 集成。

上述研究中,针对模型进行 HLA 兼容性改造的方法,受限于模型的数据结构与可修改能力,因此适用范围有限,即使针对同一仿真软件开发的模型,都需要封装和编译,方法的自动化程度较低;基于中间

① 863 计划(2009AA110302)和国家自然科学基金(60874066)资助项目。

② 女,1983 年生,博士生;研究方向:分布式仿真,复杂产品协同仿真;E-mail: mac05@mails.tsinghua.edu.cn

③ 通讯作者,E-mail: xty-dau@tsinghua.edu.cn

(收稿日期:2011-12-26)

件的 HLA 集成方法,在特定的仿真软件的 HLA 集成中取得了进展,但是面向多种商用仿真软件时,其集成过程仍需逐个开发,缺乏通用的和自动化程度较高的集成能力;扩展联邦成员架构的 HLA 集成方法,较大地促进了非 HLA 仿真系统的 HLA 兼容性改造工作,但这类方法多是概括性地针对非 HLA 仿真系统,没有充分考虑商用仿真软件的特点,在将这类方法应用于商用仿真软件时,仍有许多问题技术问题需要解决。

本文的研究目标是:商用仿真软件的 HLA 通用集成技术,适用于不同领域各类商用仿真软件开发的模型,能够屏蔽仿真软件的异构仿真算法、模型接口和数据结构,支持面向不同仿真任务、仿真模型的协同仿真系统快速组建和配置。

2 商用仿真软件的 HLA 通用集成框架

本文所提出的商用仿真软件 HLA 通用集成技术,其“通用”的含义可分为软件级、模型级、任务级三级。

(1) 软件级

典型商用仿真软件通常是面向特定领域的,具有不同的数据结构、模型结构和求解机理,提供的开放接口和二次开发能力也有所区别,因此,对某一种特定软件所设计的 HLA 集成方法往往难以规范化和重用到其他软件。想要实现通用集成框架的仿真软件级的重用,就需要充分考虑典型商用仿真软件的仿真引擎运行的特点,对 HLA 集成所需的特定功能接口进行抽象和归纳。

(2) 模型级

在基于 HLA 架构的仿真体系中,可重用和互操作的最小粒度是联邦成员级,对仿真模型的重用是通过对该仿真模型构建联邦成员的重用来体现的。要想实现仿真模型级的通用,就必然要求在联邦成员代码中不对模型进行编码,因此,需要将在联邦成员中封装仿真模型的过程规范化和参数化,实现一次开发,所有模型通用。

(3) 任务级

传统开发模式下的联邦成员虽然可以在同一联邦对象模型(federation object model, FOM)的联邦中任意地加入和退出,却难以在不同 FOM 的联邦中重用,其主要原因是在联邦成员代码中对成员仿真对象模型(simulation object model, SOM)中的数据映射和交互信息进行编码。要想实现不同仿真任务

的通用,就需要将联邦成员与 SOM 信息相分离,将发布订购等数据交互关系在仿真运行时实现动态实例化。

为满足上述三级通用的需求,本文提出了一种商用仿真软件的 HLA 通用集成框架,通过研究商用仿真软件 HLA 集成的仿真引擎以及仿真任务无关的动态适配技术,实现多种仿真软件、多类模型和多仿真任务的通用集成。商用仿真软件的 HLA 通用集成技术框架如图 1 所示。

通用集成框架由集成引擎与动态适配器两大模块组成。集成引擎包括实验参数数据、功能结构与仿真引擎三部分,主要解决模型级通用和联邦级通用的问题。动态适配器实现了对象类及其属性、交互类及其参数、订购发布关系等数据结构的动态生成,同时能够调用 RTI 提供的各项接口功能,主要解决仿真任务级通用的问题。根据功能特征不同,通用集成框架又可以分为配置层和执行层。配置层包括集成引擎模块中的参数数据和功能结构,以及动态适配器模块中的对象类交互类及发布订购数据结构,主要功能是根据具体仿真任务,选择仿真模型并加载相应商用仿真软件版本的功能函数,配置商用仿真引擎所需的实验参数,以及解析 SOM 文件,实例化对象类交互类和发布订购数据结构等,这些都是仿真执行开始前的初始化工作,配置完成后在仿真执行过程中不需改变。执行层包括集成引擎模块中的仿真引擎和动态适配器中的 RTI 接口服务,主要功能是驱动商用仿真软件求解器执行仿真和动态适配器仿真逻辑的推进,实现仿真执行过程中的输入输出数据与发布订购对象类交互类之间的映射,并调用 RTI 接口功能函数实现基于 HLA 的协同仿真过程。

基于该通用集成框架,面向具体仿真应用的集成过程包括:根据仿真任务概念模型交互式配置实验参数、模型描述信息,并在对象模型建模工具的辅助下生成 SOM、FOM 文件;在集成引擎中,根据实验参数完善商用仿真软件的初始化配置;根据模型描述信息,获知模型对应的商用仿真软件,选择对应版本的功能函数以及加载指定的模型;在动态适配器中,推理机自动解析 SOM 文件,获取对象类交互类以及发布订购关系等信息,动态生成适配器联邦成员类相关数据结构的实例。所有配置完成后,则可启动仿真引擎,它分别调用集成引擎中的功能函数和动态适配器中的 RTI 接口服务函数,控制仿真执行的推进,并实现该过程中模型输入输出数据与 HLA

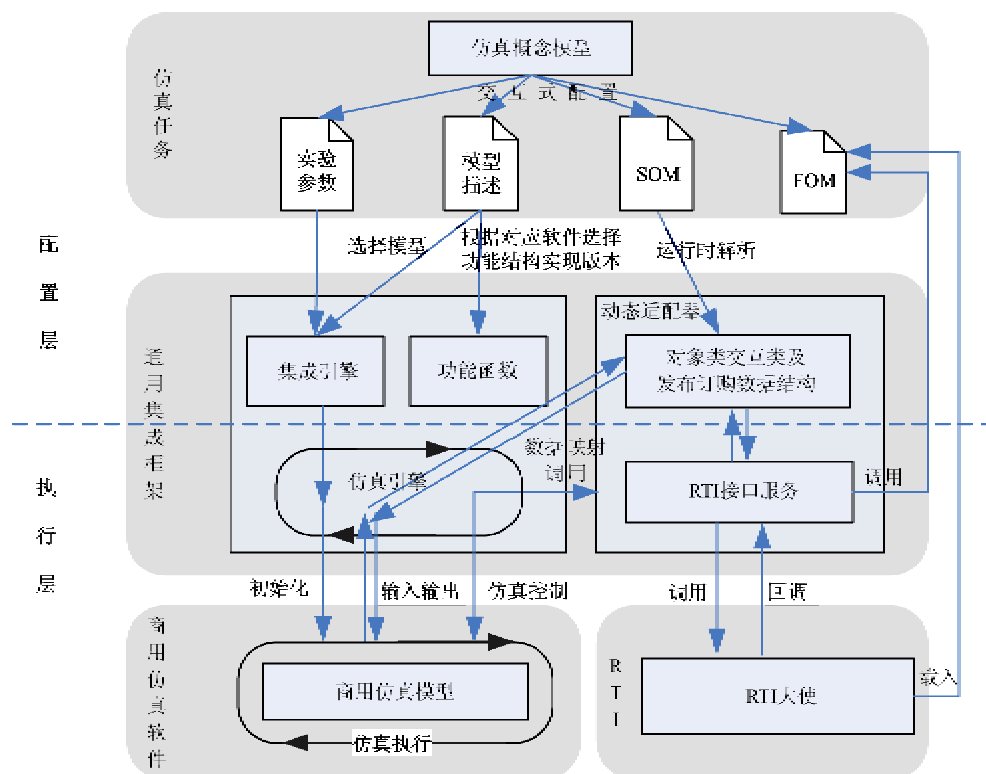


图1 商用仿真软件的HLA通用集成技术框架

发布订购数据之间的映射直至仿真结束。

3 集成引擎

集成引擎作为驱动仿真推进的源动力,主要解决软件级通用和模型级通用两方面的问题。

3.1 软件级通用的解决方案

不同商用仿真软件,往往具有不同的数据结构和求解机理,提供的开放接口和二次开发能力也有所区别,对一种软件设计的 HLA 集成方法往往难以重用到其它软件。因此商用仿真软件仿真引擎的封装一直是 HLA 仿真集成研究所关注的重点和难点。

本文针对典型商用仿真软件协同仿真的特点,设计了一种用于商用仿真软件的 HLA 通用集成引擎,它屏蔽了不同软件仿真机理的内部实现,将其对外交互的接口归为实验参数数据接口和功能函数接口两部分。

实验参数数据接口是商用仿真软件外部接口之一。实验参数数据与模型无关,在同一仿真类型的不同仿真软件间具有一定的通用性,如连续系统仿真的实验参数包括仿真运行时间、仿真步长、求解器名称、误差精度等。但实验参数也受具体仿真软件的约束,如特定仿真软件支持的求解器和算法类型

有限,因此在对实验参数数据的配置中会根据模型对应的商用仿真软件来限定实验参数可取的有效值范围。

功能函数接口是将商用仿真软件二次开发应用程序编程接口(API)中HLA集成所需要的功能标准化为接口函数,供仿真引擎模块调用。在对典型商用仿真软件的仿真引擎进行集成的过程中,实现模型配置、仿真推进和信息交互等功能往往需要利用商用仿真软件提供的二次开发API接口。HLA集成所必需的API功能与其对应的标准化函数主要包括:

- (1) 在每一积分步时间点上从工作空间获取模型输出变量值 `GetValue(…)`;
- (2) 在每一积分步时间点上以新属性值为指定输入变量赋值 `SetValue(…)`;
- (3) 仿真软件的启动和模型的初始化 `InitializeModel(…)`;
- (4) 模型仿真结束后的有关处理以及仿真软件的关闭 `Terminate(…)`;
- (5) 利用仿真软件实现模型的仿真步进 `RequestTimeAdvance(…)`。

为了将商用仿真软件进行封装,使其能参与基于HLA的协同仿真,商用仿真软件的API接口需要

直接或间接支持上述 5 个标准化函数。根据 API 开放性的不同,在考虑可封装引擎的商用仿真软件时需注意以下两种情况:

(1) 商用仿真软件的 API 接口较为开放,如 MATLAB 的 S-Function 函数接口、ADAMS 的用户自定义子程序接口等。对这类商用仿真软件,可以直接实现上述标准化函数。

(2) 商用仿真软件自身 API 接口较为局限,但可以通过第三方软件或者二次开发实现上述标准化函数,如 Saber/Simulink co-simulation tool 实现了 Saber 模型在 MATLAB 环境中的封装。对这类商用仿真软件,可在实现该第一类商用仿真软件引擎封装的基础上,利用该接口,间接地实现上述 5 个标准化函数。

以动力学仿真软件 Adams 为例,其结构功能函数的定制实现方法如下:

(1) `InitializeModel(...)`、`Terminate(...)` 函数:基于 Solver 命令和仿真脚本驱动和控制 Adams 引擎的启动关闭和模型的初始化;

(2) `GetValue(...)`:应用 REQSUB 用户自定义子程序,该子程序在输出步长点上由 Adams 引擎自动调用,实现在步长点上输出数据;

(3) `SetValue(...)`:应用 VARSUB 子程序,该子程序会在 ADAMS 求解器每次请求输入变量值时被自动调用,实现向 Adams 模型输入数据,更改模型参数;

(4) `RequestTimeAdvance(...)`:与仿真引擎函数共享内存空间,采用互斥信号量的方式实现仿真执行的暂停与推进。

3.2 模型级通用的解决方案

模型级通用的要求是:根据仿真任务选择了某种商用仿真软件所开发的模型或同一商用仿真软件开发的不同模型,可以不需要修改模型,或只需自动或半自动地进行少量修改,即可实现模型在 HLA 通用集成框架上的即插即用。

本文的商用仿真软件 HLA 通用技术模型级的通用主要是指在同一仿真软件开发的不同模型,可采取统一的通用方案来封装,不需要针对不同模型重新封装和修改代码,在根据仿真任务切换不同模型时能够实现即插即用。

在本文的 HLA 通用集成框架中,模型级通用主要是通过模型信息描述及其解析来实现的。模型信息描述包括仿真模型名、模型运行的仿真软件名、仿真模型所在路径、模型输入变量组和模型输出变量

组等。模型信息描述是仿真引擎的输入之一,对同一商用仿真软件的不同模型,只需要修改仿真模型名和仿真模型所在路径,仿真引擎就可以实现驱动商用仿真软件求解器和加载指定模型。模型输入和输出变量组与商用仿真软件有关,有些商用仿真软件输入输出端口包含在模型数据结构中(如 Matlab/Simulink),有些则要求必须再次明确指定特定变量为输入或输出变量(如 Adams),因此模型输入输出变量组是模型初始化功能函数 `InitializeModel(...)` 的入口参数之一,在该函数中实现了针对后一类商用仿真软件实现模型的自动修改和配置。

3.3 仿真引擎

仿真引擎逻辑流程图如图 2 所示。集成引擎中的仿真引擎模块,是联系集成引擎与动态适配器,以及联系通用仿真框架与商用仿真软件仿真执行的核心部分。其功能主要包括:

(1) 通过入口参数接收实验参数和模型描述并据此实例化参数数据结构,加载对应特定商用仿真软件版本的功能函数;

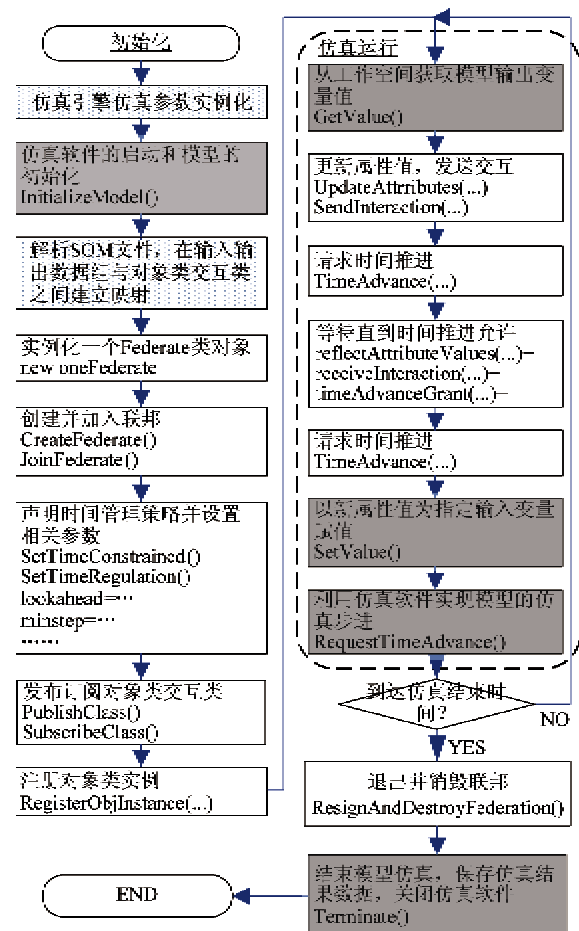


图 2 联邦成员主程序流程图

(2)遍历和解析 SOM 文件,实例化适用于本次仿真任务的对象类交互类发布订购关系数据结构;

(3)在实验参数中的输入输出数据组和对象类交互类之间建立映射;

(4)调用功能函数,实现商用仿真软件求解器启动、模型载入、模型初始化、控制仿真推进、模型数据输入输出、结束仿真推进、保存仿真结果数据、关闭仿真软件求解器等;

(5)调用动态适配器模块 RTI 接口服务函数,实现 HLA 协同仿真所需功能;

(6)在仿真模型输入输出数据组与对象类交互类数据结构间完成数据存储与事件映射。

其中,功能函数的调用在图中用灰色方框表示,动态适配器所包含的 RTI 接口服务的调用在图中用白色方框表示,其它代码用点阵方框表示。

4 动态适配技术

动态适配器作为 HLA 集成的基础和支撑,主要解决仿真任务级通用的问题。动态适配器的功能包括通过推理机解析 SOM 文件,动态实例化 HLA 对象类交互类和发布订购关系数据结构,以及与 RTI 交互的核心服务功能。动态适配器由对象类交互类发布订购数据结构和 RTI 接口服务模块两部分组成。

对象类交互类及发布订购关系数据结构如图 3 所示,核心类是 Federate 联邦成员类,在联邦成员类中采用标准模板库(standard template library, STL)的 list 双向链表结构,定义了可动态构建的发布对象类组、订购对象类组、发布交互类组和订购交互类组,其链表单元分别为对象类数据结构和交互类数据结构。对象类数据结构中定义了属性类型链表和对象类实例链表,交互类数据结构中定义了参数类型链表,属性数据类型和参数数据类型中定义了存储输入输出变量值的变量值链表,可存储指定长度的历史数值,能够有效支持步长不同以及变步长的协同仿真算法。在联邦成员类的构造函数中通过推理机解析入口指定的 SOM 文件,能够动态实例化上述发布订购对象类交互类等链表结构。

RTI 接口服务模块负责动态适配器与 RTI 通信,其实现应用了代理机制,如图 4 所示。联邦成员 Federate 类继承了 RTI 向联邦成员的开发者提供的 libRTI 类库中的抽象类联邦成员大使 FederateAmbassador 类,联邦成员大使类规定了每一联邦成员必须提供并实现的回调函数,提供了一种 RTI 进行

Federate:FederateAmbassador	ObjectClass
char* federationName	char* objectClassName
char* federateName	RTI::ObjectClassHandle
int federateHandle	objectClassID
char* FEDFile	list<Attribute> attributes
char* SOMFile	list<ObjectInstance> objectInstances
list<ObjectClass>	
publishObjectClasses	Attribute
list<ObjectClass>	char* attributeName
subscribeObjectClasses	char attributeType
list<InteractionClass>	RTI::AttributeHandle
publishInteractionClasses	attributeID
list<InteractionClass>	list<Variable> variables
subscribeInteractionClasses	insert(Value,...)
RTI::Time lookahead	getValue(...)
RTI::Time minstep	
RTI::RTIAmbassador rtiAmb	InteractionClass
.....	char* interactionClassName
Federate(char* SOMFile)	RTI::InteractionClassHandle
CreateFederation()	interactionClassID
ResignAndDestroyFederation()	list<Parameter> parameters
JoinFederation()	Parameter
SetTimeConstrained()	char* parameterName
SetTimeRegulation()	char parameterType
PublishClass()	RTI::ParameterHandle
SubscribeClass()	parameterID
UpdateAttributes(...)	list<Variable> variables
SendInteraction(...)	Variable
TimeAdvance(...)	char* attributeName
RegisterObjInstance(...)	char attributeType
reflectAttributeValues(...)+	RTI::AttributeHandle
receiveInteraction(...)+	attributeID
timeAdvanceGrant(...)+	list<Variable> variables
discoverObjectInstance(...)+	
.....	

图 3 动态适配器数据结构

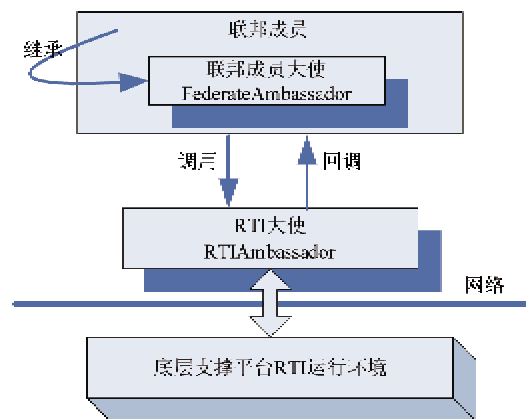


图 4 动态适配器与 RTI 通信的代理机制

联邦通知并向成员提供信息的机制, Federate 类重载了这些回调函数。同时, Federate 类实例化了一个 RTI 大使 RTIAmbassador 类的对象作为本类的一个对象成员,并通过这个对象成员来调用 RTI 提供的标准功能函数。Federate 类中对联邦成员初始化、时间管理和同步机制设定、创建并加入联邦、发布订阅、请求时间推进,更新和反射对象属性值,发送和接收交互,退出并撤销联邦等通用功能进行了归纳和模式化封装。

5 应用案例

本文结合 863 项目“支持 400km/h 高速检测列车研制的协同仿真平台建设和应用”,应用商用仿真软件的 HLA 通用集成技术,开发了针对高速列车的 HLA 仿真通用集成平台,以高速检测列车为对象进行了应用和验证。协同仿真对象为八车厢动车组,包括由 4 类商用仿真软件分别开发的共 10 个仿真模型。其中 Adams 对前四车车体分别进行建模

仿真,Vampire 对后四车整体模型进行建模仿真, Matlab 对一二车、二三车、三四车和四五车的车间钩缓进行建模仿真,Abaqus 对转向架框架进行有限元分析。模型部署和交互图如图 5 所示。在案例验证中,应用本文开发的商用仿真软件 HLA 通用集成平台实现了 Adams、Vampire、Matlab 和 Abaqus 四种仿真软件的 HLA 集成。限于篇幅,这里以 Adams 软件为例,说明应用本文 HLA 通用集成技术进行商用仿真软件 HLA 集成的过程。

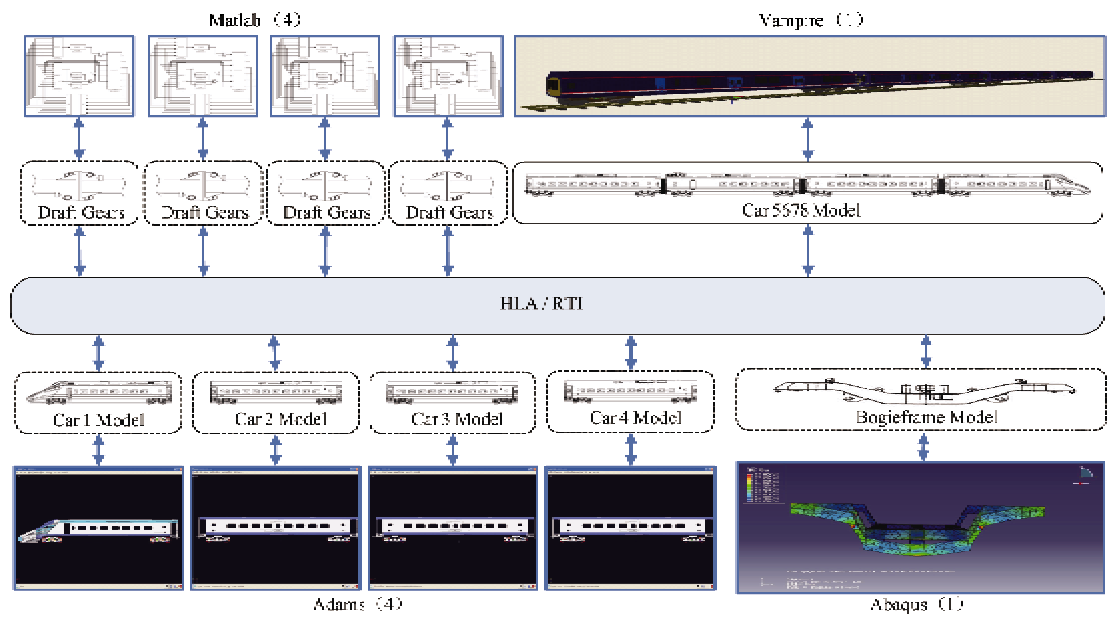


图 5 八节列车多学科建模仿真模型部署交互图

- (1) 根据仿真任务概念模型,配置实验参数。
本例中仿真时间 20s,步长 0.01s,求解器算法 Newmark,相对误差 1.0E-005
- (2) 根据仿真任务概念模型,选择模型交互式提示用户完成模型描述信息的填写。本例中模型信息如图 6 所示。

```
<Model name="car1">
  <OutVariableList value="car1*car1rear_dx
    car1*car1rear_dy car1*car1rear_dz
    car1*car1rear_vx car1*car1rear_vy
    car1*car1rear_vz car1*uel_SS_fx
    car1*uel_SS_fy car1*uel_SS_fz"/>
  <InVariableList value="chegou12*chegou12front_fx
    chegou12*chegou12front_fy
    chegou12*chegou12front_fz"/>
  <OutEventList value=""/>
  <InEventList value=""/>
  <Path value="C:\car400"/>
  <SoftwareName value="Adams"/>
</Model>
```

图 6 模型信息描述

- (3) 根据模型描述信息,仿真引擎后台调用模型初始化功能函数 InitializeModel(...),自动完成修改指定模型,标明输入输出变量的工作。Adams 模型中添加的语句如下:
! adams _view _name = PINPUT_1'
PINPUT/1, VARIABLES = 9, 10, 11
! adams _view _name = POUTPUT_1'
POUTPUT/1, VARIABLES = 3, 4, 5, 6, 7, 8, 27, 28, 29
- (4) 根据模型描述信息,仿真引擎自动加载商用仿真软件 Adams 版本的功能函数,Adams 功能函数的实现见 4.1 节。
- (5) 提示用户选择已存在的 SOM,若没有可用的文件,则根据模型描述信息自动生成 SOM 文件,自动生成的 SOM 片段如图 7 所示。

```

<?xml version="1.0" ?>
<objectModel DTDversion="1516.2" type="SOM" version="1.0 Alpha" other="Could
be something" references="CIM601" pocOrg="CIM601" pocName="Someone"
sponsor="Someone" appDomain="Simulation" purpose="Test" date="2011-07"
pocPhone="NA" pocEmail="NA" FEDPath="C:\Sim\CGuiAccess\W"
ModelNumber="10" FederationName="Car863" name="car1"
ModelPath="C:\car400\ ">
<objects>
  <objectClass name="HLAobjectRoot" sharing="Neither">
    <attribute name="HLAprivilegeToDeleteObject" dataType="NA"
updateType="NA" updateCondition="NA" ownership="NoTransfer"
sharing="Neither" dimensions="NA" transportation="HLAreliable"
order="TimeStamp" />
    <objectClass name="Simulation" sharing="PublishSubscribe" semantics="NA">
      <objectClass name="chegou12" sharing="Subscribe" semantic="NA">
        <attribute name="chegou12front_fx" sharing="Subscribe" semantic="NA" />
        <attribute name="chegou12front_fy" sharing="Subscribe" semantic="NA" />
        <attribute name="chegou12front_fz" sharing="Subscribe" semantic="NA" />
      </objectClass>
      <objectClass name="car1" sharing="Publish" semantic="NA">
        <attribute name="car1rear_dx" sharing="Publish" semantic="NA" />
        <attribute name="car1rear_dy" sharing="Publish" semantic="NA" />
        <attribute name="car1rear_dz" sharing="Publish" semantic="NA" />
        <attribute name="car1rear_vx" sharing="Publish" semantic="NA" />
        <attribute name="car1rear_vy" sharing="Publish" semantic="NA" />
        <attribute name="car1rear_vz" sharing="Publish" semantic="NA" />
        <attribute name="uel_SS_fx" sharing="Publish" semantic="NA" />
        <attribute name="uel_SS_fy" sharing="Publish" semantic="NA" />
        <attribute name="uel_SS_fz" sharing="Publish" semantic="NA" />
      </objectClass>
      <objectClass name="synObjectClass" sharing="PublishSubscribe"
semantic="NA">
        <attribute name="synAtt" sharing="PublishSubscribe" semantic="NA" />
      </objectClass>
    </objectClass>
  </objects>
  <interactions>
    <interactionClass name="HLAinteractionRoot" sharing="Neither"
dimensions="NA" transportation="HLAreliable" order="TimeStamp" />
  </interactions>

```

图 7 自动生成的 SOM 片段

(6) 动态适配器程序将 SOM 文件作为入口函数传入, 基于推理机解析 SOM 文件, 实例化发布订阅对象类交互类链表等相关数据结构。

(7) 仿真引擎驱动商用仿真软件求解器运行, 并在仿真模型输入输出数据组与对象类交互类数据结构间完成数据存储与事件映射。

(8) 仿真引擎调用动态适配器模块 RTI 接口服务函数, 实现与 RTI 通讯。

(9) 执行仿真过程直至仿真结束条件到达。

(10) Adams 仿真结果文件自动存储到模型所在路径, 可以用 Adams 后处理模块查看分析。

图 8 是对八节列车多学科异构仿真模型进行步长为 0.01s, 仿真时间为 20s, 列车运行速度 20m/s 的过弯道仿真得到的脱轨系数列车性能指标。案例通过对四种仿真软件进行 HLA 集成, 实现了不同仿真软件在分布式环境下的协同仿真, 为用户分析不

同仿真软件的仿真算法的差异提供了便利条件, 为高速检测列车的设计和研发提供支持。仿真结果验证了本文 HLA 通用集成技术在应用于多种仿真软件协同仿真时集成的通用性和有效性。

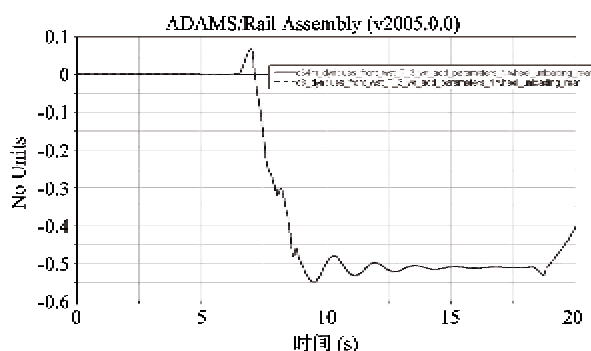


图 8 第三节列车前转向架后轮右轮脱轨系数

6 结 论

本文针对商用仿真软件与 HLA 标准的集成问题,提出一种 HLA 的通用集成框架,研究了商用软件 HLA 集成引擎与动态适配技术,开发出了商用仿真软件的 HLA 通用集成平台,从软件级、模型级和任务级三个级别解决了商用仿真软件与 HLA 标准集成中的通用性问题。在此基础上,本文应用该通用集成技术,在实际案例中实现了多种商用仿真软件的 HLA 仿真集成,验证了该集成框架在商用仿真软件 HLA 封装上的通用性和有效性。

本文开发的商用仿真软件 HLA 通用集成平台目前已经在连续系统仿真和离散系统仿真软件上实现了对 HLA 集成的支持,如何在现有基础上支持多商用软件实现的仿真模型,是需要进一步研究的内容。

参考文献

- [1] 王克明,熊光楞. 复杂产品的协同设计与仿真. 计算机集成制造系统,2003,9(2):15-19
- [2] IEEE Computer Society. IEEE standard for modeling and simulation (M&S) high level architecture (HLA)-Federate Interface Specification (IEEE Std 1516.1-2000). NewYork: The Institute of Electrical and Engineers, 2001
- [3] Xu Y, Xiao T, Liang C, et al. Federated integration of networked manufacturing service platforms. Advanced Engineering Informatics. 2008, 22: 317-327
- [4] 田新华,冯润明,翁千飞等. 一种将 SIMULINK/Stateflow 模型改造成 HLA 成员的方法. 系统仿真学报, 2002,14(7):883-886
- [5] 陈晓波,熊光楞,郭斌等. 基于 HLA 的多领域建模研究. 系统仿真学报,2003,15(11):1537-1542
- [6] 郭斌,熊光楞,陈晓波等. MATLAB 与 HLA/RTI 通用适配器研究与实现. 系统仿真学报,2004,16(6): 1275-1279
- [7] 王克明,熊光楞. 一种基于 HLA 适配器的领域模型封装方法. 高技术通讯,2005,15(2):34-38
- [8] 龚建兴,钟蔚,黄健等. 基本对象模型(BOM)在 HLA 仿真系统中的应用. 系统仿真学报,2006,18(增刊 2):327-331
- [9] 龚建兴,王达,邱晓刚等. HLA 联邦成员中模型的重用性研究. 系统仿真学报,2005,17(11):2652-2655
- [10] 战玉芝,刘怀勋,李佳. HLA 及其在研究传统工具协同仿真中的应用. 见:系统仿真技术及其学术交流会议,2006. 282-285
- [11] 黄志军,曾斌. 基于 HLA 的传统仿真软件集成技术研究. 系统仿真学报,2003,15(7):1017-1020

Research on HLA-compatible general integration for commercial simulation software

Ma Cheng, Xiao Tianyuan, Sun Hongbo, Li Li

(Nation CIMS Engineering Research Center, Tsinghua University, Beijing 100084)

Abstract

Aiming at the problem that commercial simulation software usually can not be applied to the high level architecture (HLA) to participate in collaborative simulations because it is designed for some specific domains, this paper studies HLA's compatible, general integration with commercial simulation software, and firstly proposes a general HLA-compatible integration framework for commercial simulation software. Its techniques of HLA-compatible integration engine and dynamic adapter are analysed in details. A platform is developed to give the general solution of HLA-compatible integration from the three levels of the software level, the model level and the task level. Based on this platform, the integrated of a variety of simulation software, models and simulation tasks can be achieved. An industrial application case shows the success of the HLA-compatible integration of Matlab, Adams, Vampire and Abaqus, which demonstrates the feasibility of the proposed framework.

Key words: complex product, high level architecture (HLA), collaborative simulation, integration engine, dynamic adapter