

# 基于狄拉克符号的异构模型集成方法

王晓路, 贾长伟, 张升升

(中国运载火箭技术研究院, 北京 100076)

**摘要:** 针对数字工程中异构模型集成仿真的问题, 提出了一种基于狄拉克符号的系统建模方法。该方法可以屏蔽子系统的建模逻辑和模型实现差异, 描述全系统的集成模型和动力学演化。利用类 TNEA 中间件提供的状态定义和方法调用机制, 可以将异构模型映射为具有相同建模逻辑和软件接口的对象模型, 构成数字主线, 开展虚实融合系统集成试验。该方法可以支撑航天“簇”演进模式下的新技术快速插入、新系统原位替代和新状态快速验证, 并针对数字航天开展全要素虚实融合试验, 在虚拟环境中摸边探底, 验证核心指标。

**关键词:** 虚实融合; 异构模型集成; 数字化仿真

中图分类号: V416.6

文献标志码: A

文章编号: 2096-4080 (2024) 02-0006-09

## Heterogeneous Model Integration Method Based on Dirac Notation

WANG Xiaolu, JIA Changwei, ZHANG Shengsheng

(China Academy of Launch Vehicle Technology, Beijing 100076, China)

**Abstract:** This article proposes a system modeling method based on Dirac notation for the integrated simulation of heterogeneous models in digital engineering. This method can mask the differences of sub-system modeling logic and implementation, and describe the integrated system model and dynamic evolution. Using the state definition and method invocation mechanism provided by the TNEA-like middleware, heterogeneous models can be mapped to object models with the same modeling logic and software interface, forming the digital thread, and conducting integrated testing of virtual-real systems. It can support the cluster evolution mode of new technologies being quickly inserted, new systems being in-situ replaced, and new states being quickly verified for aerospace. It can conduct full-element virtual-real fusion tests on digital aerospace, explore the boundaries in the virtual environment, and verify core indicators.

**Key words:** Heterogeneous model integration; Virtual-real fusion; Digital simulation

### 0 引言

为实现新技术快速插入、新系统原位替代、新状态快速验证的航天“簇”演进模式<sup>[1-2]</sup>, 适应新技术的涌现性、动态性, 国外提出了以数字工程计划<sup>[3]</sup>为代表的数字化技术体系。利用权威真

相源, 整合系统工程全部阶段的可用知识<sup>[4]</sup>, 为复杂产品提供全生命周期的集成视图, 开展虚拟化集成验证。数字主线是一种可连接数据流的通信框架<sup>[5]</sup>, 它可以在获取关键数据的同时, 驱动产品模型双向同步, 最终实现模型在产品全生命周期内的有效传递、引用与反馈, 随时站在系统全局

收稿日期: 2023-08-06; 修订日期: 2024-01-03

基金项目: 国家部委基础研究项目 (XX202105ZJ0001)

作者简介: 王晓路 (1982—), 男, 博士, 高级工程师, 主要研究方向为先进试验方法与验证技术。

的角度,通过虚拟化手段对设计结果进行验证。

数字主线需要建立一种具有广泛适应性的工程对象互操作架构,兼容封装各种虚拟或实物样机,使全部子系统的设计结果具备互操作性,可以集成为一个整体的数字样机,并开展跨阶段及跨虚实的集成、演绎、验证<sup>[6]</sup>。

参照样机系统模型演进框架<sup>[7]</sup>归纳的航天全生命周期样机模型类型,以狄拉克符号<sup>[8]</sup>作为基础数学架构,建立航天系统模型的统一建模范式。以类 TNEA 中间件作为样机模型统一封装的软件实现方法<sup>[9]</sup>,形成基于狄拉克符号的异构模型集成方法。基于该方法构建的虚实融合验证环境,具有基于开放式架构和标准化集成协议的模型集成机制,可以使航天全生命周期的模型具有足够的互操作性,支持航天领域基于数字主线开展全要素虚实融合试验,在虚拟环境中摸边探底,验证核心指标。

## 1 总体方案

将航天研制过程中形成的各类异构样机集成为一个整体的数字平台,主要有两类技术路线。

第 1 种是基于统一建模语言建立样机模型,在统一软件平台上进行多领域统一建模<sup>[10]</sup>。其优点是能够形成同构的系统模型库,具有良好的互操作性,同时可以积累模型资产。缺点是需要改造各个专业的工作流,例如将控制系统的传统建模工作流改造为 Modelica<sup>[11]</sup>工作流。

第 2 种是基于异构网关转换封装各类模型样机的交互数据接口,基于信息总线将全系统集成成为一个整体<sup>[12]</sup>。其优点是可以不改变现有分系统研制的工作流,仅对其研制形成的各类样机进行封装,对研制过程的干扰较小。但是该途径需要解决子系统统一封装、数据同步和运行解耦等分布式集成试验的技术问题。

以第 2 种技术路线为基础,基于狄拉克符号的数学语言,将多专业异构样机的交互接口映射为描述系统状态的态,将交互业务映射为描述系统状态演化的相互作用算符。基于类 TENA 中间件实现数据接口,使全部子系统样机映射的对象模

型,可以在虚拟空间中发生相互作用,并随时间演化,最终完成试验。虚实融合系统集成试验的总体方案如图 1 所示。典型航天装备研制可以按照专业分为结构、动力、控制等多个子系统,在此基础上,基于狄拉克符号建立每个子系统对应的数字模型。

1) 在描述每个子系统的物理参量中,根据集成试验所关心的学科领域(如力学、电学、热学等),选取尽可能正交的物理参量,构建系统的态矢量,并基于类 TENA 中间件的对象模型,构建该态矢量的数据结构。无论子系统的样机是何种状态(实物样机、数据样机、机理模型样机等),均可通过传感器采集、数据提取或模型计算等手段,获取相应的状态参量值,并映射到对象模型的数据结构。

2) 子系统之间互相影响的交互机制和子系统随时间变化的演化规律,可以通过对实物样机的状态采集、对数据样机的插值/外推或对机理模型的演化计算等途径获得。将其进一步封装映射为类 TENA 中间件的方法类对象模型,可以在集成仿真验证的过程中对态矢量的数据结构进行交互和演化相关的仿真计算操作。

基于这种模式,可以屏蔽不同类型样机的计算机理差异,实现统一的计算机操作。

## 2 基于狄拉克符号的统一建模方法

狄拉克符号在量子力学中广泛应用于描述量子态及其相互作用,是描述量子多体问题的标准符号,并且可以推广应用到信息或宏观物理等领域。基于狄拉克符号的统一建模方法,是实现全生命周期各类异构样机模型集成互操作的数学基础,具有数学完备、物理通用、系统易解耦、时间离散、屏蔽异构差异性、软件实现直观等特点。以狄拉克符号作为基础的数学语言,使用态来描述系统对象的全部状态参量,用算符来描述系统之间的相互作用和动力学演化过程。航天领域的实物样机、数据样机和机理模型样机,其状态参量被直接封装为态,态的演化过程或交互行为被封装为算符。

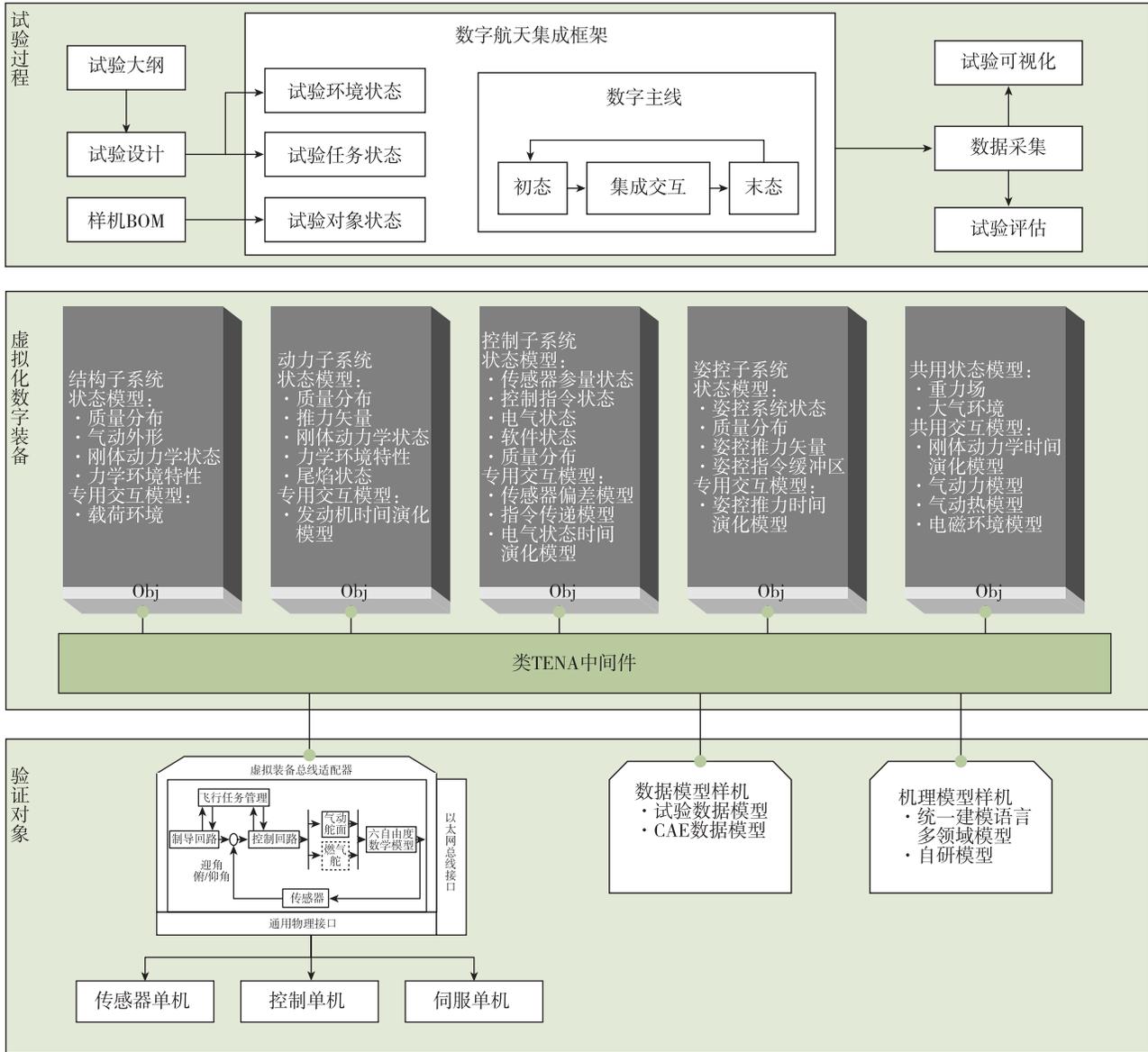


图 1 总体架构

Fig. 1 Overall architecture

### 2.1 态

系统工程问题中，在任意时刻  $t$ ，系统中的元素具有确定的状态，简称为态，记为  $|A(t)\rangle$ ，在与时间无关的场合可简写为  $|A\rangle$ 。 $|A(t)\rangle$  是一个希尔伯特空间广义坐标系的态矢量，在计算机中为一个变量池或结构体。

虚实融合集成试验通常需要对多个系统或一个系统的多个子系统建模。假定系统  $S$  由系统  $A$ 、系统  $B$ 、系统  $C$  组成，可以记为

$$|S\rangle = |A\rangle + |B\rangle + |C\rangle = \begin{pmatrix} |A_1\rangle \\ \vdots \\ |A_i\rangle \\ |B_1\rangle \\ \vdots \\ |B_j\rangle \\ |C_1\rangle \\ \vdots \\ |C_k\rangle \end{pmatrix} \quad (1)$$

这里  $|A\rangle$ ,  $|B\rangle$ ,  $|C\rangle$  分别为对系统  $A$ ,  $B$ ,  $C$  的状态描述, 参量包括几何形状描述、质量分布描述、动力学属性、运动学属性、温度、压强等。 $i, j, k$  均为参量索引序号, 在计算机系统中,  $|A\rangle$  实际上是一个具有完整索引的变量池, 包含了子系统  $A$  的所有状态数据。需要注意,  $|A_i\rangle$  不一定是一个数值, 也可以是一个指向复杂数据结构的指针或索引。算符在计算机实现的过程中, 可以对不同的  $|A_i\rangle$  采用不同的处理算法, 例如坐标变换、特征提取、动力学演化等。

## 2.2 相互作用

相互作用是物体状态改变的原因, 系统的状态因为相互作用而随时间演化。

系统中, 元素或子系统之间的相互作用使用算符来表示, 例如  $A$  对  $B$  的作用记为  $\hat{H}(A-B)$ , 元素  $A$  为相互作用的使能者, 元素  $B$  为相互作用的接受者。和基本物理学问题不同, 在工程中, 通常  $\hat{H}(A-B)$  不等价于  $\hat{H}(B-A)$ , 例如, 控制系统的输入信号和输出信号。

某一元素  $A$  经过元素  $B$  的作用之后, 状态改变可以表示为

$$|A(t+\Delta)\rangle = \hat{H}(B-A)|A(t)\rangle \quad (2)$$

即元素  $A$  在  $t$  时刻的状态  $|A(t)\rangle$ , 经过  $\hat{H}(B-A)$  的作用后变化为  $t+\Delta$  时刻的状态  $|A(t+\Delta)\rangle$ 。

系统中的相互作用算符, 在集成仿真的过程中通过“方法”<sup>[13]</sup>来实现, “方法”是一个计算机语言的函数, 给定输入, 计算得到输出。描述相互作用算符的“方法”, 其输入与输出的变量集是一致的。

假定系统  $S$  从初始状态  $|S(t)\rangle$  经过一个设计预期的物理过程  $\hat{H}$  之后, 其状态变化为  $|S(t+\Delta)\rangle$ , 对于航天领域来说, 其速度、质量、温度分布、几何形状等参量会发生变化。

在这一过程中,  $\hat{H}$  与子系统  $A, B, C$  的关系可以表示为

$$\begin{aligned} |S(t+\Delta)\rangle &= \hat{H}|S(t)\rangle \\ &= \begin{bmatrix} \hat{H}_{AA} & \hat{H}_{AB} & \hat{H}_{AC} \\ \hat{H}_{BA} & \hat{H}_{BB} & \hat{H}_{BC} \\ \hat{H}_{CA} & \hat{H}_{CB} & \hat{H}_{CC} \end{bmatrix} \begin{pmatrix} |A(t)\rangle \\ |B(t)\rangle \\ |C(t)\rangle \end{pmatrix} \end{aligned} \quad (3)$$

式中,  $\hat{H}_{AA}, \hat{H}_{BB}, \hat{H}_{CC}$  是子系统  $A, B, C$  内在相互作用的过程, 例如惯性运动、燃料消耗、控制计算机解算等。 $\hat{H}_{AB}$  类的矩阵元代表子系统之

间的相互作用, 例如控制信息的传递、物体之间的受力、温度的传导等。在计算机中,  $\hat{H}_{AA}$  实际上是一个函数, 其运行的输入是  $|A(t)\rangle$  ( $A$  在某一时刻的态矢量)。

在复杂系统中, 经常出现多个子系统之间存在相互耦合的问题, 其解算顺序可能导致不同的演化结果。基于狄拉克符号, 可以直观地解决多个子系统之间的相互解耦问题。按照狄拉克符号的描述方法, 物体的状态决定其产生场的形式。场即物体间相互作用的机制, 决定物体的状态如何改变。在仿真计算过程中, 存在相互耦合关系的子系统先根据状态计算相互作用场, 将这些场叠加之后, 再计算这些子系统的状态演化方式。

在系统工程早期阶段, 部分参量的变化规律尚不清楚, 可以认为其不发生变化。随着认识的深入, 可以通过在原有演化算符的基础上, 增加微扰算符来体现相关效应。

## 2.3 样机模型粒度变化的数学描述

随着系统工程过程的深入, 可以预期, 描述系统的态会包含越来越多的状态参量。对应到数学描述, 在系统工程早期, 系统  $A$  的态可以表示为

$$|A\rangle = \sum_i^m |A_i\rangle \quad (4)$$

随着工程学的开展, 状态参量的数量  $m$  会增加,  $|A\rangle$  会包含更多的状态参量。存在一种情况, 为了更详细地描述系统的某种设计, 针对  $|A\rangle$  中的某个  $|A_i\rangle$ , 需要增加  $|A_o\rangle, |A_p\rangle \dots (o, p > m)$  等一系列新的参量来描述  $|A_i\rangle$ 。在这种情况下, 需要引入一个特殊的预处理算符, 根据  $|A_o\rangle, |A_p\rangle \dots$  计算  $|A_i\rangle$ , 以保证其一致性。

假定系统工程的早期阶段, 用式 (4) 描述某个设计状态, 配套的过程演化算符是  $\hat{H}_{AA}$ 。随着系统研究的深入,  $|A\rangle$  复杂性提高,  $|A_i\rangle$  拓展增加了  $n$  个参量, 则  $|A\rangle$  可以表示为

$$|A\rangle = \sum_i^{m+n} |A_i\rangle \quad (5)$$

配套的算符是  $\hat{H}'_{AA}$ 。经过预处理后,  $|A\rangle$  仍然包含自洽的  $m$  个参量, 执行的计算过程仍然可以使用  $\hat{H}_{AA}$ 。

综上, 这种处理模式可以把不同阶段的设计结果, 按照早期粗粒度状态兼容的模式集成起来, 符合工程的客观规律。

## 2.4 典型集成验证问题的数学描述

飞行轨迹计算是典型的集成验证问题，通常被描述为一个变质量刚体组合后的刚体动力学问题。基于统一建模方法，全系统由  $n$  个变质量量子系统构成，第  $i$  个子系统的态可以定义为

$$|S_i\rangle = \begin{pmatrix} |r_i\rangle \\ |v_i\rangle \\ |p_i\rangle \\ |y_i\rangle \\ |r_i\rangle \\ |\omega_i\rangle \\ |m_i\rangle \\ |I_i\rangle \\ |F_i\rangle \\ |M_i\rangle \\ |t_i\rangle \end{pmatrix}, i=1,2,\dots,n \quad (6)$$

式中， $|r_i\rangle$  为该子系统质心位置矢量， $|v_i\rangle$  为该子系统速度矢量， $|p_i\rangle$  为该子系统的俯仰角， $|y_i\rangle$  为该子系统的偏航角， $|r_i\rangle$  为该子系统的滚转角， $|\omega_i\rangle$  为该子系统的角速度， $|m_i\rangle$  为该子系统的质量， $|I_i\rangle$  为该子系统的惯量， $|F_i\rangle$  为该子系统所受合力， $|M_i\rangle$  为该子系统的力矩， $|t_i\rangle$  为该子系统的仿真逻辑时间。尽管  $|S_i\rangle$  包含了全部的状态参量，由于狄拉克符号矩阵运算的特点，与算符矩阵不相关的状态参量并不需要参与运算，故在软件实现过程中，不会增加额外的计算量。此外，为了方便计算机程序实现，在追求态矢量完备性的同时，并不强制要求态矢量具备正交性，故可能存在对同一状态的重复描述。这种重复描述在计算机实现的过程中，需要进行针对性的预处理，以保证计算的正确性。

虽然任何子系统所处的时刻在物理上是一致的，但是对于一个分布式的仿真系统来说，其逻辑时间不一定一致。针对每一个子系统定义时间，是为了预留未来精细化处理的自由度。此外， $|t_i\rangle$  的存在，可以使每个子系统相对独立地进行时间演化计算，非常适用于处理分布式实时计算机系统过程中的延迟问题<sup>[14]</sup>。

式(6)未限定一些矢量的具体描述参量。例如位置矢量  $r_i$ ，没有规定其坐标系形式，计算过程中可以根据需要，按照坐标变换—计算—坐标变换的次序完成计算与恢复。

全系统，即  $n$  个刚体的组合体可以表示为

$$|S\rangle = \begin{pmatrix} |S_0\rangle \\ |S_1\rangle \\ \vdots \\ |S_i\rangle \\ \vdots \\ |S_n\rangle \end{pmatrix}, i=0,1,2,\dots,n \quad (7)$$

式中， $|S_0\rangle$  为该组合体整体的态，其存在是为了方便计算机完成状态计算。对于以  $S$  作为子系统的情形，可以直接使用  $|S_0\rangle$ 。对于最常见的纯刚性连接计算，大多数  $|S_i\rangle$  的分量等价于  $|S_0\rangle$  的分量。

对于  $|S\rangle$ ，其动力学计算过程可以用算符描述为

$$|S(t+\Delta)\rangle = \hat{D}_\Delta \hat{F} |S(t)\rangle \quad (8)$$

式中， $\hat{F}$  为根据状态计算受力(含力和力矩)的算符， $\hat{D}_\Delta$  为根据受力和上一时刻状态，计算时间间隔为  $\Delta$  的动力学状态演化的算符。

构成组合体的子系统之间也存在相互作用力，根据仿真粒度的需求，可以选择将这些相互作用力如式(9)表示

$$\hat{F} |S\rangle = \begin{bmatrix} \hat{F}_{00} & 0 & 0 & \cdots & 0 \\ 0 & \hat{F}_{11} & \hat{F}_{12} & \cdots & \hat{F}_{1n} \\ 0 & \hat{F}_{21} & \hat{F}_{22} & \cdots & \hat{F}_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \hat{F}_{n1} & \hat{F}_{n2} & \cdots & \hat{F}_{nm} \end{bmatrix} \begin{pmatrix} |S_0\rangle \\ |S_1\rangle \\ \vdots \\ |S_i\rangle \\ \vdots \\ |S_n\rangle \end{pmatrix} \quad (9)$$

式中， $\hat{F}_{ij}$  代表  $S_i$  和  $S_j$  连接产生的作用于  $S_i$  的力(含力和力矩)。不断迭代式(8)，即可实现时间推进计算。不存在相互作用的  $\hat{F}_{ij}$  为 0，可以进一步根据精度要求，忽略一些影响较小的  $\hat{F}_{ij}$ 。

基于狄拉克符号建立系统模型最大的优势，可以在不建立全系统代数方程的情况下，对系统进行完备描述。式(9)可以精准表示并选择计算或忽略  $\hat{F}_{ij}$ ，相应的计算过程均有严格的数学表达式。

## 3 基于类 TENA 中间件的软件实现方式

基于狄拉克符号的异构模型集成方法，需要类 TENA 中间件提供交互接口实现。类 TENA 中间件提供 3 种基本互操作实现方式，分别为对象发布订阅、消息传递和方法调用，三者具有不同的软件实现流程。

1) 对象发布订阅直接读取对象的态数值, 在仿真计算周期开始时获取, 只能获取最新的值。

2) 消息传递是一种基于队列机制的信息交互方法。仿真计算周期结束时, 已收到的消息按队列顺序处理。

3) 方法调用区别于对象交互、消息传递的处理, 可以触发指定对象执行一个动作, 以中断的方式修改实体的各种状态<sup>[15-16]</sup>。

在基于狄拉克符号的数学框架之下, 对象属性定义用于在计算机中描述系统的态, 消息传递用于实现不同分系统之间的控制、传感器信息的传递, 基于方法调用机制实现物体之间的相互作用算符, 对象发布订阅方法用于实现数据记录和

数据镜像等功能。物体的态、交互消息、相互作用算符都被封装为对象模型, 可以作为跨域异构中间件应用程序的插件, 以动态链接库的形式灵活分配。

基于狄拉克符号异构模型集成方法构建的集成试验验证系统, 是由大量应用程序组成的分布式应用系统。不同应用程序之间通过试验网络实现交互。如图 2 所示, 在试验环境中, 主应用程序用于调度整体的动力学计算流程和时间推进, 其他应用程序皆为从应用。从应用的作用包括两个方面, 第一是就近维护状态, 方便维护实物设备的状态与交互接口; 第二是提供高性能或冗余的方法调用服务, 用于复杂算法实现或试验系统容错。

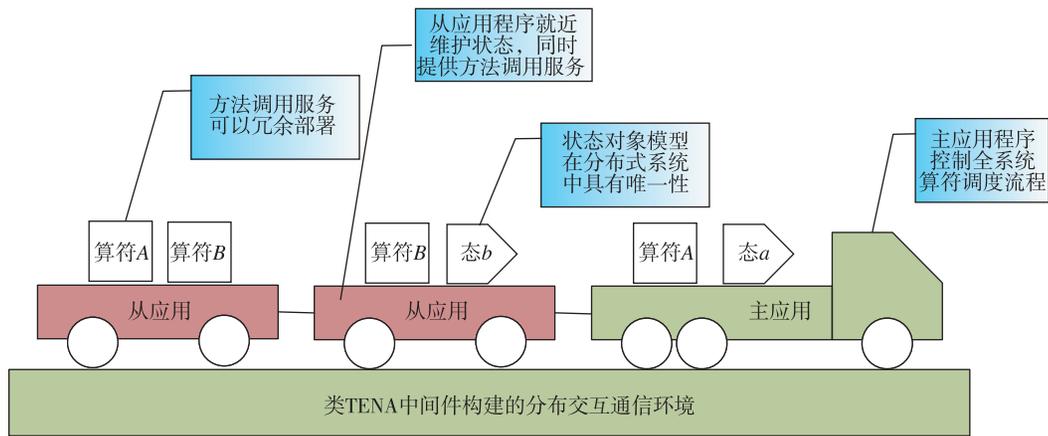


图 2 类 TENA 中间件运行模式

Fig. 2 Operating mode of the heterogeneous prototype integration middleware

主应用程序将样机对象模型组装成在虚拟试验场的数据空间中, 可以随逻辑时间演化的数字主线, 按照状态初始化、状态时间演化、状态更新循环步骤实现试验对象的多体动力学演化计算。

#### 4 仿真应用验证

以一个三级火箭的飞行轨迹计算为例, 进行仿真验证。该应用例包含发动机、载荷、控制等 5 个子系统, 外部环境模型包含重力和气动力。根据式 (6) 建立全箭和 5 个子系统的动力学状态对象模型。

建立重力算符  $\hat{G}$ , 其数学表示如下

$$\begin{pmatrix} |r(t)\rangle \\ |m(t)\rangle \\ |G(t)\rangle \end{pmatrix} = \hat{G} \begin{pmatrix} |r(t)\rangle \\ |m(t)\rangle \\ |G(t-\Delta)\rangle \end{pmatrix} \quad (10)$$

根据  $t$  时刻的位置矢量  $|r(t)\rangle$  和质量  $|m(t)\rangle$ , 计算  $t$  时刻的重力  $|G(t)\rangle$ , 使系统态中的重力部分  $|G\rangle$  从  $t-\Delta$  时刻推进到  $t$ 。

气动算符  $\hat{A}$  数学表示如下

$$\begin{pmatrix} |r(t)\rangle \\ |v(t)\rangle \\ |p(t)\rangle \\ |y(t)\rangle \\ |r(t)\rangle \\ |\tilde{S}(t)\rangle \\ |A(t)\rangle \\ |MA(t)\rangle \end{pmatrix} = \hat{A} \begin{pmatrix} |r(t)\rangle \\ |v(t)\rangle \\ |p(t)\rangle \\ |y(t)\rangle \\ |r(t)\rangle \\ |\tilde{S}(t)\rangle \\ |A(t-\Delta)\rangle \\ |MA(t-\Delta)\rangle \end{pmatrix} \quad (11)$$

式中,  $|\tilde{S}(t)\rangle$  是系统在  $t$  时刻的气动外形,  $|A(t)\rangle$  和  $|MA(t)\rangle$  为  $t$  时刻的气动力和力矩矢量。

质量求和算符  $\hat{R}$  数学表示如下

$$\begin{pmatrix} |m_0(t)\rangle \\ |I_0(t)\rangle \\ |m_1(t)\rangle \\ |I_1(t)\rangle \\ \vdots \\ |m_i(t)\rangle \\ |I_i(t)\rangle \\ \vdots \\ |m_n(t)\rangle \\ |I_n(t)\rangle \end{pmatrix} = \hat{R} \begin{pmatrix} |m_0(t-\Delta)\rangle \\ |I_0(t-\Delta)\rangle \\ |m_1(t)\rangle \\ |I_1(t)\rangle \\ \vdots \\ |m_i(t)\rangle \\ |I_i(t)\rangle \\ \vdots \\ |m_n(t)\rangle \\ |I_n(t)\rangle \end{pmatrix}, i=1,2,\dots,n \quad (12)$$

其计算功能是根据全部子系统  $S_i (i=1, 2, \dots, n)$  在  $t$  时刻的质量特性, 累加计算得到  $S_0$  在  $t$  时刻的质量特性。

建立一级发动机、二级发动机、三级发动机演化算符  $\hat{T}_I, \hat{T}_{II}, \hat{T}_{III}$ , 以  $\hat{T}_I$  为例, 其数学表示如下

$$\begin{pmatrix} |C(t-\Delta)\rangle \\ |m(t)\rangle \\ |I(t)\rangle \\ |T(t)\rangle \\ |MT(t)\rangle \end{pmatrix} = \hat{T}_I \begin{pmatrix} |C(t-\Delta)\rangle \\ |m(t-\Delta)\rangle \\ |I(t-\Delta)\rangle \\ |T(t-\Delta)\rangle \\ |MT(t-\Delta)\rangle \end{pmatrix} \quad (13)$$

式中,  $|C(t-\Delta)\rangle$  为上一时刻的发动机矢量推力控制指令。  $|T(t)\rangle, |MT(t)\rangle$  为  $t$  时刻的推力和力矩矢量。发动机演化算符的计算功能, 是根据控制指令计算  $t$  时刻的推力矢量和发动机剩余质量来实现的。发动机演化算符的实际计算与飞行时序有关, 是一个分段函数。

建立控制算符  $\hat{C}$ , 其数学表示如下

$$\begin{pmatrix} |C(t)\rangle \\ |S(t)\rangle \end{pmatrix} = \hat{C} \begin{pmatrix} |C(t-\Delta)\rangle \\ |S(t)\rangle \end{pmatrix} \quad (14)$$

计算机实现为根据飞行状态  $|S(t)\rangle$  计算控制信号  $|C(t)\rangle$ 。

全系统的计算顺序如下

$$|S(t+\Delta)\rangle = \hat{D}_\Delta \hat{F} \hat{A} \hat{G} \hat{R} \hat{T}_{III} \hat{T}_{II} \hat{T}_I \hat{C} |S(t)\rangle \quad (15)$$

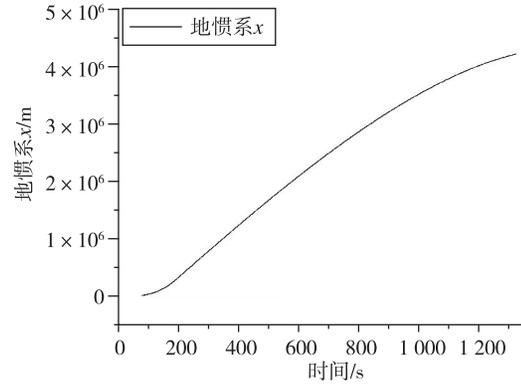
根据算符的计算法则, 靠近态的算符先计算。

按照表1参数建立模型, 根据式(15)进行仿真。所得飞行轨迹如图3所示。

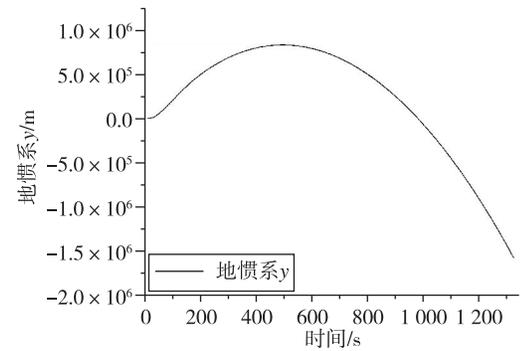
表1 仿真模型基本参数

Tab. 1 Basic parameters of simulation

参数	数值
一级初始质量/kg	22 680
一级燃料质量/kg	22 080
一级工作时间/s	61.6
一级推力/kN	912
一级长度/m	7.49
二级初始质量/kg	7 050
二级燃料质量/kg	6 246
二级工作时间/s	65.2
二级推力/kN	270
二级长度/m	4.11
三级初始质量/kg	5 770
三级燃料质量/kg	3 372
三级工作时间/s	59.6
三级推力/kN	157.5
三级长度/m	3.25
载荷质量/kg	2 020
载荷长度/m	3.41



(a) 地惯系位置矢量  $x$  轴分量



(b) 地惯系位置矢量  $y$  轴分量

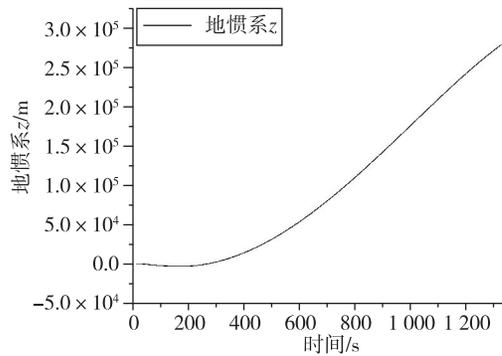
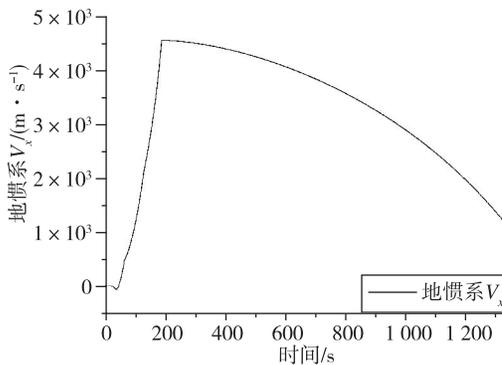
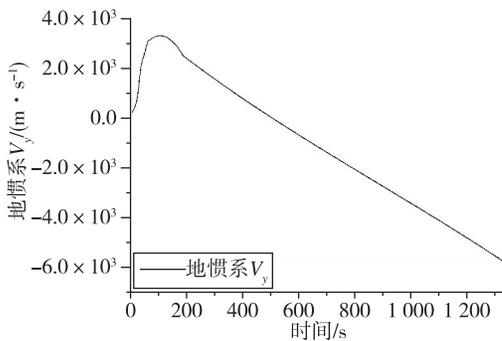
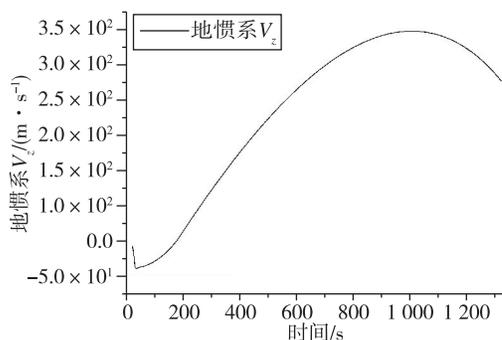
(c) 地惯系位置矢量  $z$  轴分量(d) 地惯系速度矢量  $x$  轴分量(e) 地惯系速度矢量  $y$  轴分量(f) 地惯系速度矢量  $z$  轴分量

图 3 飞行轨迹仿真结果

Fig. 3 Simulation results of flight trajectory

## 5 结束语

面向航天领域综合性能验证与评估需求,提出了一种以狄拉克符号作为基础数学架构的样机统一建模方法,具有系统易解耦、时间离散、屏蔽异构差异性、软件实现直观等特点。利用类 TE-NA 中间件提供的软件基础服务,构建基于该建模方法的数字系统模型,可以在虚拟空间中开展虚实融合系统集成试验,满足航天“簇”演进模式下,新技术快速插入、新系统原位替代和新状态快速验证等全生命周期虚实融合快速迭代要求。

## 参考文献

- [1] 张柏楠, 戚发轫, 邢涛, 等. 基于模型的载人航天器研制方法研究与实践[J]. 航空学报, 2020, 41(7): 023967.
- [2] Office of the deputy assistant secretary of defense for systems engineering. DoD digital engineering strategy [R]. Washington, D.C.: U.S. DoD, 2018.
- [3] 陈建伟, 杨春雷, 杨亮, 等. 美军数字工程最新进展及趋势分析研究[J]. 导弹与航天运载技术(中英文), 2023(1): 153-156.
- [4] 王巍巍, 王乐. 美国数字工程战略发展分析[J]. 航空动力, 2022(5): 23-26.
- [5] 任璐英, 张海峰, 王庆国, 等. 数字主线与数字孪生技术发展及在复杂产品研制中的应用[J]. 国防科技, 2023, 44(2): 75-81.
- [6] 王小军. 基于样机与模型的系统工程探索研究[J]. 导弹与航天运载技术(中英文), 2022(5): 1-6.
- [7] 赵民, 贾长伟, 张冶. 航天装备数字化建设研究与实践[J]. 宇航总体技术 2023, 7(2): 27-34.
- [8] Dirac P A M. 狄拉克量子力学原理[M]. 凌东波, 译. 北京: 机械工业出版社, 2018.
- [9] 董志明, 郭齐胜. 试验训练一体化仿真支撑技术研究[J]. 计算机仿真, 2021, 38(2): 1-3, 23.
- [10] 石小林, 王为. 数字空间站建设及其应用[J]. 航天器工程, 2022, 31(6): 76-85.
- [11] 赵建军, 丁建完, 周凡利, 等. Modelica 语言及其多领域统一建模与仿真机理[J]. 系统仿真学报, 2006, 18(S2): 570-573.
- [12] 廖建, 赵雯, 彭健, 等. 复杂产品虚拟试验支撑框架[J]. 计算机测量与控制, 2015, 23(4): 1249-1252.
- [13] 杜楠, 谭亚新, 冯斌. 基于对象元模型的 LVC 实验资源服务化方法研究[J]. 系统仿真学报, 2022, 34(8): 1834-1846.
- [14] 冯琦琦, 董志明, 贾长伟, 等. 面向 LVC 仿真的多层

- 分级时间管理方法研究[J]. 计算机仿真, 2020, 37(12): 1-4, 11.
- [15] 张丽晔, 廖建, 蔡斐华, 等. 试验训练使能体系结构(TENA)的研究与应用[J]. 计算机测量与控制, 2015, 23(10): 3461-3464.
- [16] 贾长伟, 莫怡华, 廖建, 等. 面向虚拟试验的可视化对象建模工具设计与实现[J]. 计算机测量与控制, 2014, 22(2): 4019-4021.

**引用格式:** 王晓路, 贾长伟, 张升升. 基于狄拉克符号的异构模型集成方法[J]. 宇航总体技术, 2024, 8(2): 6-14.

**Citation:** Wang X L, Jia C W, Zhang S S. Heterogeneous model integration method based on dirac notation [J]. Astronautical Systems Engineering Technology, 2024, 8(2): 6-14.