

体系仿真平台大样本仿真设计与验证

武小平, 曹 达, 董 帅, 罗汝斌

(北京宇航系统工程研究所, 北京 100074)

摘要: 大样本仿真推演对于提高仿真效率, 支撑体系研究、效能评估等具有重要作用, 针对大样本仿真推演面临的变量因子多、想定组合复杂、无法自动生成等关键问题, 对体系仿真平台大样本仿真设计的总体架构、系统组成进行研究, 对大样本仿真软件设计核心功能模块进行设计, 形成大样本仿真软件并进行仿真试验验证, 经试验验证该系统能够连续稳定运行, 批量生成仿真想定并完成仿真推演, 支持仿真倍速, 仿真总推进时长和并行仿真性能满足大样本仿真需求。

关键词: 体系仿真平台; 大样本; 仿真设计; 试验验证

Large Sample Simulation Design and Verification on System Simulation Platform

WU Xiaoping, CAO Da, DONG Shuai, LUO Rubin

(Beijing Institute of Astronautical Systems Engineering, Beijing 100074, China)

Abstract: Large sample simulations play an important role in improving simulation efficiency, supporting system research and efficiency evaluation. In view of the key problems faced by large-sample simulation and deduction, such as many variable factors, complex hypothetical combinations, and inability to automatically generate, this paper presents the overall architecture and system composition of the large-sample simulation design of the system simulation platform, designs the core functional modules of the large-sample simulation software design, achieves the large-sample simulation software, and verifies the simulation test, the results show that the system can run continuously and stably, generate the simulation assumptions in batches, and complete the simulation deduction, it supports the simulation speed and total simulation propulsion time and parallel simulation performance, and meeting the needs of large-sample simulations.

Keywords: system simulation platform; large sample simulations; simulation design; test verification

0 引言

目前体系仿真应用研究和效能评估分析等对仿真样本数量需求不断增长, 对仿真效率要求越来越高, 为此基于高性能计算的批量并行仿真技术是解决此类问题的重要研究方向, 也成为最近体系仿真平台应用发展的热点^[1]。大样本仿真是指包含大量仿真实体、参数因子等, 通过它们之间不同的交互和组合进行的仿真, 目前, 由于体系仿真场景越来越复杂, 仿真实体数量越来越多, 模型复杂度和模型间信息交互也成倍增加^[2-3], 并行仿真得到越来越多的重视, 往往需要在想定空间上进行大样本试验设计、自动想定生成和并行仿真推演^[4-5], 因此, 开展体系仿真平台大样本仿真设计与验证, 是解决体系仿真平台大样本仿真推演面临的变量因子多、想定组合复杂、无法自动生成等问题的关键, 对于满足体系大样本仿真应用研究需求具有重要作用。

国外比较典型的基于离散事件的仿真框架有 SPEED-ES、TWOS、IDES 等, 其中, 应用最广泛的支持大样本仿真的 SPEEDES 是由美国国家航空航天局组织开发的, 他将仿真同步并行能力和高性能计算相结合, 为计算多样本问题提供并行和分布的高性能仿真应用框架。当前美国国防部的一些主流仿真项目, 包括联合仿真系统 (JSIMS)、联合建模与仿真系统 (JMASS) 等, 都以 SPEEDES 作为基础仿真框架, 进行外围改造, 以支持仿真系统的多节点、大样本集成与运行。此外, SimKit 是一个基于支持设计、仿真、评估一体化的分布式仿真平台, 采用了先进的分布仿真运行技术, 支持仿真引擎的分布和协调, 具备对大样本进行分布式配置和运行控制的能力, 使得分布仿真过程可以按照需要动态调整, 从而达到根据机器的性能进行负载均衡的目的, 充分共享计算机的计算能力, 真正体现分布仿真的优势。

国内文献 [4] 针对目前采用 HLA 框架下并行仿真分

收稿日期: 2023-11-08; 修回日期: 2023-12-06。

作者简介: 武小平(1990-), 男, 硕士, 工程师。

引用格式: 武小平, 曹 达, 董 帅, 等. 体系仿真平台大样本仿真设计与验证[J]. 计算机测量与控制, 2024, 32(6): 307-312.

布式网络解决方案不足的问题,提出了基于并行处理的分析仿真解决方案。文献 [6-12] 对并行仿真引擎的开发和实现进行了大量的研究,其中文献 [12] 在分析研究 SPEEDES 仿真引擎的基础上,提出了基于高性能计算的并行仿真建模框架组成及功能,开发了并行离散事件仿真引擎,可支持共享内存并行仿真和集群及工作站互联。

国内仿真公司在借鉴国外典型体系仿真软件的同时,不断吸收与工业部门合作的成果,集成国外先进软件的设计思想,产品化了一批较为成熟的商业软件,如 XSimStudio、ISim 等仿真平台,对大样本仿真也有了一定的基础。国内高等院校也是引进国外先进联合仿真技术的另一支重要力量,在引入一些新的设计理念的同时,由于专业背景深厚,大大推动了国内体系仿真软件的技术革新。体系仿真平台 SoSim 是中国航天科技集团有限公司体系中心体系平台系列第一代产品。SoSim 平台面向体系的设计、推演、试验、鉴定、训练和评估业务,能够提供完整的仿真开发和应用环境、高效的仿真引擎和一系列支撑工具。大样本仿真是体系仿真平台的重要组成部分,主要有试验设计和样本分发等功能,能够根据仿真想定、仿真模型参数,完成想定随机样本数据的生成,并提供多个计算引擎节点并行执行多想定推演,突破了数量规模、环境模拟、运用场景等方面所限,形成可信的样本试验数据,为方案论证人员提供推演结果数据,支撑效能评估、体系仿真和体系研究等。

本文首先讨论了体系仿真平台大样本仿真的特点,指出当前面临的难点关键问题,然后从体系仿真平台发展趋势出发,分析并行和分布的高性能仿真应用框架,提出了基于 SoSim 体系对抗仿真平台大样本仿真设计总体架构和系统组成,并对大样本仿真软件核心功能模块进行设计,包括大样本试验设计和试验任务样本分发等;最后基于设计的大样本仿真软件进行仿真试验验证。

1 总体架构

大样本仿真的总体架构如图 1 所示,可分为基础环境层、基础资源层、基础服务层和业务应用层共 4 个层次。其中,基础环境为系统运行所要求的软件基础环境;基础资源层包括了系统中运行所需要的各类存储数据,如基础信息数据、试验想定数据、试验过程数据、地理信息数据、模型体系数据等,主要用于仿真模型、仿真服务的初始化和数据输入,累积仿真推演数据成果,作为大样本仿真结果的输出;基础服务层包含了应用所需的基础支撑服务,将各核心功能划分为较小且相对独立的功能点,每个功能点由一个独立的基础服务实现,如时间管理服务、事件管理服务、并行计算服务,样本分发服务、引擎监控服务等;业务应用层涵盖了试验任务管理、大样本仿真设计、试验任务样本分发和试验任务监控管理等功能,是业务功能的核心组成部分,在平台应用过程中,响应用户界面操作,完成基础服务输入输出的处理。



图 1 总体架构图

2 系统组成

大样本仿真主要包含试验任务管理模块、大样本试验设计模块、试验任务样本分发模块、试验任务监控管理模块 4 个功能模块。

1) 试验任务管理模块: 试验任务管理模块主要是对大样本仿真试验任务进行添加、查询和删除。该模块能够选择想定场景或选择样本模板并设置相关参数,最后生成试验任务完成大样本仿真试验任务创建。

2) 大样本试验设计模块: 试验设计在整个大样本仿真设计过程中,占有极其重要的地位。基于试验设计方法和试验因子参数设置,实现仿真试验大样本想定基础配置,实现对大样本仿真需要的基础想定进行设置,设定仿真试验参数,进而便捷、快速生成仿真样本。

3) 仿真试验任务样本分发模块: 仿真试验任务样本分发模块能够基于大样本试验任务进行并行仿真运行配置和分发,统一将样本数据装配打包,并将其分发到相应的仿真节点。样本分发分为手动分配和自动分配两种,手动分配即选择指定仿真节点,系统自动准备数据分发至指定节点,自动分配为系统按用户设置节点数量自动选择相应数量仿真资源节点进行数据分配。

4) 试验任务监控管理模块: 试验任务监控管理模块能够对大样本仿真试验的运行状态和仿真节点状态进行监控,包括当前运行已完成样本、未完成样本,当前大样本计算引擎状态、当前大样本运行结果统计信息,以及大样本仿真累计运行时间,预估剩余时间等。此外,能够对试验的进程进行控制,包括开始、暂停、继续和结束,在出现故障异常终止后,能重新加载仿真样本继续仿真。

3 系统软件设计

3.1 试验任务管理模块

试验任务管理模块主要是对大样本仿真试验任务进行添加、查询和删除。

1) 添加仿真试验任务: 用户新增仿真试验任务时选择想定场景后选择需要生成样本数据的模型实例信息, 然后设置模型实例相关初始化参数接口因子水平和样本生成算法, 系统根据用户设置生成试验任务样本并完成大样本仿真试验任务创建。初始化参数选择仅支持连续和离散两类类型参数, 样本生成算法有均匀设计、随机设计、正交设计、拉丁超立方设计。

2) 查询仿真试验任务: 用户可查询仿真任务列表并选择大样本仿真任务查看任务详情, 包括运行配置信息、样本信息以及任务运行记录信息列表, 并可在运行记录列表中查看该记录的仿真运行数据详细信息。

3) 删除仿真试验任务: 用户在仿真任务列表选择需要删除的任务, 系统会删除仿真任务、任务运行记录、任务样本数据以及运行记录仿真模型数据信息。试验任务管理流程如图 2 所示。

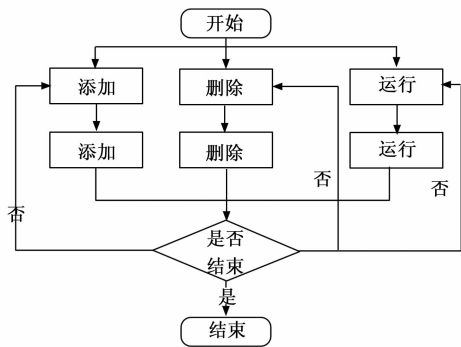


图 2 试验任务管理工作流程图

3.2 大样本试验设计模块

大样本试验设计是大样本仿真的核心, 正确的试验设计对高效地仿真运算具有重要作用。大样本试验设计是在进行试验之前, 根据研究目的, 对试验因子、研究内容和试验过程进行的规划和设计, 是一种同时研究多个输入因素对输出影响的方法^[13-15]。

3.2.1 试验因子体系构建

试验因子是大样本试验设计的基本单位。在导入基础想定后, 大样本试验设计从想定文件中选取要素, 形成试验因子。试验因子体系则是参与试验设计的所有试验因子的集合, 通过选取试验因子, 逐步构成试验因子体系。试验因子体系构建流程如图 3 所示。

3.2.2 试验因子水平设置

在进行大样本试验设计前, 需要对试验因子体系中的每个试验因子进行离散化处理, 形成若干个试验因子水平。在确定影响因子水平值时, 首先, 确定各个因子的类型以及取值可能的变化范围; 其次, 如果试验因子是离散型变量, 则其水平数就是其可能的离散点取值个数, 每个水平的值就是离散点的取值, 如果因子是连续型变量, 则第一步需要将连续的取值空间进行划分, 划分时考虑灵敏度、

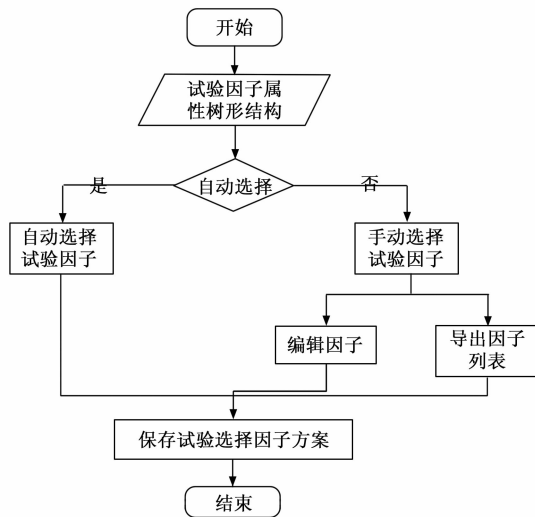


图 3 试验因子体系构建流程图

分辨率及仿真开销(代价)的原则。

试验因子水平设置可通过手动设计与自动设计实现。其中, 因子水平自动设计提供 3 种模式, 包括两端型模式、离散型模式和连续型模式; 手动设计则在仿真界面上进行手动的水平设置。试验因子水平设置模块的流程逻辑如图 4 所示。

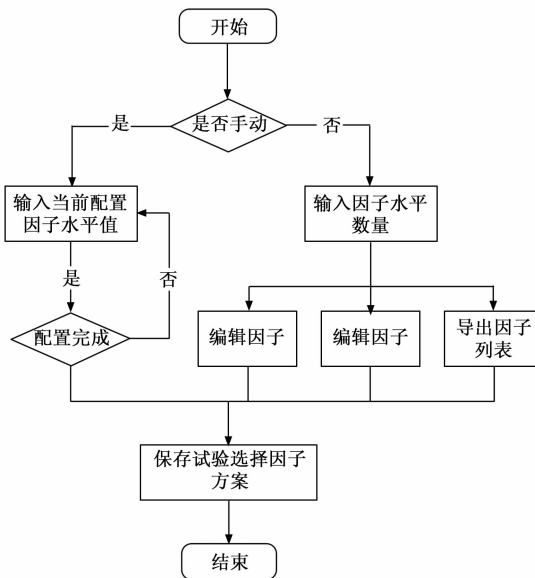


图 4 试验因子水平设置流程图

开始进行试验设计时, 允许用户选择自动生成试验因子水平或者手动生成。当用户选择自动生成方式, 需要用户输入当前因子的水平数量, 以及自动生成方式, 根据处理的因子数据类型进行计算生成水平序列; 如果用户使用手动配置方案, 则由用户依次输入各个水平数值。

3.2.3 试验设计方法

常见的试验设计方法有随机设计方法、正交设计方

法、均匀设计方法和拉丁超立方设计方法等。随机设计方法是最常见和容易实现的试验设计方法，它是将试验样本的对象进行随机分配，根据不同的因子条件形成多个不同的样本；正交设计方法是通过正交表实现的试验设计方法，它是通过多因子的优化对试验样本进行选取，从而使得样本具有一定的代表性^[16]，正交设计方法的优点是执行效率较高，能够通过较少的试验次数表征不同试验因子的组合，从而为寻找最优组合提供帮助；均匀设计方法是通过将试验因子进行均匀分配形成均匀分散的试验样本的试验设计方法，它是通过试验因子在样本空间内充分均匀地分散，来反映样本空间的特征，常用于多试验因子的快速试验设计，它的均匀性要比正交设计方法好，且便于发现试验规律，此外，它通过快速均匀设计比正交设计方法更加快速便捷，试验次数更少^[17-18]；拉丁超立方设计(LHD, Latin hypercube design)是最常用的试验设计方法之一^[19]，它是将每个因子构建一个随机矩阵，且使得矩阵的每一行、每一列都有不同的水平，它可以保持行和列的因子分布的均匀性，在设计时要求在同因子水平试验中因子间无关联性。

在通过试验因子水平设置对试验因子体系中的所有试验因子都实现离散化后，可调用试验设计模块内置的试验设计方法或进行人工设计，开展大样本试验设计，形成试验设计水平结果文件，试验设计水平结果文件中以行的形式记录每个样本中的参数，每个样本中按列逐个记录试验因子体系中各个试验因子的水平值^[20]。

试验设计方法调用模块以经过因子水平设计后的试验因子体系为输入，根据用户选择的试验设计方法，包括正交设计、均匀设计、拉丁超立方设计等，形成试验设计水平列表，列表中记录了样本数量，以及试验因子体系中每个因子在每个样本中的水平值。同时，试验设计模块支持手动设计，允许用户直接设置样本中的水平值，以及增加或删除样本等。试验设计方法模块的流程如图 5 所示。

3.2.4 试验样本空间生成

当完成试验因子体系构建和试验因子水平设置两部分内容后，对试验因子体系进行试验设计，选择相应的试验设计方法后，模块调用相应的试验设计方法，生成试验设计方案，并将试验设计方案交给试验样本空间生成模块，试验样本空间生成模块将试验因子水平数据和想定内容作为输入，通过不同类型的数据，按照试验因子水平，进行相应计算映射，得到试验样本空间数据。

试验样本空间生成模块包括试验参数分发功能和样本空间文件生成功能。其中，试验参数分发功能包括实体属性的映射和环境要素的映射。映射完成后，可生成试验样本空间文件。试验样本空间生成功能包括样本数设置、样本空间信息输入、样本空间文件导出和存储等功能。试验样本空间文件生成完毕后，试验设计软件将该文件存储到

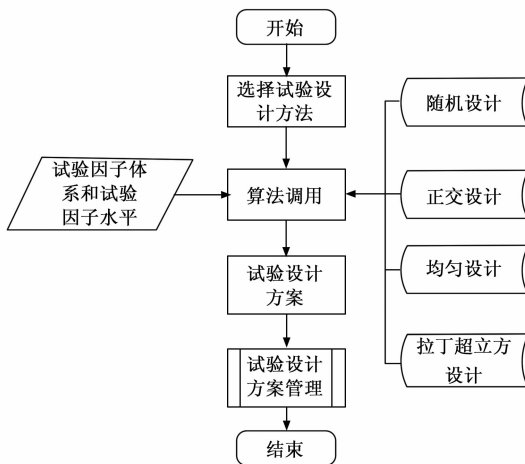


图 5 试验设计方法流程图

数据库中。试验样本空间生成流程如图 6 所示。

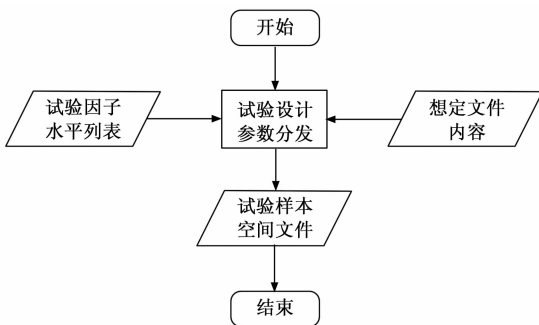


图 6 试验样本空间生成流程图

试验设计功能所形成的试验设计水平结果文件中记录的是试验因子的水平值，而非试验因子所表示的想定要素的实际取值。试验设计参数分发功能就是将试验设计水平值映射到因子所对应的要素的实际取值^[21-22]。经过试验参数分发后，形成试验样本空间文件，该文件记录了试验设计结果中每个样本的信息，主要为样本中所有试验因子对应的想定要素的实际取值。试验样本文件以单独文件的形式记录多个样本中的想定要素参数值，有效地避免了一个样本对应一个想定文件所带来的文件冗余的问题。

3.3 试验任务样本分发模块

大样本仿真样本分发模块能够基于并行仿真试验任务和并行仿真运行配置方案，将方案中所设置的样本数据（主要包括模型和模型初始参数）装配打包，并将其分发到相应的仿真节点。

试验任务样本分发模块基于并行仿真运行配置模块样本分发配置信息形成分发方案，大样本仿真启动时，本模块将节点及模型资源映射关系、模型文件的路径、模型的初始化数据发送给仿真控制引擎模块，仿真控制引擎模块再将模型文件路径、模型的初始化数据下发给仿真计算引

擎, 计算引擎根据模型文件路径将下载模型资源, 然后按照控制指令驱动模型。大样本仿真试验任务样本分发分为手动分配和自动分配两种, 手动分配及选定指定仿真节点系统自动准备数据到指定节点, 自动分配为系统按用户设置节点数量自动选择相应数量仿真资源节点进行数据分配, 节点忙碌或者节点数量不足直接返回用户提示信息。试验任务样本分发主要流程如图 7 所示。

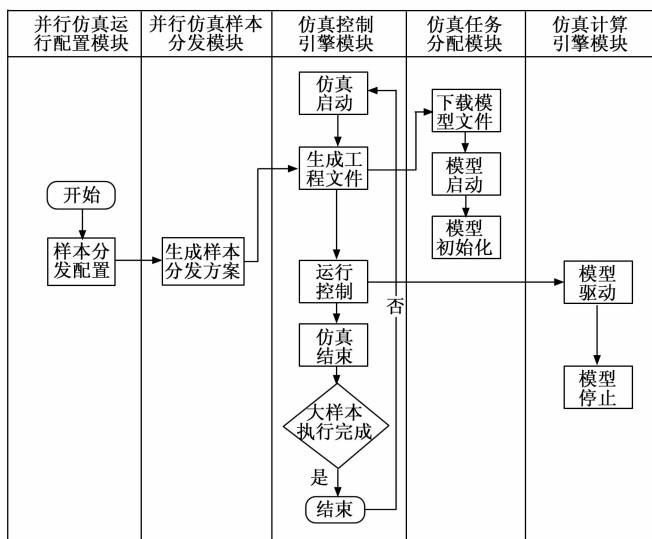


图 7 试验任务样本分发流程图

3.4 试验任务监控管理模块

针对大样本试验的特点, 试验任务监控管理模块能够实现试验运行前检查与闭环试验的多节点大样本仿真调度, 可实现对多个大样本计算引擎并行开展仿真试验, 实现多个计算节点中仿真试验任务的运行和控制, 支持仿真试验任务的新建、编辑、删除等, 具备对仿真任务的管理, 能够对多个计算节点进行负载均衡分配; 能够实现仿真试验任务运行的控制, 包括启动、暂停、继续和停止等操作, 满足大样本仿真断点续跑等功能。此外, 能够对仿真执行任务进行多次有序地调用, 对每台机器的试验状态和计算节点状态进行实时监控。

3.4.1 系统运行监控

能够对仿真试验的运行状态和仿真节点状态进行监控, 能够展示整个大样本仿真试验系统状态, 包括当前运行试验、当前仿真组(大样本计算引擎)状态、系统累计运行时间等^[23-24]。

3.4.2 试验运行监控

能够对大样本任务下计算引擎下发控制指令, 能够查询大样本仿真试验任务状态, 获取完成大样本仿真试验运行配置和任务下发的任务, 并检查其是否满足仿真运行要求, 并展示给用户该任务当前的运行试验名称和对应的样本名称, 用户确认无误后能够点击开始该试验的运行, 并将该纳入试验运行监控。用户能够对试验的进程进行控

制, 包括开始、暂停、继续和结束。能够实现试验运行后, 出现故障异常终止后, 重新加载仿真想定, 从指定衍生想定开始连续仿真。试验任务监控管理主要流程如图 8 所示。

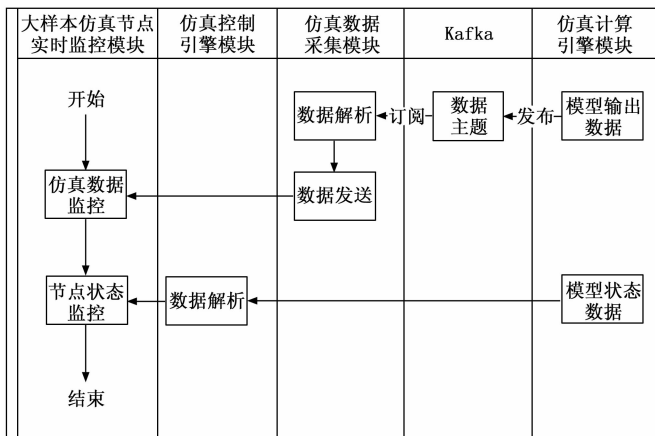


图 8 试验任务监控管理处理流程

4 大样本仿真试验验证

大样本仿真软件基于想定编辑集成工具的输出成果, 提供仿真试验设计、分布式批量仿真管理、试验运行控制等功能, 实现大样本仿真试验的批量生成与组织运行^[25]。最终构建的大样本仿真软件界面如图 9 所示。



图 9 大样本仿真软件界面

为了对大样本仿真设计有效性检验, 验证大样本仿真性能, 通过大样本试验设计模块生成 5 940 个想定 xml 文件, 分成为 30 批次, 调用 Sosisim2.0 平台导入想定的接口, 每一批次导入 200 个想定文件, 进行大样本仿真测试。大样本仿真测试信息如表 1 所示。

大样本仿真共配置有 18 个节点, 包含 12 个仿真节点, 6 个服务节点, 其中 17 个为 Windows 操作系统节点, 1 个为 Linux 操作系统节点。大样本仿真验证如下: 基于 377 个想定实体, 采用随机设计方法对 19 个试验因子进行设计, 生成 5 940 个随机任务数量的基础想定样本, 并在 12 个仿真节点并行仿真运行。调用多方案运控接口, 将同一批次的想定文件作为一条多方案运控任务, 仿真结果能够顺利完

表 1 大样本仿真测试信息表

| 项目 | 类别 | 个数 | 说明 |
|------|-------------------------|-------|-------------|
| 配置信息 | 样本数量 | 5 940 | 基础想定样本 |
| | 仿真节点 | 12 | 计算引擎、控制引擎 |
| | 服务节点 | 6 | Java 服务 |
| | 操作系统 Windows server2019 | 17 | 34.56 万亿次算力 |
| | 操作系统 Linux CentOS 7.9 | 1 | 38.01 万亿次算力 |
| 样本信息 | 实体个数 | 377 | 模型实例 |
| | 试验因子 | 19 | 试验变量 |
| | 试验设计方法 | 1 | 随机设计方法 |
| | 任务数量 | 200 | 试验设计方法随机生成 |

成 30 批次 200 个样本并行仿真推演, 单次仿真总推进时长为 4 000 s, 倍速为 50。通过上述仿真测试结果表明大样本仿真设计满足设计功能要求, 可支持创建大样本衍生想定任务, 基于试验设计方法和因子实现大样本仿真想定的生成, 协同注册至仿真平台的各个分布式计算节点进行仿真推演, 并对试验任务进行管理。目前能够实现 18 个节点连续稳定运行, 单日生成样本想定数据不少于 500 条, 解决了目前仿真平台大样本仿真自动批量仿真难、试验样本数量不足的问题。

5 结束语

大样本仿真作为体系仿真平台的重要组成, 解决了大样本仿真推演面临的变量因子多、想定组合复杂、无法自动生成等关键问题。本文提出了基于 SoSim 体系对抗仿真平台大样本仿真设计的总体架构、系统组成, 对大样本仿真设计的软件核心内容进行了重点介绍, 最终构建了大样本仿真软件并进行仿真测试验证。验证结果表明大样本仿真设计能够满足设计功能要求, 为体系效能和适用性评估、体系研究提供了有力支撑。

参考文献:

- [1] 陈慧龙. 军用并行仿真想定编辑与生成技术研究及实现 [D]. 长沙: 国防科学技术大学, 2011.
- [2] PERUMALLA K S. Switching to high gear: opportunities for grand-scale real-time parallel simulations [C] // IEEE, USA, 2009: 3-10.
- [3] FUJIMOTO R. Parallel and distributed simulation systems [M]. USA: John Wiley & Sons, Inc., 2000: 50-53.
- [4] 姚益平, 张颖星. 基于并行处理的分析仿真解决方案 [J]. 系统仿真学报, 2008, 20 (24): 6617-6621.
- [5] 姚益平. 面向大规模体系仿真的高性能仿真计算机系统 [J]. 系统仿真学报, 2011, 23 (8): 1617-1623.
- [6] 阎晋屯, 康凤举, 苏颖. 基于 HLA 的水下对抗仿真想定生成与管理 [J]. 系统仿真学报, 2005, 17 (5): 1203-1207.
- [7] 石峰, 赵雯, 王维平. 联合作战仿真应用中的想定系统框架 [J]. 系统仿真学报, 2003, 15 (2): 212-215.
- [8] 李响. 基于 MGIS 的军事想定辅助生成系统的设计与实现 [D]. 郑州: 中国人民解放军信息工程大学, 2007.
- [9] 苏年乐, 李群, 王维平. 组件化仿真模型交互模式的并行化改造 [J]. 系统工程与电子技术, 2010, 31 (9): 2015-2020.
- [10] 鄢超波, 赖华贵, 赵千川. 多智能体并行仿真框架 [J]. 系统仿真学报, 2010 (s1): 191-195.
- [11] 乔海泉. 并行仿真引擎及其相关技术研究 [D]. 长沙: 国防科学技术大学, 2006.
- [12] 刘奥, 姚益平. 基于高性能计算环境的并行仿真建模框架 [J]. 系统仿真学报, 2006, 18 (7): 2049-2051.
- [13] MORRIS M D, MITCHELL T J. Exploratory designs for computational experiments [J]. Journal of Statistical Planning and Inference, 1995, 43 (3): 381-402.
- [14] YE K Q, LI W, SUDJANTO A. Algorithmic construction of optimal symmetric Latin hypercube designs [J]. Journal of Statistical Planning and Inference, 2000, 90 (2): 145-159.
- [15] FANG K T, MA C X, BLINKER P. Centered L2 discrepancy of random sampling and Latin hypercube design, and construction of uniform designs [J]. Mathematics of Computation, 2002, 75 (71): 275-296.
- [16] 徐仲安. 正交试验设计法简介 [J]. 科技情报开发与经济, 2002, 12 (5): 148-150.
- [17] 方开泰. 均匀试验设计的理论、方法和应用历史回顾 [J]. 数理统计与管理, 2004, 23 (3): 69-80.
- [18] FANG K T. The uniform design: application of number-theoretic methods in experimental design [J]. Acta Math. Application Sinica, 1980, 3: 363-372.
- [19] MCKAY M D, BECKMAN R J, CONOVER W J. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code [J]. Technometrics, 1979, 21: 239-245.
- [20] MISRA J. Distributed discrete event simulation [J]. ACM Computer, 1986, 43 (18): 39-65.
- [21] FUJIMOTO R M. Parallel discrete event simulation [J]. Communication of the ACM (S0001-0782), 1990, 33 (10): 54-62.
- [22] FUJIMOTO R M. Parallel and distributed simulation systems [M]. New York, USA: John Wiley & Sons, Inc, 2000.
- [23] STEINMAN J S, PARK J. A proposed open system architecture for modeling and simulation [C] // Proceedings of the Fall 2007 Simulation Interoperability Workshop. Orlando, USA, 2007: 13-21.
- [24] STEINMAN J S. The distributed simulation management services layer in SPEEDES [C] // Proceedings of the Spring 2004 Simulation Interoperability Workshop. Arlington, USA, 2004: 55-58.
- [25] HARDY D R, ALLEN E C, ADAMS K P, et al. Advanced distributed simulation: decade in review and future challenges [C] // Space and Naval Warfare Systems Center. San Diego, USA, 2001: 165-175.