

建模与仿真服务化研究综述

张淼¹ 许凯¹ 彭勇¹ 尹全军¹

摘要 建模与仿真服务化是提升用户体验,支撑按需访问建模与仿真能力的有效手段.本文首先从建模与仿真服务的访问、开发以及运行与管理三个层面对建模与仿真服务化的概念进行辨析;并从服务的分类、抽象层级、基本元素和状态四个角度对建模与仿真服务的特征进行阐述.然后从基于网页的仿真、基于面向服务架构(Service oriented architecture, SOA)的仿真系统开发和服务化基础设施三个维度对建模与仿真服务化的发展历程进行梳理.在此基础上,分析了基于云的建模与仿真服务化的构建原则、基本架构和应用模式,并从访问、开发以及运行与管理三个层面给出建模与仿真服务化相关的支撑技术.最后,从理论体系、关键技术和新兴技术三个方面给出进一步发展建模与仿真服务化的建议.

关键词 仿真服务, 建模与仿真, 云仿真, 面向服务架构, 建模仿真即服务

引用格式 张淼, 许凯, 彭勇, 尹全军. 建模与仿真服务化研究综述. 自动化学报, 2023, 49(8): 1601–1620

DOI 10.16383/j.aas.c220555

Overview of Servitization of Modeling and Simulation

ZHANG Miao¹ XU Kai¹ PENG Yong¹ YIN Quan-Jun¹

Abstract Cloud-based modeling and simulation (M&S) servitization can effectively improve user experience and the efficiency in developing simulations. In this paper, we firstly illustrate the concept of M&S servitization from three aspects, namely application, development, operation and management of M&S services. The connotation of modeling and simulation service is explained from four aspects, namely the categories, abstraction levels, constituent elements and states of services. Then, the development and evolution of modeling and simulation servitization is summarized from three aspects, namely web-based simulation, SOA (service oriented architecture)-based simulation development as well as operation and management of simulation on service-oriented infrastructure. On this basis, the construction principle, basic architecture and application mode of cloud-based modeling and simulation servitization are analyzed, and the supporting technologies related to modeling and simulation servitization are given from the views of application, development and operation and management. Finally, suggestions for further development of M&S servitization are given from three aspects: Theoretical systems, key technologies and emerging technologies.

Key words Simulation service, modeling and simulation, cloud simulation, service oriented architecture (SOA), modeling and simulation as a service (MSaaS)

Citation Zhang Miao, Xu Kai, Peng Yong, Yin Quan-Jun. Overview of servitization of modeling and simulation. *Acta Automatica Sinica*, 2023, 49(8): 1601–1620

仿真技术伴随计算机技术的诞生而产生,至今已经发展了六十多年,并在国防经济生活中扮演着越发重要的角色.美国国会在2007年通过的487号决议中,明确将建模与仿真列为“国家核心技术”.2018年12月修订生效的美国新版高等教育法中,建模与仿真作为唯一一项具体技术被列入其中,并使用大量篇幅说明如何推动建模与仿真技术在大学教育中的普及^[1].

然而,随着建模与仿真技术应用的不断深入,也面临着很多新的问题^[2–4]: 1) 仿真系统的规模日益扩大,模型的粒度与功能也越发精细与复杂,需要更多的计算资源; 2) 用户购买、操作和维护高性能计算机需要较大成本,容易造成浪费; 3) 不同单位或机构开发的海量异构仿真资源难以高效共享与重用,降低系统开发效率; 4) 仿真需求动态变化时,开发人员难以快速构建所需仿真应用; 5) 仿真的领域性、专业性不断增加,用户难以简单方便地使用和访问.

云计算和面向服务架构(Service oriented architecture, SOA)技术的发展为上述问题的解决提供了一种新的思路^[5].借助于虚拟化技术和“X即服务”的计算模型,云计算能够为仿真用户提供近乎无限的计算资源和随时随地的便捷访问.同时,能够避免硬件上的巨大投入,降低仿真的应用门槛.

收稿日期 2022-07-06 录用日期 2022-12-09
Manuscript received July 6, 2022; accepted December 9, 2022
国家自然科学基金(62103420)资助
Supported by National Natural Science Foundation of China (62103420)
本文责任编辑 陈谋
Recommended by Associate Editor CHEN Mou
1. 国防科技大学系统工程学院 长沙 410073
1. College of Systems Engineering, National University of Defense Technology, Changsha 410073

基于 SOA 的仿真服务开发使得服务的实现与使用相分离,且服务自包含、松耦合、可重用的特性有助于仿真资源的高效共享与重用,进而提高开发效率,降低开发成本。

自 Gustavsson 等^[6]在 2004 年的秋季仿真互操作会议上首次提出面向服务仿真的概念以来,建模与仿真服务化便受到大量学者和从业人员的广泛关注。李伯虎等^[2]在网格仿真工作的基础上提出云仿真的概念^[7]。Fujimoto 等^[3]则强调云计算、边缘计算对于解决仿真所面临瓶颈问题的巨大潜力。民用领域,建模与仿真服务化被应用在教育^[8]、交通^[9]、能源^[10]等多个行业并产生显著经济效益。特别地,建模与仿真服务化能够为国防及工业部门的仿真能力建设带来巨大推动作用,并受到世界军事强国或组织的广泛关注。

为改善信息、资源、能力的共享,促进互操作与联合作战,美国国防部在 2007 年以 SOA 为支撑架构提出了网络中心服务战略,全球信息网格(Global information grid, GIG)则是该战略转型的重要标志^[11]。2012 年,美军提出了云使能模块化服务框架(Cloud-enabled modular services, CEMS)^[12]以实现高效、模块化、可组合的仿真训练环境的快速构建。北约科技组织下的建模与仿真研究组(NATO modeling and simulation group, NMSG)则组建了多个专业的研究小组对建模仿真即服务(Modeling and simulation as a service, MSaaS)的概念、架构、技术、实现、应用等方面进行研究。其中 MSG-136 小组活动时间为 2014 年 9 月至 2017 年 12 月,并于 2017 年底发布了 MSaaS 参考架构^[13]; MSG-164

小组活动时间为 2018 年 2 月至 2021 年 2 月,负责对 MSaaS 功能进行完善,实现并验证原型系统^[14]; MSG-195 小组的目标则是使 MSaaS 架构成熟可用,预期活动时间为 2021 年至 2024 年。同期,美国国防部建模仿真协调办公室在 2020 年 2 月推出了国防建模与仿真参考架构来促进国防部建模与仿真工作^[15],同时指出,“对于很多建模与仿真应用而言,向云计算和 SOA 的迁移可以改善可访问性和敏捷性,同时降低运营和其他生命周期成本。”

考虑到建模与仿真服务化对国防和经济建设的巨大推动作用,很多学者基于特定的应用需求对其进行深入研究,并取得丰硕成果。然而,现有工作大多只针对建模与仿真服务化的某一侧面,如参考架构、实现方式、关键支撑技术等进行研究,没有形成对建模与仿真服务化完整的研究体系。另一方面,从仿真系统全生命周期考虑,一般涉及仿真用户、系统开发人员和系统运维人员。每类人员对建模与仿真服务化具有不同的需求,但现有研究缺乏对这些多样化需求的全面考量。尽管也存在一些文献[16–20]尝试对建模与仿真服务化的研究进行梳理,但通常是针对特定问题场景或只从某个侧面对建模与仿真服务化进行阐述,研究的体系性仍存在欠缺。在表 1 中我们详细给出了本文研究与现有相关综述的异同点。

本文首先对建模与仿真服务化的背景知识进行介绍,给出其基本概念和主要特征;然后从访问、开发以及运行与管理三个角度对基于云的建模与仿真服务化的研究工作进行全面的梳理,给出其演化逻辑;之后从构建原则、基本架构和应用模式三个层

表 1 与现有相关综述的异同

Table 1 Differences and similarities with existing reviews

文献名称	发表时间	主要内容	与本文异同点
Modeling and simulation as a cloud service: A survey ^[16]	2013	首次给出 MSaaS 的服务模型,探讨 MSaaS 架构和部署模式,分析 MSaaS 可能面临的安全威胁,简要介绍服务组合技术	1) 本文同样对 MSaaS 的服务模型、架构、部署模式进行探讨; 2) 该文缺乏对 MSaaS 发展历程的梳理,且对实现 MSaaS 所需的关键技术讨论较少
面向服务的建模与仿真技术综述 ^[17]	2013	对 SOA 与 HLA、DEVS、MDA 和云计算等规范或技术的结合进行研究,探讨基于 SOA 实现建模与仿真的服务化	1) 本文同样对 SOA 在仿真领域的应用进行深入探讨; 2) 受限于当时的技术发展,该文缺乏对微服务、纳米服务等 SOA 新形态的介绍
Architectural design space for modeling and simulation as a service: A review ^[18]	2020	针对 MSaaS 的架构设计空间进行深入研究,给出 MSaaS 架构的分类标准,指出 MSaaS 架构需要具备的核心能力	1) 本文同样对 MSaaS 的架构进行探讨,并给出通用架构; 2) 该文只综述了与 MSaaS 架构相关的工作,缺乏对 MSaaS 的相关概念、发展历程、支撑技术的讨论
Towards cloud-native simulations — Lessons learned from the front-line of cloud computing ^[19]	2021	阐述云计算范式的发展对仿真领域的影响,提出云原生仿真参考模型,分析微服务、纳米服务对云原生仿真的影响	1) 本文同样讨论了单体服务、微服务、纳米服务等技术对云原生仿真的影响; 2) 该文缺乏对实现建模与仿真服务化所需技术的整体性阐述
网络化仿真及其发展趋势 ^[20]	2021	给出网络化仿真的含义与特征,阐述网络化仿真发展历程,探讨未来网络化仿真发展趋势	1) 本文同样对建模与仿真服务化的发展历程和未来趋势进行深入讨论; 2) 该文对建模与仿真服务化发展历程的介绍不够全面,也没有给出关键的支撑技术

面阐述建模与仿真服务化的内涵;在此基础上,从访问、开发以及运行与管理的角度给出建模与仿真服务化的关键支撑技术,为仿真用户、开发人员、运维人员等不同类型人员提供参考;最后,对如何进一步提升建模与仿真服务化水平给出一些建议。

1 建模与仿真服务化背景知识

本节首先介绍建模与仿真服务化相关的概念,给出服务的定义,对服务与组件的关系进行辨析,并从三个层面阐述建模与仿真服务化的含义。然后对建模与仿真服务的特征进行分析,探讨了仿真服务的分类、抽象层级、组成元素和状态。

1.1 建模与仿真服务化基本概念

1) 服务定义。服务并不是一个新的概念,很多学者和组织都给出了服务的定义。结构化信息标准推进组织将服务定义为^[21]“按照服务描述约定的接口、约束和策略访问一种或多种能力的机制”。开放团体(The open group)认为服务是“一种自包含的、具有特定输出的可重复黑盒逻辑表示”^[22]。也有学者从功能的角度考虑,认为“服务”这一概念,体现了常用软件组件的松耦合、可重用、可组合与可发现^[23-24]。

总的来说,服务是一种可通过标准化的接口进行访问的、具有特定能力的逻辑表示,具有独立、自包含、松耦合的特征,可通过服务描述实现服务的发现、组合、集成与重用。

2) 服务与组件概念辨析。服务与组件是两个容易令人混淆的概念,它们之间具有很多的相似之处,如都可通过组合与集成实现系统的快速开发与资源重用。但它们之间也存在一些显著的差异。

对象管理组织的“建模语言规范”中,将组件定义为“系统中一种物理的、可代替的部件,封装并实现了一系列可用接口”。计算机百科全书的描述中,组件是“系统中与语境有明显依赖关系的、独立、可组装软件实体”。Szyperski^[25]认为,组件“是一个彼此独立,且只基于定义好的方式进行通信的部署单元,能够通过重部署支持动态场景”。文献[26]则认为,组件是仿真环境的结构分解,服务则是仿真环境的功能分解,一个服务所执行的功能可通过组件的方式实现。

综合上述对于组件的定义,我们认为组件与服务主要在三个方面存在差异:a)上下文依赖性。服务不依赖于上下文,可进行独立的更新与替换,组件则与上下文环境高度相关。b)平台依赖性。服务是对功能更高级的抽象,可使用任意编程语言实现,而典型的组件技术则与具体的编程语言或平台相

关(如DCOM只支持Windows)。c)可发现能力。服务是服务描述和多个服务实现的组合^[24],可基于标准化的服务描述实现服务的发现。组件更多强调系统在结构层面的拆分,难以实现组件的快速发现。

3) 建模与仿真服务化。建模与仿真服务化是指以服务为中心构建模型与仿真系统,其具有三个层面的含义:a)面向服务的访问。主要强调用户可以基于浏览器或客户端在任何时间、任何地点方便快捷地访问所需的仿真服务,允许用户基于标准化的仿真服务描述实现服务的发现与组合,支持用户实现快速的定制化仿真。b)面向服务的开发。基于面向服务架构(SOA)的软件思想开发仿真模型和仿真支撑环境,通过对通用功能组件的服务化封装以及标准化的服务描述,实现仿真服务的自包含、松耦合、可发现与可组合。c)面向服务的运行与管理。建模与仿真服务往往以分布的方式在云数据中心等远端部署和执行,这给仿真的运行与管理带来了许多挑战。具体而言,面向服务的运行与管理可以从三个角度去考虑,包括仿真应用的正确运行、仿真资源的高效管理、仿真过程的安全可靠。

1.2 建模与仿真服务特征

1) 服务的分类。对于建模与仿真服务的表现形式,存在多种不同的理解。从应用模式的角度看,Cayirci等^[16]借鉴云计算服务的思想,将其分为建模即服务、模型即服务和仿真即服务。Liu等^[27]则将服务模式扩充为四种,增加了VV&A(Verification, validation & accreditation)即服务。

从仿真资源的角度看,任何资源都可被封装为服务。根据美军建模与仿真资源库(Modeling and simulation resource repository, MSRR)对仿真资源的定义,与仿真相关的数据、元数据、算法、模型、仿真应用和工具集等不同类型的资源均属于仿真资源^[28]。事实上,利用虚拟化等手段,实装、模拟器等资源也可被封装为服务。

同时,服务可能拥有不同层级的分辨率^[29],可能是软件组件、已部署的应用,或是整个的系统。

2) 服务的抽象层级。根据服务与仿真架构、仿真平台及编程语言的关系,可将服务划分为四种抽象层级^[24]:a)实现无关层级,是对服务最高级的抽象。服务包含与仿真架构和实现相独立的服务描述,可用于设计时的服务发现和组合。在该层级,本体语言(如基本对象模型)可用于描述仿真服务的需求规范,以辅助后续的软件开发。b)仿真架构相关层级,服务包含架构依赖的服务描述。基于本体的服务描述可以被细化为仿真架构相关的描述,如高

级体系架构 (High level architecture, HLA)、分布式交互仿真 (Distributed interactive simulation, DIS)、试验与训练使能体系结构 (Test and training enableing architecture, TENA), 或其他新型仿真架构. c) 服务平台相关层级. 服务描述可被细化为与平台相关的格式, 如网页服务描述语言 (Web services description language, WSDL) 及其变种等. d) 实现相关层级. 在该层级, 服务基于特定编程语言或代码框架实现.

3) 服务的基本元素. 基于北约对服务的描述, 服务的基本元素包括属性、策略、功能以及接口^[24]. 服务具有服务属性 (如可用性), 受服务策略约束 (如服务必须在 X 天内提供), 执行服务功能, 并提供访问接口.

4) 服务的状态. 服务可以是有状态或无状态的. 无状态服务是无记忆的, 不同的服务调用间彼此独立, 输出只与输入有关, 如路径规划服务. 有状态服务则是有记忆的, 输出不仅与输入有关, 也与服务实例当前的状态有关, 如模型服务. 在仿真应用中, 很多服务可能是有状态的^[30-31], 这给服务调度、时间同步、数据管理等都引入了新的挑战.

2 建模与仿真服务化的发展与演化

建模与仿真服务化具有丰富的内涵. 本节分别从基于网页的仿真、基于 SOA 的仿真系统开发和服务化计算基础设施三个角度对建模与仿真服务化在访问、开发以及运行与管理三个层面的发展脉络进行梳理.

2.1 基于网页的仿真

随着 90 年代初 WWW 的出现与发展, 考虑到应用网页化的趋势, 冬仿会议在 1996 年第一次开始了关于基于网页仿真 (Web based simulation, WBS) 的讨论^[32]. Wang 和 Wainer^[33-34] 认为“在 WBS 中, 仿真器及仿真环境位于远端服务器, 用户

通过浏览器和网页提供的资源与技术提交请求并获取所需的仿真资源.”

WBS 本质是 Web 技术与分布式仿真技术的融合^[35-36], 浏览器在建模与仿真的过程中扮演着重要的角色, 或者只作为一个图形化接口, 或者作为仿真引擎的容器^[37]. 理论上, 用户可在任何支持浏览器的设备上获取所需的建模与仿真服务.

相比于传统的仿真模式, WBS 的优势包括^[33, 37]: 1) 易于使用; 2) 便于部署; 3) 跨平台; 4) 广泛可用; 5) 集成与互操作等. 但也存在一些缺点, 如: 1) 访问延迟; 2) 图形化能力受限; 3) 易受攻击; 4) 稳定性差等.

鉴于 WBS 是分布式仿真基于 Web 技术的一种应用模式, Web 技术的发展会对其设计、开发、运行以及用户的访问模式产生重大的影响. 到目前为止, Web 技术的发展与演化主要经历了三个阶段^[37-38], 即 Web 1.0, Web 2.0 和 Web 3.0. 这三个阶段不是基于某种技术而划分, 而是基于应用理念而划分. 表 2 总结了网页技术的发展对 WBS 的影响.

2.1.1 Web 1.0

Web 1.0 的时间跨度大约为 1996 年至 2004 年, 主要特征是网页内容由少数人发布, 用户只能通过浏览器读取或使用服务供应商提供的有限内容及服务, 典型的例子是搜狐、雅虎等门户网站. 相应的, 该时期 WBS 所提供的功能也相对受限, 比较典型的应用模式是将仿真应用整体地不做实质改变地迁移到远端的服务器上, 仿真用户通过浏览器提交简单的服务请求, 后端服务器在执行完相应计算任务后将结果返回给用户.

尽管该时期也有一些基于网页的仿真应用, 很多是被创造出来以证明可以使用网页进行仿真, 而不是因为需要使用网页进行仿真^[39]. 同时, 仿真应用开发、部署与执行的架构或模式并没有发生本质变化, 仍然基于 DIS 或 HLA 实现分布式仿真的互联互通, 没有为模型的组合、重用提供新的能力^[40].

表 2 网页技术的发展对 WBS 的影响
Table 2 Impact of the development of web technology on WBS

类别	时间跨度	主要特征	仿真应用情况	优点	缺点
Web 1.0	1996 ~ 2004	用户只能“读”取网站上的内容, “写”的能力受限	应用较少, 更多的是证明可以基于网页实现仿真, 而不是需要使用网页进行仿真	简化访问; 降低对用户端设备的依赖	稳定性易受影响; 图形化能力有限; 用户对仿真的控制手段也相对匮乏
Web 2.0	2004 ~ 至今	用户间可以自由地交互, 用户既可以“读”也可以“写”	应用领域大大扩展, 并被吸纳到多种仿真标准, 如 HLA Evolved	广泛访问; 支持用户通过浏览器实现仿真服务的组合与集成	难以有效管理大量服务资源; 无状态的网页服务难以保存模型状态
Web 3.0	未成熟	主要强调对用户数字资产的尊重与保护, 核心技术是区块链	还处于探索阶段	促进仿真资源的共享; 增强用户数据的隐私保护	核心的区块链技术仍面临伸缩性和吞吐率的问题; 与仿真结合的研究较少

Wiedemann^[41]指出, WBS 当前不能简化仿真的建模过程, 仍需要使用传统的仿真工具开发模型. 事实上, 该时期对于 WBS 的研究从 1996 年至 2000 年快速增长, 从 2002 年之后便开始迅速下降^[34, 37]. 很多学者将此归因于网页特征与基于网页的仿真的方法间存在失配.

总体而言, Web 1.0 阶段 WBS 的特征包括^[37, 39]: 1) 为用户提供丰富的应用访问能力, 可在任何支持浏览器的设备上获取仿真服务; 2) 主要强调前端用户访问模式的变化, 后端仿真应用的设计与部署与传统的分布式仿真相比没有本质区别; 3) 仿真模型常基于组件化技术开发, 能提升建模人员在模型开发和部署时的效率, 但用户不能基于网页实现模型的自由组合与集成.

2.1.2 Web 2.0

Web 2.0 的时间跨度约为 2004 年至今, 相比于 Web 1.0, Web 2.0 更加注重用户间的交互. 从使用模式上看, 用户可以自由地发布内容, 在显著丰富互联网内涵的同时, 也提升了前后端之间交互的能力及水平. 更为重要的是, 同时期出现的网页服务技术以及软件工程领域提出的面向服务架构 (SOA) 给 WBS 带来了新的活力.

该阶段对 WBS 的应用与研究也相对较多. 部分学者^[42-44]将现有的仿真系统转化为网页服务, 使得用户可以通过浏览器实现仿真的定义、提交、可视化以及结果分析. 网页服务包含两个主要框架: 基于简单对象访问协议 (Simple object access protocol, SOAP) 和基于 RESTful^[45]. 其中, SOAP 是 WBS 的主流协议, 有着大量的研究成果^[46-48]. 然而, Arroqui 等^[49]的实验表明, RESTful 能够将网页服务所需带宽减少 24%. 为此, Al-Zoubi 等^[50]提出了 RESTful 互操作仿真环境 (RISE), 并将该中间件应用于基于 DEVS 的地理信息系统仿真^[33]以及云上仿真资源的管理^[34].

同时, 研究人员也对基于网页的仿真框架和协议进行了深度的探索. 美国海军研究生院等单位在 2004 年提出了扩展建模与仿真框架 (Extensible modeling and simulation framework, XMSF)^[51], 利用一系列网页使能技术促进 HLA 在更大范围的互操作与重用. 为进一步提升 HLA 对网页服务的支持, IEEE 在 2010 年发布了 HLA Evolved, 通过在接口描述中增加对 WSDL 的支持, 以及提供 WSPRC 组件, 使得 HLA 从本质上具备了向 Web 跨平台扩展的能力^[52-53].

总体来说, WBS 的内涵在该阶段得到了很大的拓展并取得丰硕的研究成果, 但也面临一些问题.

Web 2.0 阶段 WBS 的特征包括^[30, 54]: 1) 在强调用户基于浏览器的广泛可用性的同时, 对后端仿真应用的设计与开发也提出了新的要求, 如组件化设计, 服务化封装; 2) 用户不仅可以访问远端的仿真服务, 借助于标准化的服务描述, 也可以利用相关工具实现仿真服务的按需组合与集成, 获得定制化仿真的能力; 3) 难以进行有效的资源管理, 尤其是当具有大量服务资源的时候; 4) 网页服务的无状态特性使得其难以保存模型的状态.

2.1.3 Web 3.0

与 Web 2.0 相比, Web 3.0 更难被定义, 这在很大程度上是因为其仍处于发展的初级阶段. 尽管 Web 3.0 在 2006 年即在 Zeldman 的一篇博客中被首次提及^[55], 它的内涵以及所使用的关键技术仍存在较大争议.

在 Web 3.0 概念初期, 主要强调网页的个性化、定制化和智能化, 有时也称之为语义网络^[37]. 部分学者^[56-58]尝试使用本体实现对仿真资源的描述, 以促进资源的重用. 随着互联网内容的增加以及区块链技术的发展, Web 3.0 也被赋予了新的含义, 该阶段更加强调对用户数字资产的尊重与保护. 对于建模与仿真领域而言, 以区块链为底层支撑的 Web 3.0 也会带来很多机会与挑战: 1) 能够促进不同单位间仿真资源的共享; 2) 增强对用户数据隐私的保护; 3) 实现安全可靠的仿真资源管理.

2.2 面向 SOA 的仿真系统开发

建模仿真的服务化同样对仿真应用的开发提出了要求, 软件工程领域的面向服务架构 (SOA) 则可以作为应用开发的指导思想.

SOA 并不是一种新思想, 最早在 1996 年由 GartnerGroup 提出. 很多学者和组织都曾给出对 SOA 的定义. Liu 等^[29]认为“SOA 概念将服务的创建与使用相分离, 并以简单的方式实现独立服务的灵活组合.”美国国防部建模仿真参考架构 (Defense modeling and simulation reference architecture, DMSRA)^[15]则将 SOA 定义为“支持具有指定结果的可重复业务活动的逻辑表示的体系结构样式”.

总的来说, SOA 是一种与具体实现无关, 用于构建大规模分布式软件系统的开发范式. 它将应用的不同功能单元定义为服务, 并通过标准的接口和描述实现服务间的松耦合、互操作和可重用.

SOA 包括服务提供者、服务请求者和服务代理三个角色, 彼此之间通过发布、发现和绑定调用三种操作实现交互^[59]. 将 SOA 应用于仿真系统开发的好处包括^[30, 54]: 1) 资源重用; 2) 易于集成与管理;

3) 更高的灵活性; 4) 降低成本。

本节以 SOA 在不同阶段的主流实现技术为线索, 对面向服务的仿真应用开发的发展过程进行梳理。表 3 比较了面向服务架构的不同实现技术。

2.2.1 组件技术

Davis 和 Tolk^[60] 从语法、语义、语用、假设和有效性等五个方面, 对基于组件的多分辨率建模和模型组合问题进行研究。Hofmann^[61] 以军事仿真系统为例, 从三个层面对如何将系统分解为组件进行了探讨。同时其指出, 在分解过程中, 最重要的是降低系统的复杂度, 而不是组件重用。

在传统的分布式组件技术中, 服务是一个可以通过编程接口访问的软件组件, 用户使用分布式组件协议 (如 CORBA)、分布式计算对象模型 (DCOM、RMI、BOM、EJB 等) 访问服务。史扬等^[4] 基于 Java EE/EJB 组件规范构建了一个面向服务的作战仿真平台验证系统。Gustavson 等^[62] 在 HLA Evolved 基础上提出了一个基于 SOA 的建模与仿真组合框架, 以支持仿真运行前和运行过程中模型实例的动态聚合与解聚。然而, 已有的技术大都与具体编程语言或平台相关 (如 DCOM 只支持 Windows), 程序与服务是紧耦合关系, 这导致系统变得脆弱^[30, 54]。

为改进组件的重用性, 一些学者尝试将 SOA 与基于组件的建模仿真范式离散事件系统规范 (Discrete event system specification, DEVS) 相结合。Mittal 等^[63] 对 DEVS 统一过程 (DEVS unifies process, DUNIP) 进行了讨论, 并使用 DEVSML^[64] 框架提供不同语言实现的 DEVS 模型在 SOA 中间件上的组合能力。文献 [65–66] 提出了一个服从 SOA 的 DEVS 仿真框架, 并对服务组合问题进行了研究。

2.2.2 网页服务

网页服务是网页技术与分布式组件技术融合的产物^[30]。在网页服务的技术标准中, WSDL 用于服务的发布, SOAP 用于服务间的交互, 统一描述、发现与集成规范 (Universal description discovery and integration, UDDI) 用于服务的查找与定位。网页服务具有自包含、松耦合、相互独立且具有标准化描述的特征, 通过将仿真系统以网页服务的形

式进行开发、部署和执行, 能够为用户提供更加强化的仿真应用与管理能力。

在很多文献中, SOA 常与网页服务一起使用, 这是因为网页服务技术的成熟大大促进了 SOA 在企业中的应用。SOA 定义了服务如何相互理解以及如何交互, 是一种模型架构; 网页服务则是具体的实现方法, 是 SOA 众多实现手段的一种。相比其他分布式技术, 网页服务的优点包括^[54]: 1) 松耦合; 2) 自描述与自适应; 3) 实现无关; 4) 易于集成与移植。

何强等^[54] 以网页服务作为实现技术, 对基于 SOA 的仿真服务系统进行了研究。其中系统分为仿真服务层、仿真应用层以及仿真平台层, 每一层以网页服务的形式对外提供接口, 彼此之间通过 SOAP 协议进行通信。宋莉莉^[30] 对仿真的互操作和模型的组合性进行了研究, 以网页服务为基础提出了一种基于 SOA 的建模与仿真框架, 并重点讨论了仿真服务的描述、匹配和选择技术。Park^[67] 在其博士论文中对如何在松耦合资源 (如志愿计算网络) 中执行并行离散事件仿真 (Parallel discrete event simulation, PDES) 进行了研究, 并提出了一个基于网页服务的主/从仿真架构, 该架构能为错误容忍、负载均衡、时间同步等功能提供支持。

然而, 考虑到分布式仿真的特殊性, 仿真服务与商业领域的网页服务还存在一些差别^[30], 包括: 1) 仿真服务经常是有状态的, 而网页服务一般是无状态的; 2) 仿真服务需要进行频繁的时间同步, 网页服务没有这种需求; 3) 仿真服务通过底层运行支撑环境进行交互, 网页服务通常直接利用 SOAP 等协议进行通信。

2.2.3 微服务

在基于组件和网页服务的 SOA 应用中, 服务往往采用的是单体架构, 处理请求的所有逻辑运行在唯一的进程中^[68]。这种体系结构下的系统缺乏独立性和灵活性, 服务必须同时扩展和启停, 不能满足系统快速迭代的要求。为克服上述挑战, 微服务 (Micro-service) 架构已成为构建超大规模信息系统的新架构风格^[69]。

通过综合 Lewis 等^[68] 和 Namiot 等^[70] 对微服务

表 3 面向服务架构的不同实现技术的对比

Table 3 Comparison of different implementation technologies for service oriented architecture

类别	技术/标准/架构	粒度	部署策略	可移植性	自动化部署	仿真应用情况	服务状态
组件技术	CORBA、BOM、DCOM 等	单体服务	虚拟机	一般	不支持	应用广泛	支持
网页服务	WSDL + SOAP + UDDI	单体服务	虚拟机	较好	支持	应用广泛	不支持
微服务	微服务架构	微服务	容器	较好	支持	发展阶段	支持
纳米服务	无服务器架构	函数	FaaS 平台	一般	支持	探索阶段	不支持

的定义, 可将其特点总结为: 1) 轻量级, 只实现单一功能; 2) 独立开发; 3) 自动化部署; 4) 接口定义良好. 总的来说, 微服务是 SOA 的一种实现, 但它要更加轻便、敏捷和简单. 与传统 SOA 服务间的区别主要在于服务的粒度^[26]. 在云计算环境下, 微服务常采用容器这一轻量级虚拟化方式进行部署.

文献 [26] 研究了仿真的容器化开发与部署以实现高效的海军训练, 并对容器化改造时的经验教训进行了总结. 其中, 模型及仿真支撑服务 (如 RTI 组件) 以容器的方式部署于德国和爱尔兰的两个数据中心, 地理分布的真实、虚拟和构造的仿真成员基于 HLA 实现互联. 最终的测试结果表明, 容器化能够促进 HLA 联邦的开发与管理, 但在网络化环境下, Docker 使用的网络技术如 Weave 网络会延长数据的传输链路, 进而增大时延. Zhou 等^[10]介绍了一种基于微服务的电网仿真训练平台的开发方法, 基于云计算技术实现仿真任务的高效协作. 文献 [71–72] 开发了一个基于微服务的动态编排器 MiCADO, 为云上的仿真应用提供自动伸缩的计算资源. 刘永奎等^[73]针对云制造场景提出了一种基于微服务架构的云制造调度仿真系统, 支持对不同调度算法的性能进行测试与评估. Kecskemeti 等^[74]则提出了一种方法学, 以指导如何将单一的大服务分解为多个微服务, 以实现更好的弹性.

基于微服务和容器的开发与部署能够为仿真应用的管理带来诸多好处, 但不能盲目选用. 微服务上云的最佳方式是全新开发, 对现有系统的微服务改造工作量很大. 考虑到现有的大量基于单体服务架构的 SOA 应用, 应谨慎地使用微服务.

2.2.4 纳米服务

微服务架构下, 微服务以容器的方式部署于云中心. 尽管微服务能带来很多灵活性, 但也存在一些效率方面的问题. 比如, 必须保持至少一个容器在运行状态, 以随时响应用户的请求, 而这被认为是云计算范式下很重要的一类资源消耗^[75].

基于该考虑, 一种被称为无服务器计算 (Serverless computing) 的新型计算范式被提出. 其使用函数即服务 (Function as a service, FaaS) 的概念和第三方支撑服务来替代服务器的管理与运维功能, 同时基于时间复用的原则增加资源效率^[19]. 在文献中, 一般不对无服务器计算和函数即服务这两个概念做特殊区分. 其中, “函数” 也被称为纳米服务 (Nanoservice), 是更细粒度的服务, 也可被归为微服务的一部分^[19, 24].

相应的, 对纳米服务实行统一管理的中间件也被称为 FaaS 平台. FaaS 平台本质上是一个事件处理系统, 基于事件实现函数的调用^[76]. 这一特性使

之很适合基于事件的应用, 如部分离散事件仿真系统. 在 FaaS 平台中, 函数只提供与服务、安全或计算高度相关的功能, 而传统上集中式服务器提供的功能都交由第三方服务, 如身份认证、数据管理等^[19].

由于无服务器计算还处于发展的初期, 基于纳米服务实现仿真应用开发的工作也相对较少. Kritikos 等^[77]提出了一种基于无服务器计算范式的新型仿真服务架构 SimaaS, 其通过无限制、并行化的函数调用实现仿真的加速, 同时依赖多云平台实现仿真应用的跨云部署和自适应执行. Kratzke 等^[78]借鉴商业领域云原生应用的概念, 提出了云原生仿真的架构并探讨了云原生仿真服务的发展趋势, 重点讨论了微服务和纳米服务对仿真应用在开发与部署上的影响. Villamizar 等^[79]对单体、微服务和纳米服务的对比实验表明, 纳米服务最多可减少 75% 成本.

总体而言, 无服务器计算架构更加的分布与去中心化, 更细粒度的服务也提升了服务的弹性、伸缩性和动态迁移能力. 然而其也存在一些弊端^[15, 80]: 1) 与数据的高度分离导致难以应对有状态服务; 2) 安全层面面临更多的威胁; 3) 把部分任务交给了客户端, 增加了客户端复杂度.

2.3 服务化计算基础设施

仿真技术伴随计算机技术的发展而发展. 要实现建模与仿真的服务化, 还需要底层的计算基础设施提供高效的运行支撑和资源管理. 本节主要以网格计算和云计算技术的发展为牵引, 探究计算基础设施的变化对仿真的影响. 表 4 总结了不同计算基础设施对仿真的影响.

2.3.1 网格计算

传统的仿真模式下, 用户只能基于已有的软硬件资源进行仿真, 限制了仿真的规模. 网格计算的出现则为大规模复杂系统的仿真提供了可能. 网格计算利用虚拟组织 (Virtual organization) 的概念, 将地理分散的资源集成为统一的计算基础设施. 这种方式, 能够显著改进仿真资源的管理、数据的访问、系统的安全和容错能力, 进而受到了很多学者的关注^[30].

Rycerz 等^[81]研究了如何基于开放网格服务架构 (Open grid service architecture, OGSA) 高效、容错地执行 HLA 仿真应用. 在文献 [82–84] 关于 HLA-Grid 的工作中, 提出了“成员–代理–RTI”的体系结构以将 HLA 应用迁移到网格中, 实现按需使用的 RTI 服务和仿真联邦的动态组合与集成. 其中, 成员代码在本地客户端执行, 网格端则部署成员的代理及 RTI 服务. 在执行时, 成员通过服务化的 HLA API 调用 RTI 服务, 成员代理则代表成员

表 4 不同计算基础设施的对比

Table 4 Comparison of different computing infrastructures

类别	统一运维管理	远端访问	服务化	虚拟化	弹性扩展	使用成本	安全性
本地集群	不支持	不支持	不支持	不支持	较差	高	好
网格计算	支持	支持	支持	不支持	一般	低	一般
云计算	支持	支持	支持	支持	良好	低	一般

经由真实的 RTI 在网格端实现通信。

国内, Li 等^[85] 在 2005 年提出了面向服务的仿真网格系统 Cosim-Grid, 实现仿真的协作式开发以及高效的运行与管理. 张卫^[86] 对如何基于网格实现并行分布式仿真进行讨论, 提出了一种分层的仿真网格架构, 同时对 RTI 功能的服务化改造、服务描述和仿真资源管理等问题进行了重点研究. 蔡檀^[87] 则研究了如何将网格计算与云计算的理念相结合以对仿真提供支撑.

仿真网格能够为建模与仿真提供很多优势^[31], 如动态资源共享与自动化管理. 然而, 它也存在一些缺点, 如资源由不同机构提供、主要针对大型科学应用等, 这也是网格计算逐渐被云计算替代的部分原因.

2.3.2 云计算

云计算被认为是网格计算、SOA 和虚拟化等技术融合的产物. 与网格计算跨组织共享资源不同, 云服务商通常是资源的唯一拥有者. 对云计算最广泛接受的定义由美国国家标准技术研究院 (NIST) 给出. 其中云计算具有五个基本特性 (按需服务、广泛访问、资源池化、弹性可扩展、服务可计量), 三种服务模型 (基础设施即服务 (IaaS)、平台即服务 (PaaS)、软件即服务 (SaaS)) 以及四个部署模型 (公有云、私有云、社区云、混合云)^[15].

基于云的仿真 (以下简称云仿真) 集成了基于网页的仿真和云计算技术以管理多种仿真资源并构建不同的仿真环境^[16]. Fujimoto 等^[3] 认为, 云计算为仿真带来的好处包括: 1) 降低用户成本; 2) 简化仿真的使用; 3) 便于仿真资源的管理与维护; 4) 云平台所具有的错误容忍和负载均衡能力. 事实上, 云计算被认为是解决当前和未来大规模仿真可能面临问题的关键技术^[3], 并且在学界和国防工业部门都受到了广泛关注.

并行仿真关注的是充分利用高性能计算资源对仿真加速, 分布式仿真则更多关注的是地理分布仿真应用间的互联互通. 为弱化并行仿真与分布式仿真间的差异, 基于 DEVS 规范, Risco-Martín 等^[88] 提出了一种云环境下统一的并行与分布式仿真架构模型, 基于 XML 配置文件, 无需修改应用程序即可实现仿真应用在物理机、虚拟机、容器上的灵活弹

性部署. 为更好地实现监察无人机的任务分配、轨迹设计, 部分学者基于微服务和容器技术实现无人机仿真系统的云端高效部署. 其中, Bordón-Ruiz 等^[89] 采用了 DEVS 规范实现仿真模型的标准化开发, Matlekovic 等^[90] 则利用 Kubernetes 容器部署平台实现服务的快速开发、更新和替换. 美军联合参谋部下属的联合训练委员会开发了一个基于云的、网页使能的联合训练环境 (Joint training tools, JTT)^[91]. 英国陆军在其联合训练转型计划 (CTPP)^[92] 中, 则重点对 VR、机器学习、云计算技术进行了研究.

然而, 云平台也会为仿真的运行带来一些挑战^[93], 如: 1) 安全性挑战; 2) 云平台擅长提供不频繁、高带宽的数据传输, 但仿真应用往往是高频、低带宽的通信; 3) 难以保证实时性.

3 基于云的建模与仿真服务化

实现建模与仿真的服务化是一个复杂的系统工程. 从参与方的角度看, 包括仿真用户、仿真开发人员、仿真运维人员以及计算资源提供方; 从仿真生命周期的角度看, 包括设计、开发、校验、部署、运行、管理、结果分析等; 从仿真的类型看, 又包括分析仿真、装备仿真和训练仿真. 本节将从建模与仿真服务化的构建原则、基本架构和应用模式三个角度对其内涵进行阐述, 为后续建模与仿真服务化的进一步发展提供一定的参考.

3.1 建模与仿真服务化原则

建模与仿真产品对于工业和国防部门具有很高的价值, 因此, 一个新的建模与仿真范式应该能够解决现有应用模式所面临的各种痛点问题^[15, 18].

从仿真用户的角度看, 应能支持大量用户使用浏览器等瘦客户端, 基于各自的目的, 同时且自发地访问各类建模与仿真相关的产品、数据和知识. 从仿真开发的角度看, 应提供标准化的开发工具和各类模板 (如基于 MDA 的平台无关模型), 同时支持通过服务的描述、发现和组合实现仿真资源的高效重用和仿真系统的快速集成. 从仿真运行与管理的角度看, 应提供自动化管理工具, 实现仿真应用的正确运行、仿真资源的高效管理、仿真过程的安全可靠.

为实现上述能力, 需要以恰当的方式将仿真应用部署于云端并通过网络实现远程的访问. 然而, 将仿真迁移到云端是一个缓慢的过程^[26], 需要充分考虑仿真应用和云数据中心的特点.

从仿真的角度看, 不同类型的应用可能具有完全不同的需求. 对于分析仿真而言, 往往涉及复杂的模型解算任务, 需要尽可能快地返回仿真结果. 对于装备仿真而言, 由于实装的接入, 具有非常高的实时性要求, 可能需要使用裸金属服务器甚至将仿真任务下放到边缘侧执行^[94]. 对于训练仿真而言, 往往涉及大量地理分布的人员、装备, 需要充分考虑网络传输时延对仿真性能的影响.

从云数据中心的角度看, 建模与仿真服务化并不仅仅是将仿真应用的执行环境由指定的硬件换成云端的虚拟机. Kratzke 和 Siegfried^[10] 借鉴云原生应用^[95] 的概念, 提出了四层的云仿真成熟度模型: 云可用 (Cloud ready)、云友好 (Cloud friendly)、云弹性 (Cloud resilient) 和云原生 (Cloud native). 对于云原生仿真应用而言, 其应由小的、可独立部署和替换的仿真服务组成, 且支持横向扩展^[19]. 这意味着仿真应用需要分解成微服务或纳米服务以充分利用云平台弹性扩展和自动编排的能力. 对于仿真应用而言, 考虑到其紧耦合的特性, 在分解时需要对应应用内部的工作机理进行细致的研究.

另一方面, 云服务并不总是标准化的产品. Kratzke 等^[78] 对云计算标准的研究表明, 只有一些核心组件, 如虚拟化资源实例, 是以标准化的方式提供. 但对于负载均衡、消息队列、弹性扩展等用于构建云原生应用的“胶水”服务, 不同的运营商往往采用不同的方案, 为应用在云间的迁移带来困难. 例如, Instagram 被 Facebook 收购后, 花费了超过一年时间将部署于 AWS 云上的所有服务迁移到 Facebook 数据中心, 期间还出现了服务的中断^[10]. 这一特点给部分仿真应用, 尤其是需要跨云的大型应用的设计、开发和部署带来了困难.

3.2 建模与仿真服务化架构

借鉴云计算的三种服务模型 (IaaS、PaaS 和 SaaS), Cayirci^[16] 在 2013 年给出了建模仿真即服务 (MSaaS) 最为广泛接受的三种服务模型: 建模即服务、模型即服务、仿真即服务, 具体如图 1 所示. 其中, MSaaS 层位于传统的云服务层之上, 为用户提供建模与仿真相关服务.

为对建模与仿真的服务化建设提供更加具体的指导, 很多基于云的建模与仿真架构相继被提出, 包括通用的^[2, 19, 96-97], 面向教育的^[8], 面向交通的^[9],

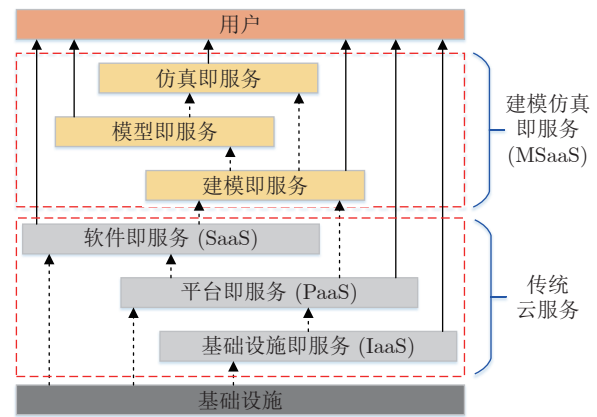


图 1 MSaaS 服务模型^[16]

Fig.1 Service model in MSaaS^[16]

以及面向作战的^[98-100]. 在当前公开存在的各种云仿真架构中, 大多采用了分层的设计, 且与 Cayirci 提出的 MSaaS 服务模型总体上保持一致^[18]. 事实上, 建模与仿真的服务化不单单是利用现有的云服务模型, 还需要对云服务模型本身进行适应性改造. 通过对现有架构的综合分析, 我们给出了一个通用的云仿真架构, 如图 2 所示. 该架构共有五层, 包括基础设施层、平台层、服务层、应用层和门户层. 其中服务层与应用层一起, 对应着 MSaaS 服务. 安全隐私和技术标准则贯穿所有层级.

1) 基础设施层. 用于提供仿真运行所需的基础硬件及相应虚拟化资源, 包括计算资源、存储资源与网络资源. 现有的云仿真架构都包含这一层. 该层资源可由一个或多个云服务供应商提供.

2) 平台层. 该层集成了传统云服务 PaaS 层的功能以及用于支撑仿真运行的功能. 其包含两个子层以及数据存储服务. 其中, 基础平台子层的功能包括资源监视、资源管理、服务编排、任务调度等, 可由商业的云平台中间件提供. 同时, 我们添加了仿真平台子层, 其功能包括仿真交互、时间同步、实体迁移等与仿真运行密切相关的基础服务. 也有一些学者将这些功能都归入了服务层 (李伯虎等^[2], Liu 等^[96], 齐和平等^[100]), 但我们认为这些服务是所有仿真应用都需要的共性支撑服务, 应该作为平台层的一部分.

3) 服务层. 该层的主要功能是对各类仿真资源和服务进行高效的管理与维护^[98], 是所有云仿真架构的核心层. 与王会霞等^[99] 和齐和平等^[100] 所提架构不同的是, 我们把服务的组合与集成放在了应用层, 服务层只用于对各类服务的管理. 服务可基于有无状态^[19]、部署形态^[8, 19, 97] (虚拟机, 容器或 FaaS 平台)、粒度^[19] (单体服务、微服务、纳米服务) 等进

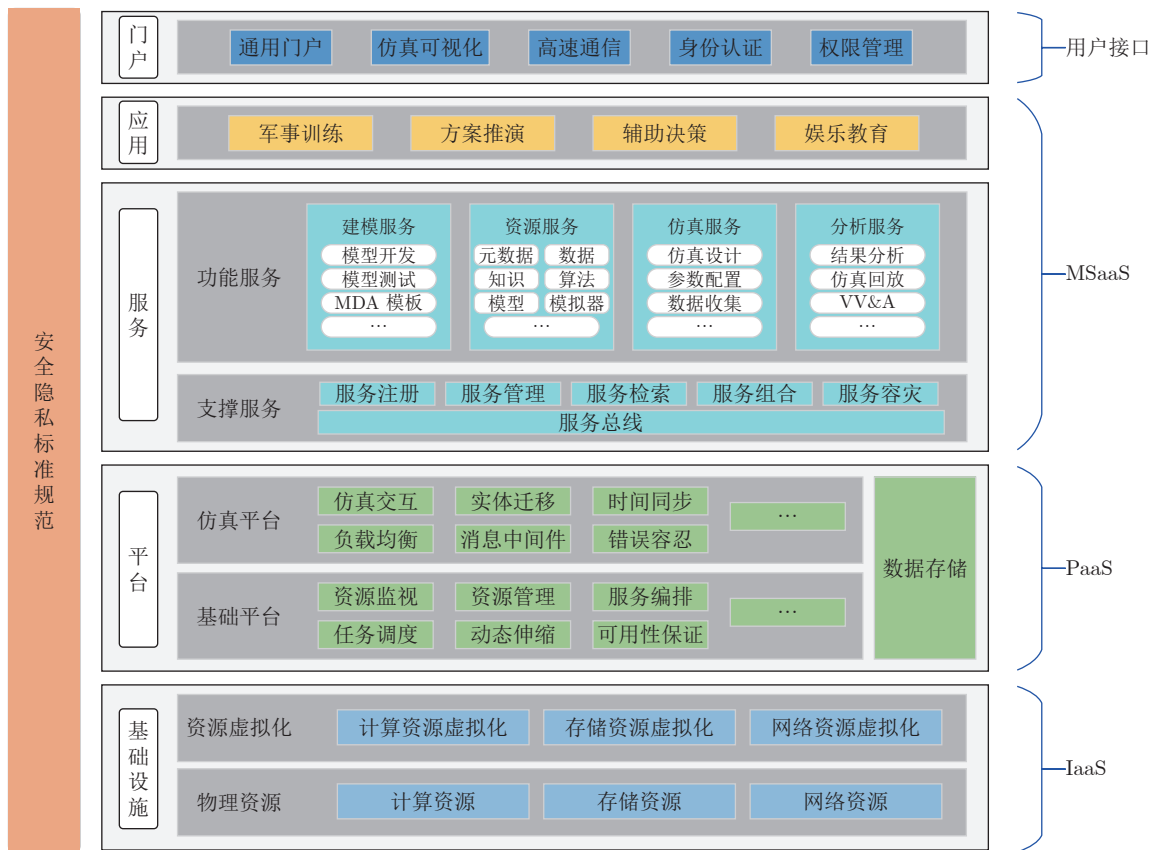


图2 建模与仿真服务化通用架构

Fig.2 General architecture of MSaaS

行划分。我们基于功能类型将服务层分为了两个子层：功能服务子层和支撑服务子层。

其中功能服务子层包含所有与建模和仿真相关的服务，具体又可分为建模服务、资源服务、仿真服务和分析服务。建模服务包括模型开发、模型测试、MDA模板服务等与建模相关的服务。资源服务则是将现有的各类仿真资源以服务化的方式封装而得到，包括（元）数据、知识、算法、模型、模拟器等。仿真服务用于支撑用户正确、高效地执行仿真，包括仿真设计、参数配置、数据收集等。分析服务则用于对仿真结果进行分析与评估，包括结果分析、仿真VV&A、仿真回放等。

支撑服务子层则用于提供对服务的高效管理，基于服务总线实现服务的注册、管理、检索、组合与容灾。

4) 应用层。主要为不同类型用户提供所需的建模仿真应用，满足多种应用场景，包括军事训练、方案推演、辅助决策、娱乐教育等。

5) 门户层。为用户访问建模仿真服务提供相应的工具与手段，同时对用户进行管理。具体功能包括通用门户、人机交互、身份认证、权限管理以及资

源开设等。

3.3 建模与仿真服务化应用模式

在建模与仿真服务化场景中，用户能够在任何时间、任何地点，在任何能够运行浏览器的设备上，通过网页访问所需的仿真服务并获取仿真结果。其中，浏览器也称为前端，提供建模与仿真服务的云仿真平台也被称为后端，如图3所示。



图3 建模与仿真服务化应用示意图

Fig.3 Application schematic diagram of MSaaS

从应用的角度看，建模与仿真即服务（MSaaS）模式下的典型应用流程可概括为^[2, 9, 13, 26]：1) 用户在前端提交仿真需求；2) 基于用户需求，在后端执行仿真服务的搜索、发现、组合、集成以及自动化编排，形成所需仿真应用；3) 执行仿真，并进行动态监视与管理；4) 收集数据，向用户反馈仿真结果。

根据建模与仿真服务的粒度和组合方式, MSaaS 也具有四种不同的部署形态^[16]: 1) 独立仿真应用, 通常以网页服务的方式提供, 一些军事组织选择使用私有云提供独立的作战模型, 如 JCATS (Joint conflict and tactical simulation) 和 JTLS (Joint theater level simulation)^[101]; 2) 仿真应用联邦, 多个独立的应用通过分布式云建立联邦; 3) 可组合仿真应用, 云不提供完整的应用, 而是提供软件(功能)模块, 如气象、光照等; 4) 自动组合仿真应用, 软件模块之间可实现自动的发现与组合. 图 4 给出了第二和第三种云仿真部署形态的示例.

从运行的角度看, 仿真应用基于所涉及的实体规模被部署于单个 VM 或虚拟集群之上, 各个仿真成员之间通过可靠的有线网络进行交互. 当仿真规模很大, 或某些资源只存在于特定的云数据中心时, 也需要考虑应用跨数据中心的运行与管理问题.

然而, 这种模式存在一个问题, 即用户与后端的距离很远, 导致数据的传输时延增大. 特别地, 对于一些有较高实时性要求, 或者需要将仿真结果以可视化的形式反馈给用户的仿真应用而言, 在带宽和时延上面临着巨大挑战. 对于实时性问题, 一种可行的解决方案是利用边缘计算的思想^[94], 将与用户交互频繁的任务从仿真应用中解耦, 并部署在靠近用户的一侧^[98]. 对于大量视频流的实时传输, 可借鉴云游戏中的解决方案^[102], 即在后端将逻辑计算与图形渲染的指令解耦, 传输时只传输图形渲染的相关指令, 而将具体的渲染工作放在边缘端或者用户端.

4 关键支撑技术

尽管基于云的建模与仿真服务化能够带来很多好处, 但将仿真应用迁移到云端仍是一个很缓慢的过程^[26], 也面临着很多挑战. 本节将从访问、开发和

运行与管理三个角度对基于云的建模与仿真服务化的关键支撑技术进行探讨, 总体结构如图 5 所示.

4.1 用户访问相关技术

4.1.1 通用门户

通用门户技术是为用户提供各类建模与仿真服务的关键使能技术. 它应具有四种能力: 1) 支持用户在任何时间、任何地点以最小化的软硬件需求访问远端的建模与仿真服务, 一种典型的解决方案是采用浏览器和网页技术; 2) 支持多样化的输入设备^[2,9], 如浏览器、触摸屏、物理传感器、逻辑传感器等; 3) 能够提供丰富的工具, 帮助用户实现仿真需求的确定, 服务的检索、发现、组合与集成以及仿真的运行与分析; 4) 支持不同的可视化组件部署方案, 既能接收控制流实现本地的可视化渲染, 也能直接接收视频流在本地进行解码和播放^[102-103].

4.1.2 智能人机交互

良好的沉浸感是提升仿真应用效果的关键, 而现有的键盘+鼠标的方式与真实世界中用户的交互模式间存在着巨大鸿沟, 降低了用户的体验. 智能人机交互技术则利用最新的语音交互、手势识别、动作捕捉等人工智能技术, 通过多模态的交互手段实现人机之间的自然交互. 这种交互模式对于军事训练等需要用户实时参与的仿真应用而言具有非常重要的意义. 其中, 美军在 2012 年推出了陆军步兵训练系统 (DSTS)^[104], 它是全球范围内首款完全沉浸式的班组虚拟训练系统. 通过捕捉士兵的动作并实时地在虚拟环境中构建士兵的数字替身, DSTS 能够实现虚实融合的高效训练.

4.1.3 高效网络通信

随着仿真模型变得更加复杂以及仿真规模的不断扩大, 需要更大的带宽来传输仿真数据. 同时, 真实和虚拟仿真成员的加入也对数据传输时延提出了

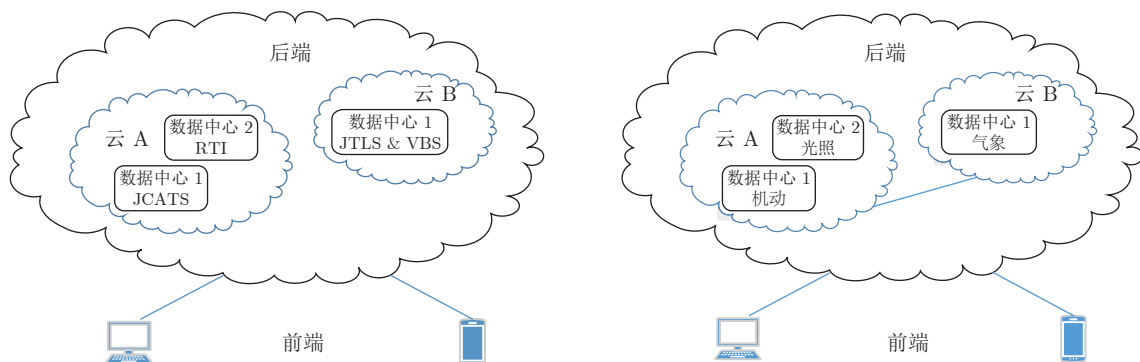


图 4 云仿真部署形态示意图^[16]

Fig. 4 Schematic diagram of cloud simulation deployment form^[16]

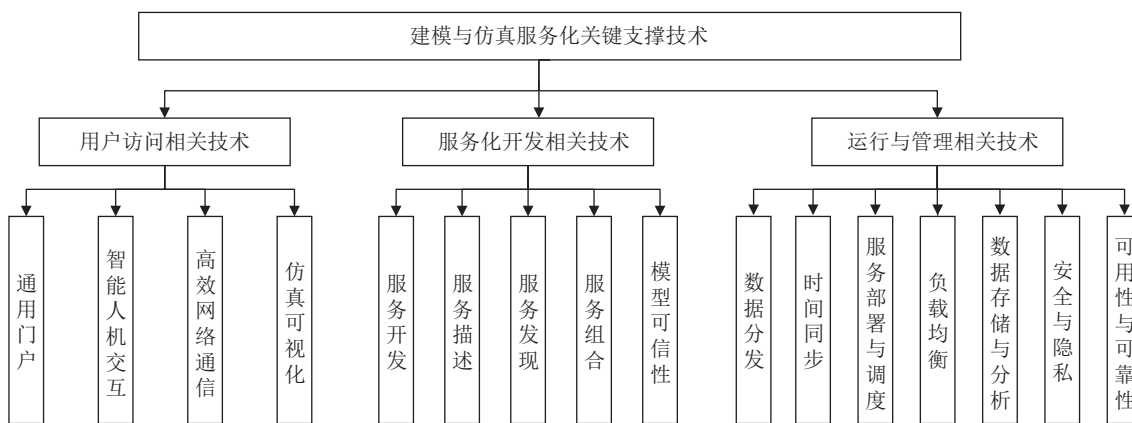


图 5 建模与仿真服务化关键支撑技术

Fig.5 Key supporting technologies for MSaaS

非常高的要求. 以 5G 为代表的新一代网络通信技术则是解决该问题的一个可行方案^[105]. 从网络基础设施的角度看, 边缘节点分布广泛, 因成本原因边缘到中心、边缘到边缘之间无法构建专线连接, 而传统的 Internet 路由协议则容易导致丢包、网络拥堵等问题. 为满足 LVC (Live, virtual and constructive) 仿真所需的高质量通信需求, 可通过实时测量网络服务质量 (Quality of service, QoS), 动态选择最优路由策略^[106].

4.1.4 仿真可视化

仿真的结果应该能够以可视化的方式实时向用户反馈, 为用户提供良好、逼真的仿真体验. 仿真可视化应该具有三个层面的特征: 1) 丰富的表现形式, 支持以 2D/3D、VR、AR 等多种方式将仿真结果投射到用户视野前, 提供各种类型的感官体验; 2) 多样的内容, 利用统计学方法和工具提供各类图、表等可视化分析结果, 为用户的下一步决策提供更好的支撑; 3) 逼真的画面, 利用高效的渲染引擎生成高质量的多分辨率仿真场景, 同时支持与用户的实时互动.

4.2 服务化开发相关技术

4.2.1 服务开发

服务的开发涉及两个层面的内容: 旧有系统的服务化封装与改造以及新服务的构建. 旧有系统的服务化改造并不是一个简单的问题, 一般而言, 有三种方法可供选择^[44]: 1) 适配器; 2) 重构; 3) 结合两种方法. 对于从头开始开发仿真服务, 则需要充分考虑云计算环境的特点. NASA 在 2018 年的研究表明^[107], “在所有情况下, 基于 NASA 计算设施运行仿真的成本要比在 AWS 公有云上的成本少 2~12 倍”. Kratzke 等^[19] 将此结果归因于基于云的仿

真应用需要遵循一种云原生的架构风格, 如微服务或纳米服务等. 考虑到传统的基于 HLA 的仿真应用大多采用单体部署的模式, 需要对其进行更细粒度的改造.

4.2.2 服务描述

Hannay 等^[24] 指出, 服务描述的内容包含三个部分: 1) 服务接口, 描述服务的功能和操作, 用于实现语法互操作; 2) 服务契约, 其既在语义层面对接口的功能和操作进行详细说明, 也对服务的非功能需求进行规范; 3) 服务模型, 用于描述服务所支持的使用环境和上下文, 用于实现语义互操作. 服务可能具有不同的抽象层级、不同的表现形式以及不同的分辨率, 需要在不同的层级对服务进行规范化的描述, 为服务的发现与组合提供支撑. 事实上, 可采用常用的资源描述方法对服务进行描述, 包括^[108]: WSDL、网页服务资源框架 (Web services resource framework, WSRF)、本体语言、对象元模型等. 例如, 杜楠等^[108] 采用对象元模型实现对各类 LVC 仿真资源的形式化描述和标准化封装, 以服务的方式为用户提供 LVC 仿真资源.

4.2.3 服务发现

在设计、开发和执行仿真应用的过程中, 需要搜索仿真和数据资产, 实现仿真资源 (服务) 的高效重用. 随着仿真资源的持续发展与建设, 云仿真平台中存储的仿真服务的数量也不断增加, 其所表现出的复杂性、异构性和动态性也为仿真服务的发现带来了巨大挑战. 本质上, 服务发现是以恰当的服务描述为基础, 基于用户的功能和非功能需求寻找相匹配服务的过程^[109]. 当服务被准确描述时, 服务发现任务常被转换为文本匹配任务. 现有服务发现方法可以粗略地划分为语法级服务发现和语义级服务发现^[110]. 语法级服务发现主要关注语法层面的关

关键词匹配, 通常结合多种信息检索信息. 然而, 当关键词数量不足或具有多种含义时, 匹配的精准度会显著下降. 语义级服务发现则同时结合了本体技术, 通过提供形式化的服务描述克服上述问题.

现有的服务发现相关研究中, 大多采用了语义级服务发现技术. 为实现网页服务的快速发现与精准匹配, Huang 等^[110] 提出了一种基于卷积神经网络的语义级服务匹配算法. 为更好地捕捉用户需求的特征, 论文使用三种不同的方法从不同侧面对用户输入的关键词进行处理, 生成多维特征. 在构建神经网络时采用了多维相似度矩阵作为输入, 实现更加精准的服务匹配. Al-Sayed 等^[111] 则提出了一种标准化的智能云服务发现框架, 基于本体技术实现对云服务的功能和非功能特征的标准化语义描述.

基于服务的方法依赖于高度的标准化和自动化来充分发挥其优势, 服务的发现则是其中的重要一环. 事实上, 北约 MSG-136 小组在其关于 MSaaS 架构的研究中, 重点关注了服务发现技术和服务组合技术, 同时强调了有意义、可共享的服务元数据描述对服务发现的重要性^[112]. 其中元数据被理解为描述服务能力的数据, 能够实现对仿真服务的分类、发现、检索和评估, 属于语义层级. MSG-136 小组主要基于元数据实现对建模与仿真服务的发现, 包括三种类型: 1) 发现元数据, 包含描述服务的上下文和内容类型所需的最小信息; 2) 使用元数据, 描述在应用中访问、传输、理解和使用仿真服务所需的信息; 3) 管理元数据, 用于支持管理者对仿真服务进行管理, 包括服务的标识符、版本数据等. 在北约进行的关于容器化仿真服务的实验中^[113], 也验证了基于元数据实现仿真服务信息跨国共享的可行性.

4.2.4 服务组合

为促进建模与仿真资源的高效共享与重用, 大量学者对模型互操作和仿真可组合问题进行了深入研究^[24, 27, 114]. 理论层面, Page 等^[115] 从可集成性、互操作性和可组合性三个维度对仿真互联问题进行了讨论, Tolk 等^[116] 则定义了五个级别的仿真模型互操作, 也被称为概念互操作模型等级 (LCIM). 该概念随后被扩展为七个等级^[117]: 无互操作性、技术互操作、语义互操作、语法互操作、语用互操作、动态互操作和概念互操作. 工程实践层面, Wang 等^[118] 基于 HLA 对 LCIM 的工程实现方法进行研究, 并指出 HLA 关注网络连通性及仿真模型间的语法和语义互操作性, 对于可组合性的研究相对匮乏.

通过将建模与仿真资源封装为服务, 服务组合技术能够有效缓解模型互操作和仿真可组合所面临的挑战. 其中, 服务组合是指基于某种目的对建模

与仿真服务的编排, 可以在不同抽象层级对组合进行描述. 服务组合不仅可以大大降低仿真开发在人力、物力、时间等方面的成本, 也能促进仿真的标准化建设, 提升管理水平. 在进行仿真服务的组合时, 需要考虑功能和非功能两个角度. 从功能角度看, 主要强调组合的正确性 (包含语法、语义、语用等层面). 非功能角度则主要涉及 QoS, 强调在多种选择下如何实现最优的组合. Zhao 等^[119] 基于计算树逻辑 (Computation tree logic, CTL) 规范提出了一种自适应服务组合机制, 以保证边缘计算场景下 IoT 服务的 QoS 动态变化时, 组合服务间的兼容性. 通过对功能需求和组合结构的抽象, 论文将组合问题形式化为 CTL 规范. 同时, 论文构建了 QoS 依赖图来捕捉服务 QoS 的变化, 实现动态 QoS 下的自适应服务组合. 考虑到当前的服务组合算法往往只在组合质量、资源效率等特定方面优于其他算法, Deshpande 等^[120] 则提出了一种组合算法选择框架, 针对不同的应用需求选择最优组合算法.

北约 MSaaS 架构的核心问题便是仿真服务的组合. 为实现仿真的互操作, 不同仿真服务间必须遵循一致的仿真环境协议和仿真数据交换模型. 考虑到仿真服务的异构性, MSaaS 使用了仿真飞地的概念. 不同的仿真飞地可以遵循不同的仿真协议, 如 DIS、HLA、TENA, 同时支持面向服务的多种工业化标准, 包括基于 SOAP 或 RESTful 的网页服务. 每个飞地中的仿真成员可采用不同的数据交互模型, 如 DIS 的 PDU、HLA 的 FOM、TENA 的 LROM 等. 通过保证不同飞地所使用的数据交换模型的独立性、语义规范性和可组合性, MSaaS 可基于协议转换模块实现不同服务的高效组合^[121]. 为验证服务机制的有效性, MSG-136 小组通过集成综合环境服务、路径规划服务和仿真引擎实现了一个简单的坦克机动案例, 结果表明达到了预期目的^[113].

4.2.5 模型可信性

只有在模型足够可信的情况下, 我们才可以相信从模型和仿真中所获取的知识. 对于模型可信性的评价, 一般有两个指标: 逼真度 (Fidelity) 和置信度 (Credibility). 其中逼真度用于反映模型与真实对象的相像程度; 置信度则是根据仿真的具体目的和需求, 评估一个模型可信程度的指标^[122]. 对于不同的仿真需求, 同一模型可能表现出不同的置信度. 尽管当前也有很多关于模型/服务可信性的研究^[114, 123], 但对于组合模型/服务的可信性研究还相对匮乏. 李伟等^[124] 从云仿真系统全生命周期角度出发, 针对仿真资源可信度、仿真子系统可信度和仿真运行环境可靠性评估问题, 建立了云仿真系统的可信度评

估过程模型和云仿真系统可信度评估指标体系. 其中的子系统即可认为是一种组合模型/服务. 考虑到仿真服务的异构性和多样性以及服务组合的复杂性, 仍需对模型的可信性进行更深入的研究.

4.3 运行与管理相关技术

4.3.1 数据分发

一方面, 云数据中心通常为高带宽通信做出了很多优化, 但对于需要频繁交换大量小消息的仿真应用而言, 仍面临较高的网络延迟^[3, 125-126]. 消息捆绑 (Message bundling)^[93] 是解决该问题的可行方案, 通过将多个小消息打包成大的消息块再统一发送, 以部分消息的延迟增加为代价降低总的通信开销. 然而这种方法往往是应用相关的, 技术的有效性受捆绑周期、捆绑消息数量等参数的影响^[127]. 另一方面, 大规模分布式仿真涉及大量实体的交互, 需要设计高效的兴趣管理机制以减少冗余消息的传输, 提高网络效率.

4.3.2 时间同步

时间同步是保证并行与分布仿真结果正确性的重要手段, 典型的方法包括乐观同步和保守同步. 文献 [26] 认为, 有状态服务需要采用恰当的同步机制以最小化服务间状态的不一致性, 而无状态服务则没有这种需求. D' Angelo 等^[128] 和 Malik 等^[129] 对云计算环境下基于乐观算法执行并行离散事件仿真进行了研究. 鉴于计算资源比通信资源更加廉价以及云计算所固有的弹性, 他们认为乐观同步算法相比保守同步是一个更好的选择. 一种可行的提升乐观同步算法性能的方案是通过预测事件到达的时间^[130] 或仿真成员推进的速度^[131] 来控制仿真应用“乐观的程度”, 进而减少事件回滚的数量. 另一方面, Vanmechelen 等^[132] 也对云环境下保守时间同步算法的性能进行了分析. 具体应该采用何种同步策略, 应该根据应用的需求具体分析.

4.3.3 服务部署与调度

不同于常规计算应用, 云环境下仿真服务的调度需要同时考虑节点的计算性能和节点间的网络传输带宽和延迟^[2]. 不同的仿真服务间存在着动态的依赖关系, 不同的节点间也有着相应的网络拓扑. 部分学者将该问题建模为图划分问题, 通过将耦合最为紧密的仿真任务划分到同一子图实现总体通信成本的最小化^[133-135]. 然而, 这类方法常常需要用户提供子图的个数. 有些学者则将该问题建模为整数线性规划问题^[136], 并利用现有的求解器求解, 但常常面临耗时过大的问题. 尽管启发式方法能够提升时效性, 但难以获取最优解^[137]. 还有部分学者采用

集成学习等方法对 PDES 仿真应用的资源需求进行预测^[138-139], 并基于预测结果实现云计算环境下更加精准高效的资源调度. 然而, 现有的预测模型大多只能在特定应用上取得较好效果, 难以应对复杂多样的用户需求. 事实上, 该问题是一个 NP 完全的动态装箱问题, 仍然需要进一步的探索.

4.3.4 负载均衡

并行与分布式仿真的时间推进速度不会超过最慢的仿真成员/逻辑进程 (Logical process, LP) 的推进速度, 需要设计恰当的负载均衡机制实现仿真成员以相似的速度推进. 特别的, 对于采用乐观时间同步算法的仿真应用而言, 不均衡的负载分布会产生大量的事件回滚^[140]. 现有研究中, 有学者基于对负载的预测提前执行负载的动态迁移^[141-142], 也有学者采用工作窃取的方法实现 LP 负载的自适应重分布^[143]. 另一方面, 考虑到事件的回滚可能产生无用的工作, CPU 周期数并不能准确衡量采用乐观同步算法的 LP 的负载. Lindén 等^[144] 对 LP 的负载衡量指标进行研究, 指出可行的指标包括: LP 推进的虚拟时间、墙钟推进时间、已处理或已确认的事件数、回滚事件数等.

4.3.5 数据存储与分析

对于大规模 LVC 仿真而言, 安装在实兵或实装上的传感器会动态生成大量时序数据 (设备状态、元数据和传感器读数等), 需要设计轻量级、分布式的边缘数据库实现对这类数据的高效与灵活管理. 另一方面, 人工智能技术已经成为军事训练效能的倍增器, 如用于构建可敬的虚拟对手、可信的虚拟队友. 边缘侧存储的数据则可以用于实现智能模型的训练与推理. 考虑到边缘侧存在的资源受限、数据样本少、数据异构、数据隐私等问题, 可采用云边协同人工智能技术, 包括增量学习、联邦学习、联合推理等^[106].

4.3.6 安全与隐私

云计算在带来诸多好处的同时, 其传输链路长、资源共享的特征也使得基于云的应用面临着巨大的安全与隐私风险. 为保证云环境下仿真应用的安全与隐私, 很重要的一个侧面即是对仿真数据的管理. Che 等^[145] 对云环境下仿真数据的管理问题进行了研究, 将云仿真的衍生数据分为五种类型, 同时提出了一个三层的云仿真平台数据管理框架. 然而, 这种集中式的管理策略很容易遇到攻击. 一种有前景的方法则是采用基于区块链的去中心化的数据管理范式, 其中, 数据以服务的形式发布, 区块链则作为数据存储的基础设施^[146]. 通过这种方式, 数据拥有者具有对数据的完全控制权, 可以对数据的创建、

收集、存储、分析、追踪等全生命周期进行管理。

4.3.7 可用性与可靠性

云仿真在运行时可能面临多种异常情况: 1) 用户与云中心漫长的网络链路可能存在不稳定或断联, 要求云中心能够正确执行相应仿真状态的保存与回退。2) 某个计算节点宕机, 需要提供高效的仿真成员或服务的迁移机制, 涉及仿真状态的保存、发送、接收与恢复^[2]。对于无状态服务, 可通过创建多个副本提升可用性^[106]。3) 某个仿真成员或服务本身出现异常, 需要管理中心能够迅速执行故障定位, 生成并保存相应出错信息。

5 总结与展望

为了更好地实现建模与仿真的服务化, 支撑随时随地、按需访问的建模仿真能力, 本文对建模与仿真服务化问题进行了概述, 给出了建模与仿真服务化的相关概念与内涵, 从访问、开发和运行与管理三个角度对其发展与演化的过程进行了梳理, 对基于云的建模与仿真服务化的构建原则、基本架构和应用模式进行了分析, 最后探讨了实现建模与仿真服务化的关键支撑技术。通过对现有建模与仿真服务化研究的梳理与总结, 我们得到以下启示:

1) 加强建模与仿真服务化理论体系建设。建模与仿真服务化具有丰富的内涵, 涉及使用、开发、运维等不同类型人员, 实装、模拟器、模型、数据、算法、工具等不同类型仿真资源, 网页技术、软件工程、计算机体系结构等不同技术体系, 具有复杂性、多样性、多领域性等特征, 需要构建从顶层架构到底层实现的完整理论体系, 为后续建模与仿真的服务化建设提供支撑和指导。

2) 强化关键技术攻关。建模与仿真服务化的实现需要突破大量关键技术, 包括智能人机交互、服务发现、服务组合以及安全与隐私保障等。一方面, 通过对上述关键技术问题进行抽象和形式化描述, 可以充分利用其他领域的研究成果, 加快研发进度。例如, 智能人机交互涉及大量语音动作识别等技术。另一方面, 要充分挖掘在建模仿真服务化语境下, 不同技术体现出的独特需求。

3) 关注新兴技术的影响。过去十年, 云计算、大数据、人工智能等技术的发展与应用为建模与仿真领域注入了新的活力。随着新兴技术的不断涌现, 智慧物联网^[147]、数字孪生^[147]、云边协同^[148]等技术也越来越多地被应用到仿真领域。在研究与实现建模与仿真服务化时, 需要充分挖掘这些新技术的潜力。

References

1 Higher Education Act of 1965. As Amended Through P.L. 115-

- 334, Enacted December 20. USA: National Education Association, 2018
- 2 Li Bo-Hu, Chai Xu-Dong, Hou Bao-Cun, Li Tan, Zhang Ya-Bin, Yu Hai-Yan, et al. Networked modeling & simulation platform based on concept of cloud computing —— Cloud simulation platform. *Journal of System Simulation*, 2009, **21**(17): 5292–5299
(李伯虎, 柴旭东, 侯宝存, 李潭, 张雅彬, 余海燕, 等. 一种基于云计算理念的网络化建模与仿真平台 —— “云仿真平台”. *系统仿真学报*, 2009, **21**(17): 5292–5299)
- 3 Fujimoto R, Bock C, Chen W, Page E, Panchal J H. *Research Challenges in Modeling and Simulation for Engineering Complex Systems*. Berlin: Springer International Publishing, 2017.
- 4 Shi Yang, Dong Han-Quan, Lu Ming-Hua. Research on simulation composability and reusability based on SOA. *Journal of System Simulation*, 2014, **26**(7): 1522–1526
(史扬, 董汉权, 陆铭华. 面向服务的可组合可重用仿真技术研究. *系统仿真学报*, 2014, **26**(7): 1522–1526)
- 5 Taylor S J E. Distributed simulation: State-of-the-art and potential for operational research. *European Journal of Operational Research*, 2019, **273**(1): 1–19
- 6 Gustavsson P M, Björk Å, Brax C, Planstedt T. Towards service oriented simulations. In: Proceedings of the Fall Simulation Interoperability Workshop. Orlando, USA: Citeseer, 2004. 219–229
- 7 Li B H, Shi G Q, Lin T Y, Zhang Y X, Chai X D, Zhang L, et al. Smart simulation cloud (simulation cloud 2.0) —— The newly development of simulation cloud. In: Proceedings of the 18th Asian Simulation Conference. Kyoto, Japan: Springer, 2018. 168–185
- 8 Caglar F, Shekhar S, Gokhale A, Basu S, Rafi T, Kinnebrew J, et al. Cloud-hosted simulation-as-a-service for high school STEM education. *Simulation Modelling Practice and Theory*, 2015, **58**: 255–273
- 9 Zehe D, Knoll A, Cai W T, Aydt H. SEMSim cloud service: Large-scale urban systems simulation in the cloud. *Simulation Modelling Practice and Theory*, 2015, **58**: 157–171
- 10 Zhou L J, Gai X P, Lu Y, Wu P, Ren D J, Zhao C J, et al. Research and application of intelligent learning system for power grid all-element simulation based on microservice. *Journal of Physics: Conference Series*, 2021, **1802**: Article No. 042103
- 11 Grimes J G. Department of Defense Net-Centric Services Strategy: Strategy for a Net-Centric, Service Oriented DoD Enterprise, Department of Defense, Chief Information Officer, USA, 2007
- 12 Edgren M G. Cloud-enabled modular services: A framework for cost-effective collaboration. In: Proceedings of the NATO Modelling and Simulation Group Symposium on Transforming Defence through Modelling and Simulation —— Opportunities and Challenges. Arlington, USA: NATO STO, 2012. 1–10
- 13 Hannay J E, van den Berg T. The NATO MSG-136 reference architecture for M&S as a service. In: Proceedings of the NATO Modelling and Simulation Group Symposium on M&S Technologies and Standards for Enabling Alliance Interoperability and Pervasive M&S Applications (STO-MP-MSG-149). USA: NATO Science and Technology Organization, 2017. 1–18
- 14 Siegfried D R. MSG-168 lecture series on modelling and simulation as a service (MSaaS): 3 [Online], available: <https://www.sto.nato.int/publications/STO%20Educational%20Notes/STO-EN-MSG-168/EN-MSG-168-03.pdf>, February 15, 2022
- 15 DOD. Defense modeling and simulation reference architecture, Version 1.0 [Online], available: <https://www.msc0.mil/MS-References/Policy Guidance.aspx>, May 1, 2022
- 16 Cayirci E. Modeling and simulation as a cloud service: A survey. In: Proceedings of the Winter Simulations Conference (WSC). Washington, USA: IEEE, 2013. 389–400
- 17 Ju Ru-Sheng, Yang Mei, Zhong Rong-Hua, Liu Xiao-Cheng, Zhou Yun, Huang Ke-Di. Summary of service oriented modeling and simulation. *Systems Engineering and Electronics*, 2013,

- 35(7): 1539–1546
(鞠儒生, 杨妹, 钟荣华, 刘晓铖, 周云, 黄柯棣. 面向服务的建模与仿真技术综述. 系统工程与电子技术, 2013, **35**(7): 1539–1546)
- 18 Shahin M, Babar M A, Chauhan M A. Architectural design space for modeling and simulation as a service: A review. *Journal of Systems and Software*, 2020, **170**: Article No. 110752
- 19 Kratzke N, Siegfried R. Towards cloud-native simulations — Lessons learned from the front-line of cloud computing. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 2021, **18**(1): 39–58
- 20 Duan Hong, Qiu Xiao-Gang. Networked simulation and its development trend. *Journal of System Simulation*, 2021, **33**(7): 1526–1533
(段红, 邱晓刚. 网络化仿真及其发展趋势. 系统仿真学报, 2021, **33**(7): 1526–1533)
- 21 Mackenzie C M, Laskey K, McCabe F, Brown P F, Metz R. Reference model for service oriented architecture 1.0 [Online], available: <https://docs.oasis-open.org/soa-rm/v1.0/soa-rm.html>, May 1, 2022
- 22 The Open Group. Service-oriented architecture ontology Version 2.0 [Online], available: <https://publications.opengroup.org/c144>, May 1, 2022
- 23 Tolk A. *Engineering Principles of Combat Modeling and Distributed Simulation*. Hoboken: John Wiley & Sons, 2012.
- 24 Hannay J E, van den Berg T, Gallant S, Gupton K. Modeling and simulation as a service infrastructure capabilities for discovery, composition and execution of simulation services. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 2021, **18**(1): 5–28
- 25 Szyperski C. Component technology — What, where, and how? In: Proceedings Of The 25th International Conference On Software Engineering. Portland, Usa: IEEE, 2003. 684–693
- 26 van den Berg T, Siegel B, Cramp A. Containerization of high level architecture-based simulations: A case study. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 2017, **14**(2): 115–138
- 27 Liu Y, Zhang L, Liu Y K, Laili Y J, Zhang W C. Model maturity-based model service composition in cloud environments. *Simulation Modelling Practice and Theory*, 2021, **113**: Article No. 102389
- 28 DODD. Department of defense modeling and simulation (M&S) master plan (DoD 5000.59-P) [Online], available: <https://biotech.law.lsu.edu/blaw/dodd/corres/html/500059p.htm>, May 1, 2022
- 29 Liu Y, Zhang L, Zhang W C, Hu X L. An overview of simulation-oriented model reuse. In: Proceedings of the 16th Asian Simulation Conference and SCS Autumn Simulation Multi-Conference. Beijing, China: Springer, 2016. 48–56
- 30 Song Li-Li. Study on SOA-Based Framework for Modeling and Simulation and the Technologies for Simulation Service Discovery [Ph.D. dissertation], National University of Defense Technology, China, 2009
(宋莉莉. 基于 SOA 的建模与仿真框架及仿真服务发现技术研究 [博士学位论文], 国防科学技术大学, 中国, 2009)
- 31 Zhang L, Wang F, Li F. Cloud-based simulation. *Summer of Simulation*. Cham: Springer, 2019. 97–115
- 32 Fishwick P A. Web-based simulation: Some personal observations. In: Proceedings of the 28th Conference on Winter Simulation. Coronado, USA: IEEE, 1996. 772–779
- 33 Wang S X, Wainer G. Web-based simulation using Cell-DEVS modeling and GIS visualization. *Modeling and Simulation-Based Systems Engineering Handbook*. Boca Raton: CRC Press, 2015. 44
- 34 Wang S X, Wainer G. Modeling and simulation as a service architecture for deploying resources in the cloud. *International Journal of Modeling, Simulation, and Scientific Computing*, 2016, **7**(1): Article No. 1641002
- 35 Page E H, Griffin S P, Rother S L. Providing conceptual framework support for distributed Web-based simulation within the high-level architecture. In: Proceedings of SPIE 3369, Enabling Technology for Simulation Science II. Orlando, USA: SPIE, 1998. 287–292
- 36 Miller J A, Seila A F, Xiang X W. The JSIM web-based simulation environment. *Future Generation Computer Systems*, 2000, **17**(2): 119–133
- 37 Byrne J, Heavey C, Byrne P J. A review of Web-based simulation and supporting tools. *Simulation Modelling Practice and Theory*, 2010, **18**(3): 253–276
- 38 Shi Pei-Chang. Research on Efficient Delivery Techniques for Cloud Services [Ph.D. dissertation], National University of Defense Technology, China, 2012
(史佩昌. 云服务的高效传递技术研究 [博士学位论文], 国防科学技术大学, 中国, 2012)
- 39 Kuljis J, Paul R J. A review of web based simulation: Whither we wander? In: Proceedings of the Winter Simulation Conference Proceedings. Orlando, USA: IEEE, 2000. 1872–1881
- 40 Paul R J, Taylor S J E. What use is model reuse: Is there a crook at the end of the rainbow? In: Proceedings of the Winter Simulation Conference. San Diego, USA: IEEE, 2002. 648–652
- 41 Wiedemann T. Simulation application service providing (SIM-ASP). In: Proceedings of the Winter Simulation Conference (Cat. No.01CH37304). Arlington, USA: IEEE, 2001. 623–628
- 42 Castronova A M, Goodall J L, Elag M M. Models as web services using the open geospatial consortium (OGC) web processing service (WPS) standard. *Environmental Modelling & Software*, 2013, **41**: 72–83
- 43 O'Leary P, Christon M, Jourdain S, Harris C, Berndt M, Bauer A. HPCCloud: A cloud/web-based simulation environment. In: Proceedings of the IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom). Vancouver, Canada: IEEE, 2015. 25–33
- 44 Kim D, Jeong D, Seo Y. Automated composition and execution of web-based simulation systems through knowledge designing and reasoning. *Advanced Engineering Informatics*, 2021, **48**: Article No. 101263
- 45 Fielding R T. *Architectural Styles and the Design of Network-Based Software Architectures*. Irvine: University of California, 2000.
- 46 Tsai W T, Fan C, Chen Y N, Paul R. DDSOS: A dynamic distributed service-oriented simulation framework. In: Proceedings of the 39th Annual Simulation Symposium (ANSS'06). Huntsville, USA: IEEE, 2006. 8–167
- 47 Brebner P. Service-oriented performance modeling the mule enterprise service bus (ESB) loan broker application. In: Proceedings of the 35th Euromicro Conference on Software Engineering and Advanced Applications. Patras, Greece: IEEE, 2009. 404–411
- 48 Smit M, Stroulia E. Simulating service-oriented systems: A survey and the services-aware simulation framework. *IEEE Transactions on Services Computing*, 2013, **6**(4): 443–456
- 49 Arroqui M, Mateos C, Machado C, Zunino A. RESTful Web Services improve the efficiency of data transfer of a whole-farm simulator accessed by Android smartphones. *Computers and Electronics in Agriculture*, 2012, **87**: 14–18
- 50 Al-Zoubi K, Wainer G. RISE: A general simulation interoperability middleware container. *Journal of Parallel and Distributed Computing*, 2013, **73**(5): 580–594
- 51 Morse K L, Tolk A, Pullen J M, Brutzman D. XMSF as an enabler for NATO M&S. In: Proceedings of NATO Modeling and Simulation Group Conference. Koblenz, Germany: NATO Science and Technology Organization, 2004. 1–20
- 52 Zhong Wei, Gong Jian-Xing, Hao Jian-Guo, Huang Ke-Di. Research and analysis of HLA Evolved specification. *Journal of System Simulation*, 2011, **23**(4): 691–696

- (钟蔚, 龚建兴, 郝建国, 黄柯棣. HLA Evolved 规范研究分析. 系统仿真学报, 2011, **23**(4): 691–696)
- 53 Gao Wu-Qi, Kang Feng-Ju, Zhong Lian-Jiong, Fu Yan-Fang. Cloud simulation technology based on HLA Evolved. *Journal of System Simulation*, 2011, **23**(8): 1643–1647
(高武奇, 康凤举, 钟联炯, 傅妍芳. 一种基于 HLA Evolved 的云仿真技术研究. 系统仿真学报, 2011, **23**(8): 1643–1647)
- 54 He Qiang, Hao Jian-Guo, Huang Jian. A simulation service system based on SOA. *Computer Simulation*, 2007, **24**(5): 98–102
(何强, 郝建国, 黄健. 基于 SOA 的仿真服务系统. 计算机仿真, 2007, **24**(5): 98–102)
- 55 Zeldman J. Web 3.0: A list apart [Online], available: <https://alistapart.com/article/web3point0/>, May 1, 2022
- 56 Miller J A, Baramidze G. Simulation and the semantic Web. In: Proceedings of the Winter Simulation Conference. Orlando, USA: IEEE, 2005. 2371–2377
- 57 Zhang T, Liu Y S, Zha Y B. Semantic web based simulation service customization and composition. In: Proceedings of the 7th IEEE International Conference on Computer and Information Technology (CIT 2007). Aizu-Wakamatsu, Japan: IEEE, 2007. 235–240
- 58 Bell D, de Cesare S, Lycett M, Mustafee N, Taylor S J E. Semantic web service architecture for simulation model reuse. In: Proceedings of the 11th IEEE International Symposium on Distributed Simulation and Real-Time Applications (DS-RT'07). Chania, Greece: IEEE, 2007. 129–136
- 59 Erl T. *Service-Oriented Architecture: Concepts, Technology, and Design*. New Jersey: Prentice Hall PTR, 2005.
- 60 Davis P K, Tolk A. Observations on new developments in composability and multi-resolution modeling. In: Proceedings of the Winter Simulation Conference. Washington, USA: IEEE, 2007. 859–870
- 61 Hofmann M A. Criteria for decomposing systems into components in modeling and simulation: Lessons learned with military simulations. *Simulation*, 2004, **80**(7–8): 357–365
- 62 Gustavson P, Chase T, Root L, Crosson K. Moving towards a service-oriented architecture (SOA) for distributed component simulation environments. In: Proceedings of the Spring Simulation Interoperability Workshop. Orlando, USA: IEEE, 2005. 1–8
- 63 Mittal S, Zeigler B P, Martfn J L R, Sahin F, Jamshidi B S M. Modeling and simulation for systems of systems engineering. *System of Systems Engineering: Innovations for the 21st Century*. Hoboken: Wiley, 2009. 101–149
- 64 Hu J P, Huang L P, Cao B, Chang X L. Executable modeling approach to service oriented architecture using SoaML in conjunction with extended DEVSSML. In: Proceedings of the IEEE International Conference on Services Computing. Anchorage, USA: IEEE, 2014. 243–250
- 65 Ramaswamy M V. System Theory Based Modeling and Simulation of SOA-Based Software Systems [Master thesis], Arizona State University, USA, 2008
- 66 Sarjoughian H, Kim S, Ramaswamy M, Yau S. A simulation framework for service-oriented computing systems. In: Proceedings of the Winter Simulation Conference. Miami, USA: IEEE, 2008. 845–853
- 67 Park A J. Master/Worker Parallel Discrete Event Simulation [Ph. D. dissertation], Georgia Institute of Technology, USA, 2008
- 68 Lewis J, Fowler M. Microservices [Online], available: <https://martinfowler.com/articles/microservices.html>, May 1, 2022
- 69 Villamizar M, Garcés O, Castro H, Verano M, Salamanca L, Casallas R, et al. Evaluating the monolithic and the microservice architecture pattern to deploy web applications in the cloud. In: Proceedings of the 10th Computing Colombian Conference (10CCC). Bogota, Colombia: IEEE, 2015. 583–590
- 70 Namiot D, Sneps-Sneppé M. On micro-services architecture. *International Journal of Open Information Technologies*, 2014, **2**(9): 24–27
- 71 Taylor S J E, Anagnostou A, Kiss T, Pattison G, Kite S, Kovacs J, et al. An architecture for an autoscaling cloud-based system for simulation experimentation. In: Proceedings of the Winter Simulation Conference. Gothenburg, Sweden: IEEE, 2018. 4088–4089
- 72 Abubakar N T, Taylor S J E, Anagnostou A. Cloud-based modeling & simulation: Introducing the distributed simulation layer. In: Proceedings of the Winter Simulation Conference. Gothenburg, Sweden: IEEE, 2018. 4218–4219
- 73 Liu Yong-Kui, Zeng Ming, Zhang Lin, Guo Jin-Wei, Yuan Si-Yang, Ping Yao-Yao. Design and development of a simulation system for scheduling in cloud manufacturing based on microservice architecture. *Journal of System Simulation*, 2022, **34**(4): 700–711
(刘永奎, 曾鸣, 张霖, 郭金维, 原思阳, 平垚垚. 基于微服务架构的云制造调度仿真系统设计与开发. 系统仿真学报, 2022, **34**(4): 700–711)
- 74 Keeskemeti G, Marosi A C, Kertesz A. The ENTICE approach to decompose monolithic services into microservices. In: Proceedings of the International Conference on High Performance Computing & Simulation (HPCS). Innsbruck, Austria: IEEE, 2016. 591–596
- 75 Weinman J. Mathematical proof of the inevitability of cloud computing [Online], available: <http://asecib.ase.ro/cc/articole/Inevitability%20of%20Cloud.pdf>, May 1, 2022
- 76 Baldini I, Castro P, Chang K, Cheng P, Fink S, Ishakian V, et al. Serverless computing: Current trends and open problems. *Research Advances in Cloud Computing*. Singapore: Springer, 2017. 1–20
- 77 Kritikos K, Skrzypek P. Simulation-as-a-service with serverless computing. In: Proceedings of the IEEE World Congress on Services (SERVICES). Milan, Italy: IEEE, 2019. 200–205
- 78 Kratzke N, Quint P C, Palme D, Reimers D. Project cloud TRANSIT or to simplify cloud-native application provisioning for SMEs by integrating already available container technologies. In: Proceedings of the European Space Project on Smart Systems, Big Data, Future Internet — Towards Serving the Grand Societal Challenges. Rome, Italy: SciTePress, 2016. 3–26
- 79 Villamizar M, Garcés O, Ochoa L, Castro H, Salamanca L, Verano M, et al. Cost comparison of running web applications in the cloud using monolithic, microservice, and AWS Lambda architectures. *Service Oriented Computing and Applications*, 2017, **11**(2): 233–247
- 80 Hellerstein J M, Faleiro J, Gonzalez J E, Schleier-Smith J, Sreekanti V, Tumanov A, et al. Serverless computing: One step forward, two steps back. arXiv preprint arXiv: 1812.03651, 2018.
- 81 Rycerz K, Bubak M, Malawski M, Słot P. Support for effective and fault tolerant execution of HLA-based applications in the OGSA framework. In: Proceedings of the 4th International Conference on Computational Science. Kraków, Poland: Springer, 2004. 848–855
- 82 Xie Y, Teo Y M, Cai W, Turner S J. Towards grid-wide modeling and simulation. In: Proceedings of the Singapore-MIT Alliance Annual Symposium. Singapore: Singapore-MIT Alliance, 2005. 1–9
- 83 Xie Y, Teo Y M, Cai W, Turner S J. Service provisioning for HLA-based distributed simulation on the grid. In: Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS'05). Monterey, USA: IEEE, 2005. 282–291
- 84 Chen X J, Cai W T, Turner S J, Wang Y. SOAr-DSGrid: Service-oriented architecture for distributed simulation on the grid. In: Proceedings of the 20th Workshop on Principles of Advanced and Distributed Simulation (PADS'06). Singapore: IEEE, 2006. 65–73
- 85 Li B H, Chai X D, Di Y Q, Yu H Y, Du Z H, Peng X Y. Re-

- search on service oriented simulation grid. In: Proceedings of the Autonomous Decentralized Systems. Chengdu, China: IEEE, 2005. 7–14
- 86 Zhang Wei. Research on Key Technologies of Service Grid for Parallel and Distributed Simulations [Ph.D. dissertation], National University of Defense Technology, China, 2009 (张卫. 面向并行分布式仿真的服务网格关键技术研究 [博士学位论文], 国防科学技术大学, 中国, 2009)
- 87 Cai Ying. Research on Key Technologies of Service-Oriented Simulation Supporting Environment [Ph.D. dissertation], National University of Defense Technology, China, 2014 (蔡旻. 面向服务的仿真支持环境关键技术研究 [博士学位论文], 国防科学技术大学, 中国, 2014)
- 88 Risco-Martín J L, Henares K, Mittal S, Almendras L F, Olcoz K. A unified cloud-enabled discrete event parallel and distributed simulation architecture. *Simulation Modelling Practice and Theory*, 2022, **118**: Article No. 102539
- 89 Bordón-Ruiz J, Besada-Portas E, López-Orozco J A. Cloud DEVS-based computation of UAVs trajectories for search and rescue missions. *Journal of Simulation*, 2022, **16**(6): 572–588
- 90 Matlekovc L, Juric F, Schneider-Kamp P. Microservices for autonomous UAV inspection with UAV simulation as a service. *Simulation Modelling Practice and Theory*, 2022, **119**: Article No. 102548
- 91 Grom A, Rheinsmith R, Blount E, Janele J. Joint staff J7 joint training tools for campaign planning. In: Proceedings of the MODSIM World Conference. Virginia Beach, USA: IEEE, 2017. 1–10
- 92 VR360. New British Army programme set to use VR, MR, and cloud software [Online], available: <https://virtualreality-news.net/news/2019/feb/05/new-british-army-programme-set-use-vr-mr-and-cloud-software/>, May 1, 2022
- 93 Fujimoto R M, Malik A W, Park A J. Parallel and distributed simulation in the cloud. *SCS M&S Magazine*, 2010, **3**: 1–10
- 94 Tolk A. Composability challenges for effective cyber physical systems applications in the domain of cloud, edge, and fog computing. *Simulation for Cyber-Physical Systems Engineering*. Cham: Springer, 2020. 25–42
- 95 Stine M. *Migrating to Cloud-native Application Architectures*. Sebastopol: O'Reilly Media, 2015.
- 96 Liu X C, He Q, Qiu X G, Chen B, Huang K E. Cloud-based computer simulation: Towards planting existing simulation software into the cloud. *Simulation Modelling Practice and Theory*, 2012, **26**: 135–150
- 97 Shekhar S, Abdel-Aziz H, Walker M, Caglar F, Gokhale A, Koutsoukos X. A simulation as a service cloud middleware. *Annals of Telecommunications*, 2016, **71**(3–4): 93–108
- 98 Cayirci E, Karapinar H, Ozcakir L. Joint military space operations simulation as a service. In: Proceedings of the Winter Simulation Conference (WSC). Las Vegas, USA: IEEE, 2017. 4129–4140
- 99 Wang Hui-Xia, Chen Yi-Cheng, Tan Lang, Liu Jia-Run. Research on system of system simulation technology based on cloud platform. *Control and Information Technology*, 2018, (6): 100–103 (王会霞, 陈宜成, 谭浪, 柳嘉润. 基于云平台的体系化仿真技术研究. 控制与信息技术, 2018, (6): 100–103)
- 100 Qi He-Ping, Ding Wei, Wang Xue-Wen, Tian Chuan, Hou Hai-Hong. Analysis and discussion about simulation system of integrated joint operational training based on cloud architecture. *Fire Control & Command Control*, 2019, **44**(4): 69–73 (齐和平, 丁玮, 王学文, 田川, 侯海宏. 基于云架构的一体化联合训练仿真体系. 火力与指挥控制, 2019, **44**(4): 69–73)
- 101 Çayirci E, Marincic D. *Computer Assisted Exercises and Training: A Reference Guide*. Hoboken: John Wiley & Sons, 2009.
- 102 Zhang Y H, Qu P, Cihang J, Zheng W M. A cloud gaming system based on user-level virtualization and its resource scheduling. *IEEE Transactions on Parallel and Distributed Systems*, 2015, **27**(5): 1239–1252
- 103 Zhang X, Chen H, Zhao Y C, Ma Z, Xu Y L, Huang H J, et al. Improving cloud gaming experience through mobile edge computing. *IEEE Wireless Communications*, 2019, **26**(4): 178–183
- 104 Bymer M L. DSTS: First immersive virtual training system fielded [Online], available: https://www.army.mil/article/84728/dsts_first_immersive_virtual_training_system_fielded, May 1, 2022
- 105 Li B H, Chai X D, Lin T Y, Yang C, Hou B C, Liu Y, et al. Cyber-physical system engineering oriented intelligent high performance simulation cloud. *Simulation for Cyber-Physical Systems Engineering*. Cham: Springer, 2020. 89–118
- 106 Edge Computing Consortium. Edge computing and cloud computing collaboration white paper 2.0 [Online], available: <http://www.econsortium.org/Lists/show/id/522.html>, May 1, 2022 (边缘计算产业联盟. 边缘计算与云计算协同白皮书 2.0 [Online], available: <http://www.econsortium.org/Lists/show/id/522.html>, 2020)
- 107 Chang S, Hood R, Jin H, Heistand S, Chang J, Cheung S, et al. Evaluating the Suitability of Commercial Clouds for NASA's High Performance Computing Applications: A Trade Study, NAS Technical Report NAS-2018-01, NASA Ames Research Center, USA, 2018
- 108 Du Nan, Tan Ya-Xin, Feng Bin. Servicing method of LVC experiment resources based on object metamodel. *Journal of System Simulation*, 2022, **34**(8): 1834–1846 (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. 系统仿真学报, 2022, **34**(8): 1834–1846)
- 109 Achir M, Abdelli A, Mokdad L, Benothman J. Service discovery and selection in IoT: A survey and a taxonomy. *Journal of Network and Computer Applications*, 2022, **200**: Article No. 103331
- 110 Huang Z, Zhao W. A semantic matching approach addressing multidimensional representations for web service discovery. *Expert Systems With Applications*, 2022, **210**: Article No. 118468
- 111 Al-Sayed M M, Hassan H A, Omara F A. An intelligent cloud service discovery framework. *Future Generation Computer Systems*, 2020, **106**: 438–466
- 112 NATO STO. Modelling and Simulation as a Service, Volume 2: Discovery Service and Metadata, The NATO Science and Technology Organization, USA, 2019
- 113 NATO STO. Modelling and Simulation as a Service, Volume 4: Experimentation Report, The NATO Science and Technology Organization, USA, 2019
- 114 Liu Ying, Zhang Lin, Laili Yuan-Jun. Study on model reuse for complex system simulation. *Scientia Sinica Informationis*, 2018, **48**(7): 743–766 (刘莹, 张霖, 赖李媛君. 复杂系统仿真的模型重用研究. 中国科学: 信息科学, 2018, **48**(7): 743–766)
- 115 Page E H, Briggs R, Tufarolo J A. Toward a family of maturity models for the simulation interconnection problem. In: Proceedings of the Spring Simulation Interoperability Workshop. Los Alamitos, USA: IEEE, 2004. 1–11
- 116 Tolk A, Muguira J A. The levels of conceptual interoperability model (LCIM). In: Proceedings of the Fall Simulation Interoperability Workshop. Orlando, USA: Simulation Interoperability Standards Organization (SISO), 2003. 1–11
- 117 Tolk A, Bair L J, Diallo S Y. Supporting network enabled capability by extending the levels of conceptual interoperability model to an interoperability maturity model. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 2013, **10**(2): 145–160
- 118 Wang W, Tolk A, Wang W. The levels of conceptual interoperability model: Applying systems engineering principles to M&S. In: Proceedings of the Spring Simulation Multiconference. San Diego, USA: IEEE, 2009. 1–9

- 119 Zhao D, Zhou Z B, Hung P C K, Deng S G, Xue X, Gaaloul W. CTL-based adaptive service composition in edge networks. *IEEE Transactions on Services Computing*, 2023, **16**(2): 1051–1065
- 120 Deshpande N, Sharma N, Yu Q, Krutz D E. R-CASS: Using algorithm selection for self-adaptive service oriented systems. In: Proceedings of the IEEE International Conference on Web Services (ICWS). Chicago, USA: IEEE, 2021. 61–72
- 121 NATO STO. Modelling and Simulation as a Service — Rapid Deployment of Interoperable and Credible Simulation Environments, MSG-136, The NATO Science and Technology Organization, USA, 2018
- 122 Zhang Lin, Lu Han. Discussing digital twin from of modeling and simulation. *Journal of System Simulation*, 2021, **33**(5): 995–1007
(张霖, 陆涵. 从建模仿真看数字孪生. 系统仿真学报, 2021, **33**(5): 995–1007)
- 123 Zhu Rui. Research on Key Technologies for Trustworthy Service Composition [Ph.D. dissertation], National University of Defense Technology, China, 2009
(朱锐. 可信服务组合若干关键技术研究 [博士学位论文], 国防科学技术大学, 中国, 2009)
- 124 Li Wei, Zhang Huan, Ma Ping, Yang Ming. Research on credibility assessment of cloud simulation system. *Journal of System Simulation*, 2022, **34**(4): 679–687
(李伟, 张欢, 马萍, 杨明. 云仿真系统可信度评估问题探讨. 系统仿真学报, 2022, **34**(4): 679–687)
- 125 Walker E. Benchmarking Amazon EC2 for high-performance scientific computing. *Logis*, 2008, **33**(5): 18–23
- 126 Jackson K R, Ramakrishnan L, Muriki K, Canon S, Cholia S, Shalf J, et al. Performance analysis of high performance computing applications on the Amazon web services cloud. In: Proceedings of the IEEE Second International Conference on Cloud Computing Technology and Science. Indianapolis, USA: IEEE, 2010. 159–168
- 127 Sianati A, Boukerche A, De Grande R. Bundling communication messages in large scale cloud environments. In: Proceedings of the IEEE Symposium on Computers and Communication (ISCC). Larnaca, Cyprus: IEEE, 2015. 788–795
- 128 D'Angelo G, Marzolla M. New trends in parallel and distributed simulation: From many-cores to cloud computing. *Simulation Modelling Practice and Theory*, 2014, **49**: 320–335
- 129 Malik A, Park A, Fujimoto R. Optimistic synchronization of parallel simulations in cloud computing environments. In: Proceedings of the IEEE International Conference on Cloud Computing. Bangalore, India: IEEE, 2009. 49–56
- 130 Bauer P, Lindén J, Engblom S, Jonsson B. Efficient inter-process synchronization for parallel discrete event simulation on multicores. In: Proceedings of the 3rd ACM SIGSIM Conference on Principles of Advanced Discrete Simulation. London, United Kingdom: ACM, 2015. 183–194
- 131 Li Z X, Cai W T, Turner S J, Li X R, Duong T N B, Goh R S M. Adaptive resource provisioning mechanism in VEEs for improving performance of HLA-based simulations. *ACM Transactions on Modeling and Computer Simulation*, 2015, **26**(1): Article No. 1
- 132 Vanmechelen K, De Munck S, Broeckhove J. Conservative distributed discrete-event simulation on the Amazon EC2 cloud: An evaluation of time synchronization protocol performance and cost efficiency. *Simulation Modelling Practice and Theory*, 2013, **34**: 126–143
- 133 Mofrad M H, Melhem R, Hammoud M. Revolver: Vertex-centric graph partitioning using reinforcement learning. In: Proceedings of the IEEE 11th International Conference on Cloud Computing (CLOUD). San Francisco, USA: IEEE, 2018. 818–821
- 134 Dindokar R, Simmhan Y. Adaptive partition migration for irregular graph algorithms on elastic resources. In: Proceedings of the IEEE 12th International Conference on Cloud Computing (CLOUD). Milan, Italy: IEEE, 2019. 281–290
- 135 Hosseinalipour S, Nayak A, Dai H Y. Power-aware allocation of graph jobs in geo-distributed cloud networks. *IEEE Transactions on Parallel and Distributed Systems*, 2020, **31**(4): 749–765
- 136 Zhou A, Wang S G, Ma X, Yau S S. Towards service composition aware virtual machine migration approach in the cloud. *IEEE Transactions on Services Computing*, 2020, **13**(4): 735–744
- 137 Bao L, Wu C S, Bu X X, Ren N N, Shen M Q. Performance modeling and workflow scheduling of microservice-based applications in clouds. *IEEE Transactions on Parallel and Distributed Systems*, 2019, **30**(9): 2114–2129
- 138 Wang S, Zhu F, Yao Y P, Tang W J, Xiao Y H, Xiong S Q. A computing resources prediction approach based on ensemble learning for complex system simulation in cloud environment. *Simulation Modelling Practice and Theory*, 2021, **107**: Article No. 102202
- 139 Xiao Y H, Yao Y P, Chen K, Tang W J, Zhu F. A simulation task partition method based on cloud computing resource prediction using ensemble learning. *Simulation Modelling Practice and Theory*, 2022, **119**: Article No. 102595
- 140 Mikida E, Kale L. Adaptive methods for irregular parallel discrete event simulation workloads. In: Proceedings of the ACM SIGSIM Conference on Principles of Advanced Discrete Simulation. Rome, Italy: ACM, 2018. 189–200
- 141 Alkharboush R, De Grande R E, Boukerche A. Load prediction in HLA-based distributed simulation using Holt's variants. In: Proceedings of the IEEE/ACM 17th International Symposium on Distributed Simulation and Real Time Applications. Delft, Netherlands: IEEE, 2013. 161–168
- 142 De Grande R E, Boukerche A, Alkharboush R. Time series-oriented load prediction model and migration policies for distributed simulation systems. *IEEE Transactions on Parallel and Distributed Systems*, 2017, **28**(1): 215–229
- 143 Tang W J, Yao Y P, Li T L, Song X, Zhu F. An adaptive persistence and work-stealing combined algorithm for load balancing on parallel discrete event simulation. *ACM Transactions on Modeling and Computer Simulation*, 2020, **30**(2): Article No. 12
- 144 Lindén J, Bauer P, Engblom S, Jonsson B. Fine-grained local dynamic load balancing in PDES. In: Proceedings of the ACM SIGSIM Conference on Principles of Advanced Discrete Simulation. Rome, Italy: ACM, 2018. 201–212
- 145 Che Z S, Zhao C, Laili Y J, Zhang L. Research on the dynamic management of cloud simulation derived data. *International Journal of Modeling, Simulation, and Scientific Computing*, 2017, **8**(3): Article No. 1750026
- 146 Liu X Z, Sun S X, Huang G. Decentralized services computing paradigm for blockchain-based data governance: Programmability, interoperability, and intelligence. *IEEE Transactions on Services Computing*, 2020, **13**(2): 343–355
- 147 Li Bo-Hu, Chai Xu-Dong, Zhang Lin, Qing Du-Zheng, Shi Guo-Qiang, Lin Ting-Yu, et al. New embedded simulation technology for smart internet of things. *Journal of System Simulation*, 2022, **34**(3): 419–441
(李伯虎, 柴旭东, 张霖, 卿杜政, 施国强, 林廷宇, 等. 面向智慧物联网的新型嵌入式仿真技术研究. 系统仿真学报, 2022, **34**(3): 419–441)
- 148 Di Yan-Qiang, Li Ting, Feng Shao-Chong, Liu Qiong-Yao, Liu Jian-Hong, Chen Zhi-Jia, et al. Parallel simulation system of equipment precision maintenance based on cloud-edge-end architecture. *Journal of System Simulation*, 2022, **34**(9): 1909–1919
(邸彦强, 李婷, 冯少冲, 刘琼瑶, 吕建红, 陈志佳, 等. 云边端架构的装备精确维修平行仿真系统. 系统仿真学报, 2022, **34**(9): 1909–1919)



张 淼 国防科技大学系统工程学院讲师. 2022 年获得国防科技大学博士学位. 主要研究方向为云仿真.

E-mail: zhangmiaol5@nudt.edu.cn
(**ZHANG Miao** Lecturer at the College of Systems Engineering, National University of Defense Technology.

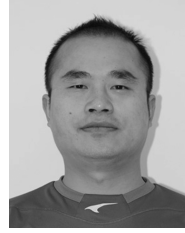
He received his Ph.D. degree from National University of Defense Technology in 2022. His main research interest is cloud simulation.)



许 凯 国防科技大学系统工程学院讲师. 2020 年获得国防科技大学博士学位. 主要研究方向为行为建模, 云仿真.

E-mail: xukai09@nudt.edu.cn
(**XU Kai** Lecturer at the College of Systems Engineering, National University of Defense Technology.

He received his Ph.D. degree from National University of Defense Technology in 2020. His research interest covers behavior modeling and cloud simulation.)



彭 勇 国防科技大学系统工程学院副研究员. 2011 年获得国防科技大学博士学位. 主要研究方向为并行与分布式仿真, 云仿真. 本文通信作者.

E-mail: yongpeng@nudt.edu.cn
(**PENG Yong** Associate researcher at the College of Systems Engineering,

National University of Defense Technology. He received his Ph.D. degree from National University of Defense Technology in 2011. His research interest covers parallel and distributed simulation and cloud simulation. Corresponding author of this paper.)



尹全军 国防科技大学系统工程学院研究员. 2005 年获得国防科技大学博士学位. 主要研究方向为行为建模, 云仿真.

E-mail: yin_quanjun@163.com
(**YIN Quan-Jun** Researcher at the College of Systems Engineering, National University of Defense Technology.

He received his Ph.D. degree from National University of Defense Technology in 2005. His research interest covers behavior modeling and cloud simulation.)