

基于数字孪生的网络仿真推演机制研究

石鸿伟^{1,2}, 倪中阳², 黄韬^{2,3}

(1. 东南大学网络空间安全学院, 江苏 南京 211189; 2. 紫金山实验室未来网络研究中心, 江苏 南京 211111;
3. 北京邮电大学网络与交换技术国家重点实验室, 北京 100876)

摘要: 基于数字孪生的网络仿真为 IP 承载网运维管理提供了有效的决策支持。针对传统网络仿真流程中存在的过程黑盒化、操作不可逆与结果反馈滞后等问题, 提出一种支持分步推演与状态回溯的网络仿真机制。首先, 通过分析现网操作, 设计原子化变更仿真流程, 实现仿真步骤实时跟踪与灵活回溯; 其次, 提出数模双驱动引擎的仿真划分方法, 设计路径还原算法与多维度评估方法, 为原子化变更提供实时评估反馈。实验表明, 该机制在 IP 承载网故障模拟与业务割接场景中, 能够显著提升网络规划与优化的可靠性及决策透明度。

关键词: 分步推演; 状态回溯; 数字孪生网络; 仿真机制; 原子化变更

中图分类号: TP393

文献标志码: A

doi: 10.11959/j.issn.1000-436x.TXXB250643

Research on the network simulation deduction mechanism based on digital twin

Shi Hongwei^{1,2}, Ni Zhongyang², Huang Tao^{2,3}

1. School of Cyber Science and Engineering, Southeast University, Nanjing 211189, China
2. Future Network Research Center, Purple Mountain Laboratories, Nanjing 211111, China

3. State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract: Network simulation based on digital twin provides effective decision support for the operation, maintenance, and management of IP bearer networks. To address issues like process black-boxing, irreversible operations, and delayed result feedback in traditional network simulation processes, a digital twin network (DTN) simulation mechanism supporting stepwise deduction and state backtracking was proposed. Firstly, an atomic change simulation process was designed by analyzing actual network operations to realize real-time tracking and flexible backtracking of simulation steps. Secondly, a simulation division method for the digital-emulation dual-drive engine was proposed, and a path restoration algorithm and multi-dimensional evaluation method were designed to provide real-time evaluation feedback for each atomic change. Experiments show that the proposed mechanism can significantly improve the reliability and decision transparency of network planning and optimization in IP bearer network fault simulation and service cutover scenarios.

Key words: stepwise deduction, state backtracking, digital twin network, simulation mechanism, atomic change

收稿日期: 2026-01-04; 修回日期: 2026-04-02

基金项目: 工业和信息化部工业互联网创新发展工程基金资助项目(No.TC240AAKM-137); 中国联合网络通信有限公司下一代互联网宽带业务应用国家工程研究中心项目(No.2025-STHZ-ZJSSYS-001)

Foundation Items: The Industrial Internet Innovation and Development Project of the Ministry of Industry and Information Technology (No.TC240AAKM-137), The National Engineering Research Center for Next Generation Internet Broadband Service Applications Project of China United Network Communications Group Co., Ltd. (No.2025-STHZ-ZJSSYS-001)

0 引言

作为现代通信体系的重要基础设施,IP承载网承担着海量数据传输、业务连续性保障以及复杂网络功能实现等关键任务。随着网络规模持续扩大、结构日益复杂,网络运行的动态性不断增强,对资源配置优化和运维效率提升提出了更高要求。在此背景下,网络仿真技术逐渐成为支撑网络规划与运维的重要技术手段。近年来,数字孪生网络(digital twin network, DTN)通过在虚拟空间中构建与真实网络相对应的高保真数字映射,实现网络状态的动态同步,为仿真环境提供持续更新的运行数据,使仿真能够更加准确地反映真实网络的运行状况。然而,现有主流仿真软件在复杂网络场景下仍存在局限性,制约了其工程应用效果。

首先是仿真过程黑盒化问题^[1]:无论是数值仿真还是模拟仿真,仿真过程通常是连续执行的,并不记录每个操作步骤及具体内容;这种不记录单步操作的方式,使用户在仿真过程中无法清晰地了解对仿真环境施加了哪些影响,尤其是在复杂网络环境中,难以追踪问题的根源,从而限制了仿真分析的深度和精度。其次是仿真步骤不支持单步回退问题^[2]:仿真开始后,用户若发现错误或需要重新验证某个步骤,只能手工进行逆向操作或重启整个仿真过程,不仅造成计算资源与时间浪费,还可能导致用户错过关键的分析节点,限制了仿真的效率和灵活性。最后是仿真评估结果反馈滞后问题^[3]:虽然可以通过探针或日志对仿真过程进行监控记录,但需在仿真结束后才输出KPI曲线、拓扑快照或告警摘要信息,无法提供步骤级粒度的实时分析,致使用户难以剖析环境细节、定位潜在优化点,削弱了仿真对动态网络环境的实时指导价值。

当前,学术界围绕网络仿真流程的优化展开了一系列研究。文献[4]系统分析了若干仿真软件在仿真过程的一次性执行、缺乏交互性等缺陷,指出需向交互式、增量式仿真演进。文献[5]提出了一种以优化器为中心的仿真闭环策略,根据网络意图,由优化器生成配置并进行仿真验证,实时监测结果后再由优化器迭代调优,实现闭环自适应。文献[6]提出基于检查点的增量仿真框架,通过增量状态保存与局部回滚两大技术手段,降低了并行网络仿真的内存与时间开销。但是,现有研究主要围绕虚实融合闭环、增量变更控制等维度展开,忽

略了仿真环境内的仿真操作细粒度拆解与步骤级实时反馈需求。在基于数字孪生的IP承载网运维中,若仿真过程无法提供实时、精细的操作影响反馈,将削弱仿真技术对动态网络问题的即时分析能力,难以有效指导实际运维决策。

基于此,本文提出一种支持分步推演与状态回溯的仿真机制,以实现仿真过程细粒度操作控制与步骤级实时反馈。基于现网操作设计原子化变更操作与流程,通过单步隔离与快照管理跟踪、回溯步骤,实现操作影响的精准定位与过程可追溯,解决黑盒化与不可逆问题;并提出数模双驱动引擎的范畴划分方法,设计路径还原算法与多维度评估方法,为原子化操作提供实时反馈,解决反馈滞后问题。

本文主要工作如下。

(1) 调研DTN领域研究现状,分析业界主流仿真软件,针对数值仿真与模拟仿真不同技术路线,分别梳理各自通用仿真流程,总结出仿真过程黑盒化、操作不可逆、结果反馈滞后等共性问题。

(2) 针对共性问题,提出支持分步推演与状态回溯的仿真机制,以提高仿真灵活度与透明度;同时为支撑该机制,提出基于数模双驱动仿真引擎的仿真范畴划分方法,设计业务路径还原算法及多维度仿真评估方法,并阐明其支撑与依赖关系。

(3) 基于IP承载网场景,进行网络故障模拟与业务割接场景仿真实验,结果表明该仿真机制能为IP承载网运维提供灵活且透明的决策支撑,显著提升网络规划与优化的可靠性。

1 相关工作

DTN是数字孪生技术在信息通信领域的深化应用,其借助网络仿真软件构建虚拟网络镜像,动态模拟真实网元、协议、流量特征及网络行为全过程。DTN的研究主要由标准组织牵头,由领先运营商和设备商持续实践。国际电信联盟电信标准化部门、因特网工程任务组(Internet Engineering Task Force, IETF)、欧洲电信标准组织(European Telecommunications Standards Institute, ETSI)分别在数字孪生网络总体架构、应用场景、指标系统方面进行了系统性规范与定义^[7-9]。产业层面,AT&T、德国电信等运营商已将DTN技术应用于网络自动化与跨域保障,爱立信、诺基亚等设备商也将DTN能力融入网络编排与管理系统。我国DTN发

展呈现标准与产业协同推进的格局，中国通信标准化协会设立数字孪生网络专项工作组，系统推进相关技术研究；中国移动和中国联通相继发布 DTN 白皮书，提出网络架构与典型应用场景^[10-11]。同时，运营商与设备厂商积极探索 DTN 在智能运维、云网融合等场景中的应用，推动数字孪生网络从概念研究走向工程实践。

网络仿真软件具备网络建模、运行模拟与结果验证等能力，是 DTN 的重要技术支撑。业界主流网络仿真技术路线分为数值仿真与模拟仿真两类：数值仿真基于数学模型与算法构建虚拟网络环境，以离散事件表征流量与协议，用数学模型描述信号传播，虽成本低、可控性强，适合理论研究与大規模网络性能评估，但对真实网络复杂性的反映依赖建模精度与广度；模拟仿真依托虚拟化技术，通过网元镜像构建近实际运行环境，采用真实软件模拟设备与协议行为，虽精度高、实时性强，但需要高性能硬件与网元镜像支持，扩展性受限。

在数值仿真方面，OPNET、OMNeT++、NS-3 等作为代表性工具^[12-14]，仿真流程可概括为一般性流程。仿真环境初始化后，依需求新建或导入网络模型，经节点链路配置、协议栈选择、流量模型定义与仿真参数配置，完成运行前检查并执行仿真，运行中收集数据，按需通过内置或外部分析工具处理，生成可视化图表并评估结果，覆盖从拓扑构建、协议配置到仿真分析的完整闭环。不同软件在模型构建方式、协议实现、扩展机制等环节存在技术差异，但整体遵循从构建、配置到运行、分析与评估的逻辑主线。

在模拟仿真方面，如华为的 eNSP^[15]、新华三的 HCL^[16]，以及 EVE-NG、PNETLab、GNS3 等开源仿真平台^[17-19]，仿真流程同样具有相似性：均以初始化仿真环境为起点，若存在已匹配的场景模板，则从模板中加载场景并按需微调拓扑，若没有匹配模板，则自定义仿真场景；之后启动网元并配置网元参数；接着进行运行状态检查，若未通过则重新配置网元参数，通过后可选择将该场景记录为模板，生成场景模板；最终流程结束。通过这样标准化的步骤，实现了网络场景的模拟构建与验证。

上述数值仿真与模拟仿真流程存在 3 个共性问题，制约了仿真过程的灵活性与决策的可靠性：一是仿真过程呈现黑盒化特征，仿真任务缺乏单步操作

及中间状态记录，复杂场景下难以追溯问题来源；二是仿真流程不支持操作级别的回溯机制，错误或重复验证时需依赖人工逆向操作或重启仿真，增加了计算与时间开销；三是仿真结果的反馈机制滞后，需在仿真任务完整结束后才能输出统计结果与分析信息，缺乏实时的步骤级网络状态评估。

相较于仿真流程与执行机制层面的研究，现有学术界关于 DTN 的研究更多聚焦于模型能力与系统智能性的提升。具体而言，相关工作主要集中在 4 个方面：一是高保真多尺度建模，融合物理定律与数据驱动方法，实现设备级至业务级的精准刻画^[20-21]；二是实时同步与数据治理，设计低开销数据采集传输、异构融合及模型快速更新算法^[22-23]；三是虚实映射智能决策，构建闭环框架支持假设分析，生成路由优化等策略并反馈物理网络执行^[24]；四是可信评估与大規模并行仿真，开展模型不确定性量化验证，支撑超大型网络实时孪生运行^[25-26]。

总体来说，现有研究在仿真流程细粒度可控性、执行机制可逆性及评估反馈实时性方面缺乏系统性设计。针对上述不足，本文提出一种基于数字孪生的网络仿真推演机制。首先，引入原子化操作驱动的分步推演流程，将传统整体式仿真拆解为可独立执行与回溯的最小单元，实现过程级精细控制与可追溯；其次，构建数模双驱动仿真引擎，通过模拟仿真与数值仿真的分层协同建模，在保证协议行为真实性的同时提升大規模业务建模效率；最后，设计业务路径还原算法与多维度评估方法，实现由结果后验分析向步骤级实时评估的转变，对每一步变更的业务影响进行即时、量化分析。从机制设计层面提升了仿真过程的可控性、可逆性与评估实时性，增强其在复杂网络运维场景中的工程适用性。

2 基于数字孪生的网络仿真推演架构

为实现基于数字孪生的网络仿真推演能力，本文构建了一种面向 IP 承载网络的网络仿真推演总体架构，如图 1 所示。该架构由实体网络层、仿真引擎层、仿真推演层和应用层 4 个层次组成。其中，实体网络层提供真实网络运行数据，为虚拟网络环境的构建提供基础支撑；仿真引擎层通过模拟仿真引擎与数值仿真引擎的协同计算，构建与实际网络相对应的数字孪生网络环境；仿真推演层在此基础上实现网络状态变化的推演计算，并对网络运

行状态进行分析评估；应用层则基于仿真推演结果，为网络故障模拟、业务割接和网络规划等典型运维场景提供决策支持。

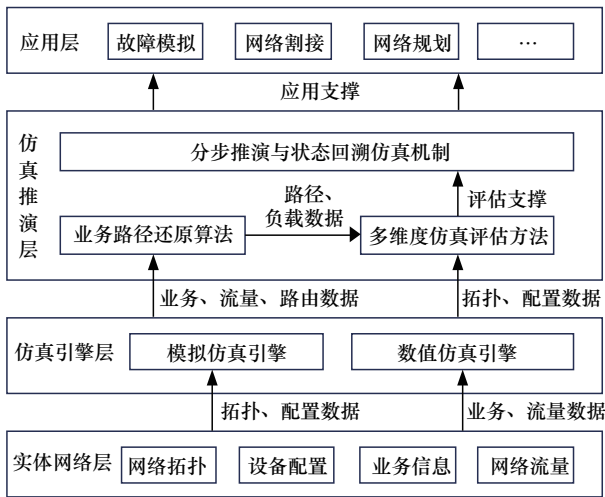


图1 网络仿真推演总体架构

从图1可以看出，仿真推演层是实现网络仿真推演能力的核心，通过构建分步推演与状态回溯机制，实现对网络状态演化过程的推演与分析。该机制的实现依赖数模双驱动网络仿真引擎、业务路径还原算法，以及多维度仿真评估方法的协同支撑。其中，数模双驱动网络仿真引擎用于构建高保真的虚拟网络环境，提供拓扑、业务和流量等建模能力；业务路径还原算法基于引擎生成的业务、流量及路由等数据，对业务转发路径进行还原并计算链路流量负载；多维度仿真评估方法对推演过程中产生的网络状态数据进行实时解析，从路径变化、业务连通性和链路负载等多个维度对网络状态变化进行综合评估。三者数据与功能层面逐级衔接，共同构成支撑分步推演与状态回溯仿真机制的关键技术体系。

2.1 分步推演与状态回溯仿真机制

针对传统网络仿真流程中存在的操作不可逆、过程黑盒化及结果反馈滞后问题，提出一种支持分步推演与状态回溯的仿真机制。该机制以原子化变更操作为核心，通过状态机模型与快照管理策略，实现仿真步骤的实时跟踪与灵活回溯。

2.1.1 原子化与机制设计

结合实际IP承载网规建维优流程的具体实施逻辑，将网络运维中的节点管控、链路调整、配置下发、业务部署等操作，拆解为最小粒度的独立执行单元，形成原子化仿真变更步骤，每个步骤均内

置明确的正向执行逻辑与逆向回溯机制。

原子化操作设计对标实际IP承载网运维中的关键实施步骤，重点刻画对转发与流量产生影响的行为，原子化操作分类如表1所示。节点类操作用于网元状态管理，包括新增、删除、启动与停止节点，模拟设备部署与上下线过程；链路类操作面向网络连接调整，涵盖链路增删、故障与恢复及质量参数调整，复现带宽、时延与丢包率变化场景；配置类操作针对网元配置管理，支持配置导入与CLI下发，模拟批量配置与逐行调试过程；业务类操作围绕业务部署，包括终端、接入点、通信对与业务流量配置，实现对业务粒度的仿真支撑。

操作类型	原子化操作
节点类	新增节点、删除节点、启动节点、停止节点
链路类	新增链路、删除链路、链路故障、链路恢复、链路QoS配置
配置类	配置导入、CLI命令行配置
业务类	终端配置、接入点配置、通信对配置、业务流量配置

原子化变更操作有限状态机模型如图2所示，执行过程划分为初始、执行、成功与失败4种状态。操作触发后由初始状态进入执行状态，系统在执行过程中进行依赖检查与冲突检测：依赖检查用于判断前置操作是否完成，如新增链路需已存在相关节点；冲突检测用于识别与当前或历史操作的资源冲突，如删除节点时该节点仍被业务流量使用。若任一条件不满足，操作进入失败状态并触发回滚；执行成功则进入成功状态，并基于写时复制(copy on write, CoW)差分快照技术记录数据变化，更新仿真环境状态。

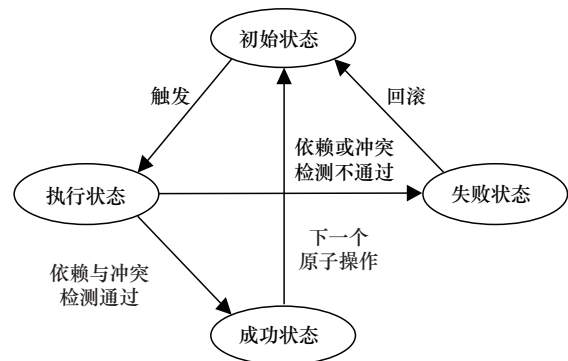


图2 原子化变更操作有限状态机模型

CoW 机制在不显著增加存储开销的情况下支持任意步骤恢复。执行原子化操作时,系统仅在数据被写入时才复制相应数据块,而非提前复制整体数据,从而避免大量冗余复制。相较传统全量快照,CoW 差分快照只记录发生变化的数据,如拓扑数据变化时,仅保存其差异部分而非完整环境状态,因此能够以较低存储成本实现精确回溯。

支持分步推演与状态回溯的仿真流程如图 3 所示。首先,完成仿真场景的搭建与环境初始化。随后,设定仿真推演基线,采集并存储基线数据,进入单步推演状态;在该状态下执行原子化变更操作,若需进行网络评估,可进一步选择执行单步评估或全量评估;无论是否执行评估,均可选择对原子化变更操作进行状态回溯,或继续推进推演进程;推演结束后,系统生成最终推演评估报告并清理过程数据。最后,根据实际需求判断是否销毁仿真场景或对场景进行重新配置后再次进入推演流程。

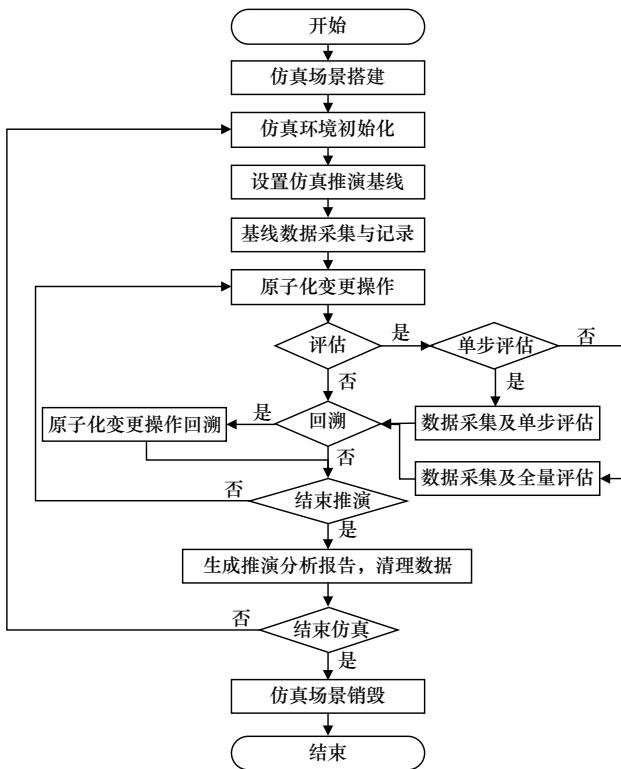


图3 支持分步推演与状态回溯的仿真流程

2.1.2 形式化模型

为增强本文分步推演与状态回溯仿真机制的理论严谨性,本节对仿真过程中的状态空间、原子操作、推演序列及回溯机制进行形式化建模,并在此

基础上分析该机制的关键性质。

设仿真系统在任意时刻的全局状态为:

$$S = (T, C, B, F) \quad (1)$$

其中, T 为拓扑状态,表示节点集合、链路集合及其运行状态; C 为配置状态,表示网元、协议、标签及策略配置; B 为业务状态,表示 VPN 实例、终端 IP 及通信对关系; F 为流量状态,表示链路流量、链路利用率及业务流量统计。定义仿真状态空间为:

$$\mathcal{S} = \{S | S = (T, C, B, F)\} \quad (2)$$

其中,由于网络节点数量、链路数量及配置集合均为有限集合,因此 \mathcal{S} 为有限离散状态空间。仿真推演过程可视为状态空间上的演化过程。设原子操作集合为:

$$\mathcal{A} = \{a_1, a_2, \dots, a_n\} \quad (3)$$

每个原子操作可抽象为定义在状态空间上的状态变换函数,即:

$$a: \mathcal{S} \rightarrow \mathcal{S} \quad (4)$$

对任意状态 $S \in \mathcal{S}$, 执行操作 a 后得到新状态为:

$$S' = a(S) \quad (5)$$

原子操作满足 3 个方面的特性:一是不可分割性,操作在系统内部执行时不可进一步拆分;二是局部性,存在最小变更子集 $\Delta \subseteq S$ 使 $a(S) = S \oplus \Delta$,即原子操作仅影响状态的局部变量;三是可逆性,存在逆操作 a^{-1} ,满足 $a^{-1}(a(S)) = S$,保证单步操作具备逻辑可回退能力,是实现状态回溯机制的基础。

在实际仿真过程中,多个原子操作按时间顺序组合形成推演过程。设原子操作集合为 \mathcal{A} ,则长度为 k 的推演序列定义为:

$$\Pi = (a_{i_1}, a_{i_2}, \dots, a_{i_k}) \quad (6)$$

其对初始状态 S_0 的作用可表示为函数复合:

$$S_k = a_{i_1} \circ a_{i_2} \circ \dots \circ a_{i_k}(S_0) \quad (7)$$

其中, S_k 为第 k 步推演后的状态。分步推演本质上是有限状态空间上的函数复合过程。每一步操作均对应一个确定的状态映射,因此推演过程可视为状态转移链:

$$S_0 \rightarrow S_1 \rightarrow \dots \rightarrow S_k \quad (8)$$

该抽象说明,分步推演并非一次性整体计算,而是由可观测、可验证的单步状态转移组成。

基于原子操作的可逆性与差分快照机制,给定推演序列 Π 与当前状态 S_k ,回溯至第 j 步状态定义为:

$$\text{Rollback}(j):S_k \rightarrow S_j \quad (9)$$

若基于逆操作链实现,则有:

$$S_j = a_{i_{j+1}}^{-1} \circ \dots \circ a_{i_k}^{-1}(S_k) \quad (10)$$

若基于快照机制实现,则有:

$$S_j = \text{Snapshot}(j) \quad (11)$$

其中, $\text{Snapshot}(j)$ 表示第 j 步记录的差分快照状态,状态回溯在数学上等价于逆函数复合或状态存档恢复,两种方式在逻辑上保持一致。

针对上述形式化模型,从性质上分析分步推演与状态回溯机制。

首先具有封闭性,针对任意 $S \in \mathcal{S}$ 与任意 $a \in \mathcal{A}$, 有:

$$a(S) \in \mathcal{S} \quad (12)$$

即原子操作不会生成非法状态,状态空间对操作集合封闭,保证推演过程始终处于合法仿真域内。

其次具有可逆性,对于任意推演序列 Π ,存在逆序列 Π^{-1} ,即:

$$\Pi^{-1}(\Pi(S_0)) = S_0 \quad (13)$$

该性质保证仿真系统可回溯至任意历史状态,实现操作级可逆推演。

最后具有确定性,若给定相同初始状态 S_0 与相同推演序列 Π ,则:

$$\Pi(S_0) = \Pi(S_0'), S_0 = S_0' \quad (14)$$

即推演结果唯一,系统状态演化不依赖随机扰动,从而保证仿真分析的可重复性与结果一致性。

2.2 仿真机制关键支撑技术

2.2.1 数模双驱动网络仿真引擎

数模双驱动是融合数值仿真与模拟仿真两大技术路线优势的混合仿真技术体系^[27]。其技术融合的核心理念源于IP承载网核心层的架构特征与转发逻辑差异,该网络具备高速传输与标签交换两大特点,标签交换技术涵盖MPLS LDP与MPLS SR等,在转发机制上,以外层标签作为寻址依据,实现与内层业务IP前缀的分离。

为了更清晰地阐述数模双驱动在仿真内容上的

划分,引入分层抽象模型。该模型将承载网分为Underlay层 (U) 和Overlay层 (O)。网络仿真的标注函数为:

$$\begin{aligned} \varphi:U &\rightarrow \text{模拟仿真} \\ \varphi:O &\rightarrow \text{数值仿真} \end{aligned} \quad (15)$$

Underlay层负责拓扑构建、IGP协议运行及标签生成,作为网络基础架构,其配置相对固定、规模有限,并直接决定业务转发的底层路径。该层对仿真的物理细节还原度与协议行为准确性要求较高,因而适用于模拟仿真。Overlay层则承载业务前缀、VPN实例及业务流量等动态信息,其规模可随业务需求扩展,且业务流量特征不改变底层转发路径。该层更强调建模灵活性与大规模扩展能力,因此更适合采用数值仿真方法。整个仿真环境可表示为:

$$E = \varphi(U) \oplus \varphi(O) \quad (16)$$

其中, \oplus 表示跨层绑定关系,表现为Underlay层与Overlay层的协同联动。

数模双驱动仿真遵循模拟仿真协议转发逻辑和数值仿真扩展网络业务规模的协同原则,IP承载网数值仿真与模拟仿真协同工作逻辑划分与映射关系如图4所示。在该协同体系中,两类仿真引擎并非简单地并行运行,而是通过分层建模、协同映射、事件驱动机制实现紧耦合协作。

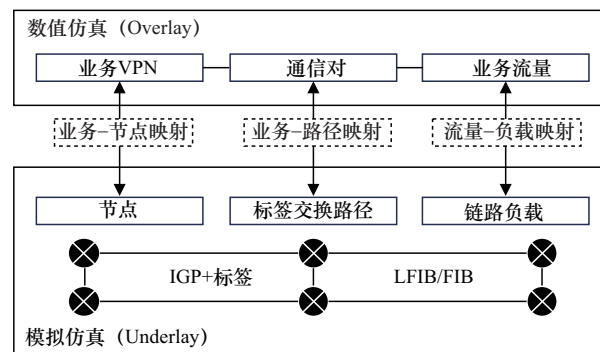


图4 IP承载网数值仿真与模拟仿真协同工作逻辑划分与映射关系

在分层建模层面,模拟仿真引擎与数值仿真引擎分别面向Underlay与Overlay进行解耦建模。模拟仿真引擎负责Underlay层的高保真建模计算,基于真实网元镜像复现设备配置、IGP路由计算、标签分发及网络收敛过程,生成拓扑结构、路由表及转发表等底层转发状态;数值仿真引擎则面向Overlay层进行抽象建模,构建VPN实例、业务通

信对及流量模型,刻画业务之间的逻辑关系与流量分布特征。二者分别从网络转发语义与业务表达语义两个维度对仿真环境进行建模。

协同映射层面,通过构建跨层绑定关系实现 Overlay 与 Underlay 的语义对齐。数值仿真引擎以模拟仿真生成的转发状态为约束,将业务端点映射至具体网络节点,将业务路径映射至底层转发路径,使业务流量的传播严格遵循 Underlay 的转发逻辑,业务流量基于链路叠加后形成链路负载,保证业务推演结果与真实网络行为的一致性。同时,该映射关系具备稳定性与可复用性,在 Underlay 状态不变的情况下,可支持大规模业务的快速加载与复用计算。

联动执行机制上,采用事件驱动的协同更新策略,当 Underlay 层发生原子化变更,如链路故障、节点上下线或配置变更时,由模拟仿真引擎首先触发协议重收敛并更新转发状态,随后将变化以事件形式通知数值仿真引擎,驱动其对受影响业务的路径与流量进行增量重计算;而当变更仅发生在 Overlay 层,如业务增删或流量调整,且不影响底层转发结构时,则仅在数值仿真引擎内部完成计算,无需触发模拟仿真,从而避免重复协议计算的开销。

与传统主要依赖模拟仿真或数值仿真的网络仿真方法相比,本文数模双驱动网络仿真引擎通过构建模拟仿真与数值仿真的协同分工机制,实现了网络状态计算能力与业务建模能力的有机融合。两类仿真方式协同驱动,在保证协议行为真实性的同时,提高了大规模业务场景的建模效率,从而在仿真精度与计算性能之间取得较好的平衡。

2.2.2 业务路径还原算法

业务路径还原的核心原理是基于 IP 承载网核心层 Underlay 决定转发、Overlay 承载业务的架构特性,通过解析标签交换路径的生成结果,反向推导源与目的业务接入点之间的实际转发路径。在 Underlay 层面,核心网网元通过内部网关协议实现域内互联,并依赖专用标签协议或 IGP 扩散标签信息,从而生成公网转发信息库及标签转发表。在 Overlay 层面,边缘 PE 设备通过 VPN 配置与路由协议生成本地业务前缀,并借助 BGP 将其通告至对等体,完成路由迭代与收敛过程。

业务数据包在转发过程中可抽象为 3 层结构:

有效载荷 (payload)、Overlay 头部和 Underlay 头部。其中, payload 承载实际业务数据,与转发逻辑无关; Overlay 头部标识业务 IP 通信对,仅在路径的首末跳设备进行处理; Underlay 头部作为外层标签,是转发的核心依据,路径中的设备通过解析该标签以匹配转发表,确定出接口。特别地,在路由收敛后,固定源与目的 PE 之间的 Underlay 标签交换路径具有确定性,不受 Overlay 层业务通信对数量的影响。

在业务路径还原过程中,所需的拓扑、路由及业务数据由数值仿真与模拟仿真构成的双驱动引擎联合提供。模拟仿真依托虚拟化技术构建与真实网络一致的 Underlay 协议运行环境,同步输出标签转发表、公网 FIB 表等关键底层转发数据;数值仿真则通过参数化建模,明确 Overlay 业务与 Underlay 转发路径之间的关联规则,确保在路径还原中能够精确匹配业务与底层转发间的对应关系。

在路径计算方面,基于可满足性模理论 (satisfiability modulo theories, SMT) 框架^[28]提供了一种将 PE 可达路径转化为可解约束模型的方法。将网络抽象为有向图 $G(V,E)$,其中, V 为节点集合, E 为有向链路集合,为了求解网络路径,采用拓扑、标签转发表集合 $\{LFIB(i)\}$ 与 Loopback 地址映射 $\{\text{loopback}(i)\}$ 作为输入参数来求解网络路径。Loopback 地址为每个设备提供一个唯一的标识符,确保源和目的业务端点能够稳定地映射到具体的节点 PE_s, PE_d ,在路径计算中,定义路径各跳节点、标签与边选择变量分别为 $x_k, l_k, y_k(e)$,利用 SMT 的约束求解能力,在标签转发信息库 (label forwarding information base, LFIB) 规则的引导下,构建一个精确的路径推理逻辑,实现网络路径的计算与还原。

输入集合定义了路径推理的基础数据,包括拓扑、LFIB 及 Loopback 地址,使 SMT 求解器能在已知的确定性数据空间内进行符号推导与优化。输入参数为:

$$G(V,E), \{LFIB(i)|i \in V\}, \{\text{loopback}(i)|i \in V\} \quad (17)$$

利用 Loopback 地址稳定性建立与 PE 的唯一映射,避免因物理端口变更导致的节点不一致问题,确保路径推理可靠性。Loopback 与节点映射关系为:

$$PE_s = \text{nodeof}(LB_s), PE_d = \text{nodeof}(LB_d) \quad (18)$$

路径求解从源PE启动并以目的PE终止, 限定求解器只在合法的端到端业务路径上进行求解, 避免游离解出现, 起止约束为:

$$x_0 = PE_s, x_k = PE_d \quad (19)$$

根据实际网络中的转发语义, 由入标签决定下一跳与出标签, 标签递推与转发约束为:

$$y_k(u \rightarrow v) = 1 \Rightarrow (l_{k+1} = \text{LFIB}(u)[l_k].\text{outlabel} \wedge x_{k+1} = v) \quad (20)$$

在实际转发过程中, 结合ECMP, 每一跳可能存在一个到多个出接口, 链路约束为:

$$\sum_{e \in E} y_k(e) \geq 1 \quad (21)$$

无环限制避免节点重复访问, 从而防止求解器生成死循环路径或不收敛的标签序列, 保证推导的路径满足网络的基本可达性原则, 无环约束为:

$$x_i \neq x_j, \forall_i \neq j \quad (22)$$

通过最小化代价函数, 引导求解器选择实际网络倾向的路径, 最优目标为:

$$\min \sum_k \text{cost}(x_k \rightarrow x_{(k+1)}) \quad (23)$$

业务路径还原算法以模拟仿真引擎输出的拓扑结构、LFIB及Loopback地址映射为输入, 这些数据在Underlay协议收敛完成后具有确定性和稳定性, 因此路径推理问题被限定在一个有限且确定的解空间内。在给定源PE与目的PE的条件下, SMT求解器仅在满足实际转发约束的路径集合中进行推理, 避免了传统基于全局搜索的路径枚举问题, 从而保证了路径还原结果的确定性与可解释性。

计算复杂度方面, 路径还原算法的求解规模主要受网络节点规模、链路数量及标签转发表规模的影响。由于SMT约束条件直接来源于LFIB转发规则, 路径推理过程仅在实际可达路径空间内展开, 其复杂度与路径跳数近似呈线性关系。Overlay层业务数量的增加不会引入新的转发约束, 仅复用已有的Underlay路径, 因此业务规模扩展对单次路径求解复杂度的影响有限。

传统图论算法主要解决最短路径问题, 难以处理IP承载网络中复杂的协议约束, 如MPLS标签交换、分层隧道封装、SR节点标签等。与传统的Dijkstra等图论算法相比, SMT算法则具备理论优势, 在多约束的表达能力方面, 能将路径还原转化为多

维度约束满足问题, 不仅考虑拓扑连通性, 还可将标签递推规则和ECMP负载分担建模为谓词逻辑; 在符号推导与确定性方面, 求解器通过符号推导在LFIB规则引导下寻找合法解空间, 这比单纯的路径枚举更高效, 且能保证还原结果与真实协议行为的高度一致性; 在业务逻辑解耦方面, 其复杂度与路径跳数呈线性关系, 且与Overlay业务规模解耦, 适合处理大规模网络中的复杂逻辑推演。路径还原方法对比如表2所示。

表2 路径还原方法对比

方法	适用问题类型	约束表达能力	复杂度特性	适用场景
Dijkstra	最短路径	仅数值权重	多项式	单一权值网络
SMT	约束满足	逻辑+数值+映射	NP完全	多策略复杂网络

本文基于SMT的业务路径还原算法如算法1所示。

算法1 基于SMT的业务路径还原算法

输入 网络拓扑 $G(V, E)$ 、LFIB集合 $\{\text{LFIB}(i)\}$ 、Loopback映射 $\{\text{loopback}(i)\}$ 、源/目端点 LB_s/LB_d 、业务约束集合 C

输出 业务路径 P

- 1) 通过 LB_s/LB_d 映射, 获取源节点 s 与目的节点 d
- 2) 初始化SMT求解器Solver
- 3) 定义链路选择变量集合 $\{x_e | e \in E\}$, 其中 $x_e \in \{0, 1\}$
- 4) 构建源节点约束: $\sum_{e \in \text{Out}(s)} x_e = 1$
- 5) 构建目的节点约束: $\sum_{e \in \text{In}(d)} x_e = 1$
- 6) for 每个节点 $v \in V \setminus \{s, d\}$ do
- 7) 构建流守恒约束: $\sum_{e \in \text{In}(v)} x_e = \sum_{e \in \text{Out}(v)} x_e$
- 8) end for
- 9) for 每个节点 $v \in V$ do
- 10) 构建无环约束: $\sum_{e \in \text{In}(v)} x_e \leq 1$
- 11) end for
- 12) 将业务约束集合 C 编码为逻辑表达式
- 13) Solver.add(拓扑约束 \wedge 无环约束 \wedge 业务约束)
- 14) 基于式(23)构建目标函数并进行优化求解

- 15) result \leftarrow Solver.check()
- 16) if result == SAT then
- 17) $M \leftarrow$ Solver.model()
- 18) $E' \leftarrow \{e \in E | M(x_e) = 1\}$
- 19) 根据拓扑顺序由 s 到 d 重构路径 P
- 20) else
- 21) $P \leftarrow \emptyset$
- 22) end if
- 23) 返回 P

与传统网络仿真侧重网络拓扑状态或路由收敛结果分析不同, 本文业务路径还原算法利用仿真引擎生成的网络转发表信息及拓扑关系, 对业务在网络中的实际转发路径进行推导与重建。该算法根据业务通信关系逐跳匹配网络转发路径, 并进一步计算各链路上的业务流量负载情况, 从而实现对网络运行状态的细粒度分析, 为网络变更影响评估提供基础数据支撑。

2.2.3 多维度仿真评估方法

基于 IP 承载网真实运维场景中故障、网元割接与业务部署等变更产生的数据特征, 结合双仿真引擎与业务路径还原能力, 构建用于仿真推演分析的数据评估模型。模型包含拓扑、配置、业务与流量 4 类核心数据: 拓扑数据描述网络结构及节点、链路状态; 配置数据为虚拟网元的接口、协议、标签与策略等命令行信息; 业务数据包括 VPN、终端 IP、通信对及转发路径; 流量数据反映背景流量、业务流量、链路负载与利用率。4 类数据构成从网络结构、配置、业务到性能的多维观测体系, 在仿真过程中按需采集并统计分析, 为单步推演效果评估提供量化决策依据。

仿真推演基线建立时, 系统记录场景的拓扑、配置、业务与流量 4 类基线数据。后续单步推演过程中, 可选择单步分析或全量分析: 当采用单步分析时, 仅记录该变更涉及的数据类型, 如拓扑变更则对比变更前后的拓扑并得到对应变化结果; 当采

用全量分析时, 则重新采集 4 类完整数据, 分析结果涵盖拓扑结构、配置调整、业务路径变化及链路负载波动等影响。每一步采用的分析方式由用户根据操作重要性和影响范围自主选择。

在明确多维评估流程与数据采集方式的基础上, 为进一步提高评估结果的可量化与可对比性, 本文选取具有代表性的核心指标构建路径还原效果评估。表 3 从业务健康度与网络承载质量两个关键维度出发, 分别给出了业务可达率与链路利用率的计算公式、物理含义及阈值判定依据。其中, 业务可达率用于衡量推演结果对业务连续性的保障能力, 链路利用率则反映路径重构后网络资源占用与潜在拥塞风险。通过对上述指标的联合分析, 可实现对路径还原方法有效性与稳定性的综合评估, 为后续算法优化与策略调整提供定量支撑。

多维度仿真评估方法从路由变化、业务可达性、链路流量及网络性能等多个维度对仿真结果进行综合分析, 可在每一步推演操作后对网络状态变化进行实时评估, 并通过对比推演前后的网络运行状态, 分析网络变更可能带来的业务影响, 从而为网络规划与运维决策提供更加全面的分析依据。

3 实验验证

本实验依托紫金山实验室已有科研成果^[29], 在数模双驱动网络仿真原型系统上扩展实现分步推演与状态回溯能力。为验证其在 IP 承载网运维中的有效性, 选取网络故障模拟与网络割接预演两类场景开展实验。网络故障模拟以链路中断为典型故障, 通过故障注入复现受影响的业务与转发路径; 网络割接预演面向拓扑与配置变更, 基于原子化操作编排割接步骤, 并评估关键步骤的应用效果。

实验环境包括两台硬件服务器, 规格均为 128 核 CPU、256 GB 内存与 2 TB 磁盘, 其中一台部署 KVM

表 3 网络质量量化评估方法

评估维度	关键性能指标	量化计算公式/测量方法	物理含义	阈值说明
业务健康度	业务可达率 R_c	$R_c = \frac{\text{Count}(\text{Connected}_{\text{pairs}})}{\text{Count}(\text{Total}_{\text{pairs}})} \times 100\%$	业务连通性	正常值为 1
网络承载质量	链路利用率 U_l	$U_l = \frac{\sum \text{Traffic}_{\text{business}} + \text{Traffic}_{\text{bg}}}{\text{Capacity}_l} \times 100\%$	链路负载情况	超过 80% 告警



图5 IP承载网仿真实验环境系统

虚拟化环境, qemu版本为4.2.1, 作为模拟仿真引擎; 另一台部署Kubernetes (K8s) 微服务环境, 运行数值仿真引擎及各类系统微服务, K8s版本为1.19.9. 两个节点共同组成一套网络仿真实验平台。

仿真实验网络以IP承载网为原型搭建, 开展仿真机制的验证, IP承载网仿真实验环境系统如图5所示, 共部署24个节点, 包括8个pe节点、8个p节点及8个数据中心 (data center, DC) 业务接入终端, 采用基于MPLS SR的业务承载方式, 以标签进行报文转发。其中, pe与p节点通过新华三VSR1000网元镜像在模拟仿真引擎中运行; 8个DC终端及其互通业务由数值仿真引擎以数值化方式模拟。

3.1 故障模拟场景

针对组网中的8个接入DC, 每两个DC间均开通多条业务, 同一DC采用VPC区分不同业务, 共100条业务。业务配置完成后, 仿真平台根据仿真引擎提供的拓扑、路由、业务等数据, 还原出每条业务的流量转发路径, 以单条链路为粒度, 统计出每条链路所承载的业务数量; 在链路中断后, 评估故障影响的业务数量, 以及业务转发路径变化导致的链路承载业务量的变化。故障模拟场景仿真推演步骤编排如图6所示, 仿真环境初始化完成后设置推演基线, 通过step1模拟链路故障, 由step2撤销链路故障, 在step1进行一次全量评估, 分析网络的具体影响。

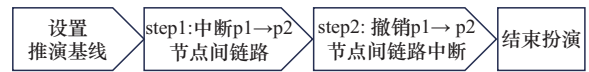


图6 故障模拟场景仿真推演步骤

具体地, 推演基线设置后, 将p1与p2节点间的GE4/0-GE4/0链路执行中断操作, 该步骤导致Underlay层的IGP路由重新收敛。执行网络评估后, 仿真平台依据最新的转发状态重新计算100条业务的转发路径, 并对比故障前的路径数据。从分析结果看, 链路中断使12条业务受到影响, 由于冗余组网设计, 业务并未中断, 均为转发路径发生变化; 再次统计每条链路的业务承载量, 链路p1|GE4/0-p2|GE4/0中断后所有链路承载业务量变化如图7所示。明显地, 由于p1与p2间链路中断, 业务承载量为0; 另外, 共有4条链路业务承载量增加, 如dc1到dc2的业务, 原最优路径经过p1→p2, 现需要从p1→p3或p1→p4绕行, 导致相应的业务承载量增加, 增幅均为33%; 同时, 共有6条链路业务承载量下降, 如dc1到dc4的业务, 其中一条负载分担路径为dc1→pe1→p1→p2→p6→pe4→dc4, 现p1与p2间不可达, 导致该条路径失效, 进而导致p2→p6间承载业务变少, 降幅为18%。由于冗余拓扑设计, 故障前后业务通达率没有变化, 均为100%。

上述实验表明, 在典型链路故障模拟场景下, 本文仿真推演方法能够准确刻画故障触发后的业务

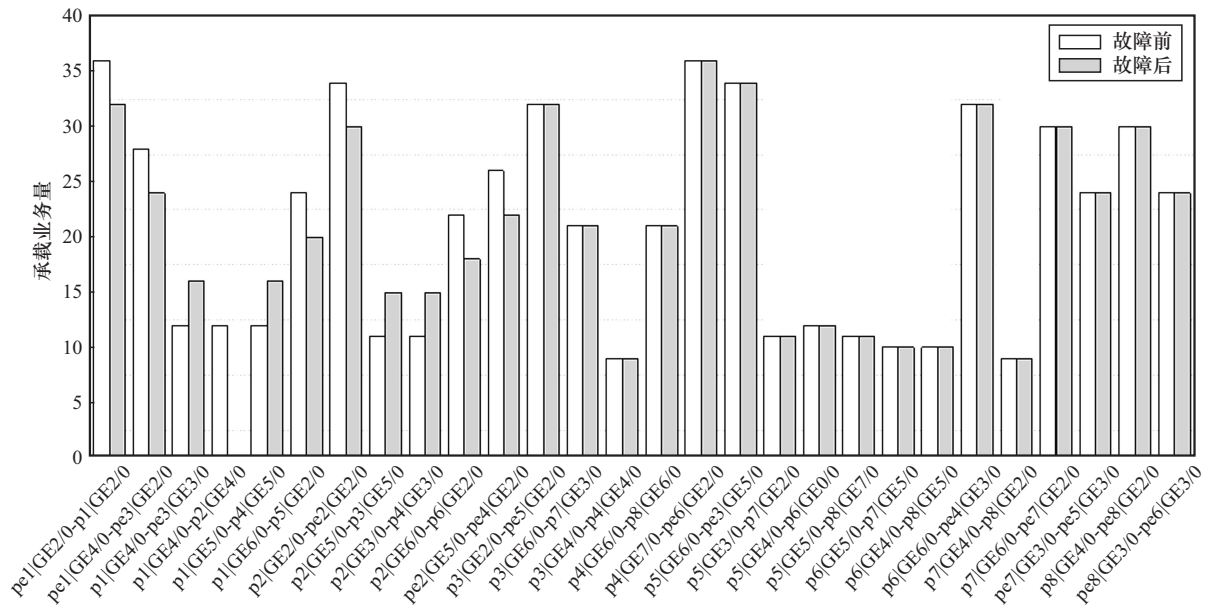


图7 链路p1|GE4/0-p2|GE4/0中断后所有链路承载业务量变化

迁移与路径变化过程。通过对多种故障恢复方案的预演分析，可在不影响现网运行的前提下评估其对业务承载与转发路径的影响，结果符合IP承载网对可靠性、冗余性和可恢复性的工程设计要求，验证了本文方法在实际网络运维与故障处置场景中的工程可行性与应用价值。

3.2 业务割接场景

业务割接场景仍使用故障模拟场景的100条预置业务，结束上述故障模拟推演流程，将仿真环境还原为初始状态，此时为每条业务配置一对IP通信对，并预置120 kbit/s双向流量，结合业务转发路径与负载分担情况，可得到环境中每条链路的双向流量负载情况。现拟将pe2设备做替换，将pe2的业务短暂迁移到pe4，pe2替换完成后，再切回业务。割接过程中，业务迁移会引起链路负载的变化；为了保证割接顺利，需要确保割接过程中业务转发正常，监控链路负载是否超过指定阈值。

业务割接场景仿真推演步骤如图8所示。割接前pe2上共承载28条业务，割接过程逻辑上分为4个阶段：第一阶段将pe2上的28条业务转移至pe4，第二阶段移除pe2节点，第三阶段创建新的pe2节点，第四阶段将28条业务迁回至新pe2节点。每个阶段均拆解为若干原子化操作，按顺序依次执行。

仿真推演过程中，在step2阶段执行完成后进行全量仿真分析，判断业务健康度与网络承载质量。具体地，节点pe2业务迁移至pe4后所有链路单向承载流量变化如图9所示。由图9可以看出，在迁移后，pe2暂时不承载业务，p2→pe2与pe2→pe4两条链路流量归零；共有11条链路流量负载增加，其中链路p6→pe4增长最多，达到123%，此时pe4除了承载自身20条业务外，还额外承载pe2上的28条业务，共计48条业务，所以到达pe4的流量增多；另外，同样有11条链路负载降低，其中，链路pe1→p1与pe1→pe3分别下降了83%与

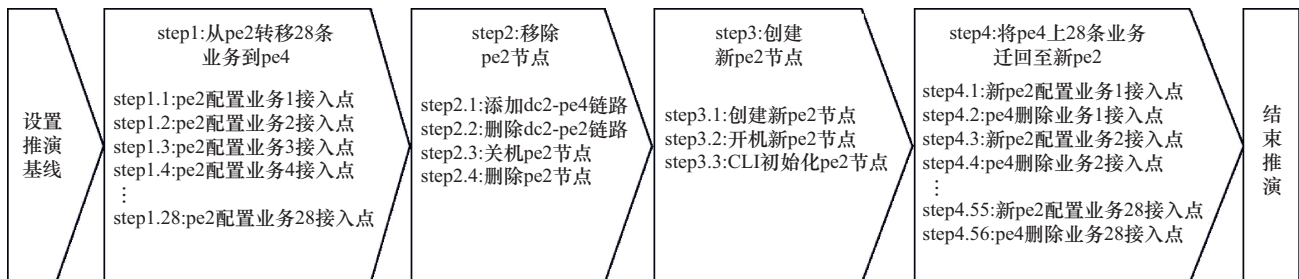


图8 业务割接场景仿真推演步骤

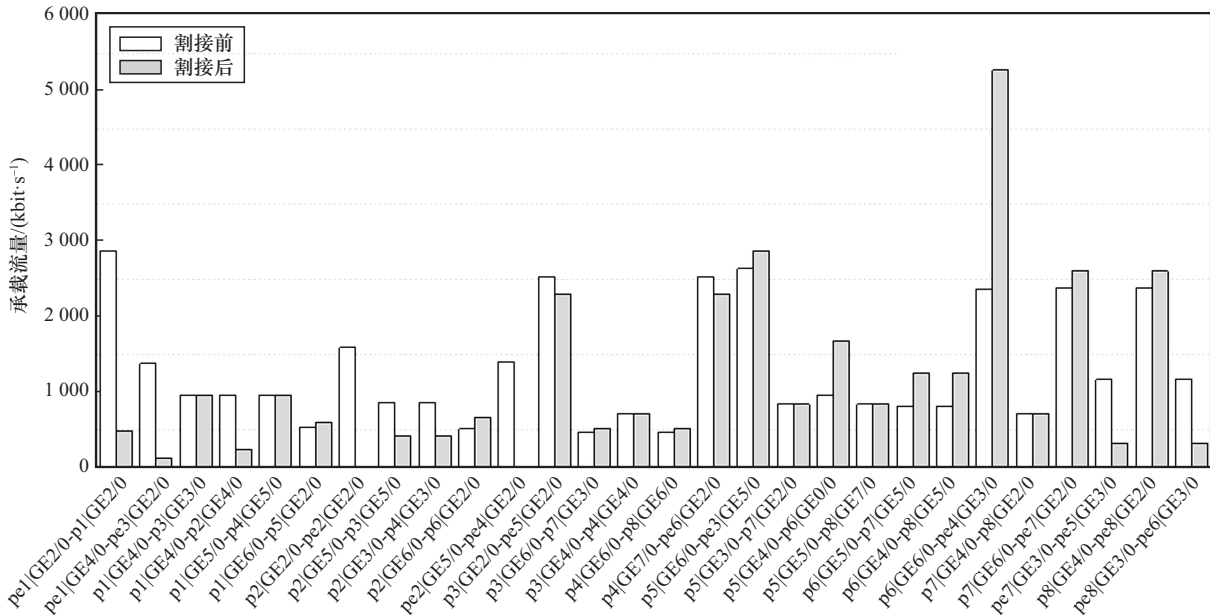


图9 节点pe2业务迁移至pe4后所有链路单向承载流量变化

91%，这是由于经由该路径的负载分担流量减少，主要保留了pe1自身承载业务的流量。以接口带宽来看，所有链路承载流量均未超过链路负载上限，不会出现拥塞或业务丢包现象；另外，关于业务迁移后的网络通达情况，与故障模拟中的分析方法相同。

实验表明，在典型业务割接场景下，本文方法能够对割接前后的业务承载关系与转发路径变化进行准确刻画；通过在仿真阶段评估割接操作对业务连续性和链路负载的影响，可有效识别潜在的拥塞与路径异常风险，为IP承载网的业务运维提供可靠的决策支撑与风险规避手段。

3.3 性能对比与分析

为进一步验证本文方法在回溯机制推演效率与实时反馈分析粒度上的优势，选取传统数值仿真(NS3)、传统模拟仿真(EVE-NG)作为基线方法，在相同24节点IP承载网实验环境、100条VPN业务配置的基础上，设计针对性对比实验。另外，为分析大规模网络业务场景下推演分析性能及精度，对实验环境进行扩容，额外部署两套KVM虚拟化节点，形成3节点虚拟化资源池，以2vCPU、4GB内存为虚拟网元规格，设计多种规模组合对比实验。

3.3.1 推演效率分析

实验基于IP承载网业务割接场景，按原子化操作分类依次执行10个连续原子操作，在第10个原子操作执行后检测到网元配置冲突错误，需统一回退至第5个原子操作执行后的状态。核心指标为状态

恢复耗时 (time for state recovery, TSR)，定义为从发现配置错误到仿真环境完全恢复至目标历史状态的总耗时。分别采用3种方法完成回退操作，本文方法基于CoW差分快照机制一键调用历史快照恢复；EVE-NG通过人工逐一步骤逆向删除配置，失败则重启网元镜像并重新执行前5步操作；NS3通过修改仿真脚本、删除第6~10步代码后重新运行前5步仿真程序。实验重复5次，结果取平均值。

3种仿真方法的TSR对比结果如表4所示。本文方法基于CoW差分快照仅记录数据变化部分，实现秒级精准恢复，平均TSR仅为5s，状态恢复成功率为100%；EVE-NG依赖人工逆向操作，平均TSR为126s，因配置依赖问题存在20%的恢复失败率，失败后需重启镜像并重新配置，进一步增加耗时；NS3需重新运行前5步仿真全流程，无法复用已完成的仿真状态，平均TSR为95s。本文方法较EVE-NG、NS3的平均TSR分别降低96%、95%，推演效率优势显著。

表4 3种仿真方法的TSR对比结果

仿真方法	状态恢复方式	平均TSR/s	状态恢复成功率
本文方法	CoW 差分快照一键恢复	5	100%
EVE-NG	人工逆向配置+镜像重启	126	80%
NS3	修改脚本+重新运行前序仿真	95	100%

3.3.2 故障定位效率分析

基于故障模拟场景的多步变更序列, 依次执行 7 个连续原子操作, 在第 5 步原子操作执行过程中注入配置故障, 表现为部分业务流量丢包。核心指标为故障根因定位时延 (fault localization delay, FLD), 定义为从故障注入到系统输出准确故障根因的总耗时。本节分别采用 3 种方法完成故障根因定位: 本文方法通过步骤级实时评估, 每步操作后即时采集拓扑、配置、业务、流量数据并分析; EVE-NG、NS3 均采用仿真全流程结束后输出整体性能指标的模式, 需人工反向推导故障根因。实验重复 5 次, 结果取平均值。

3 种仿真方法的 FLD 对比结果如表 5 所示。本文方法较 EVE-NG、NS3 的平均 FLD 分别降低 91%、88%。通过引入步骤级实时评估机制, 突破了传统仿真全流程结束后整体分析的局限性, 实现了故障根因的单步即时、自动化、精准化定位, 显著提升了仿真分析粒度与实时反馈能力。

表 5 3 种仿真方法的 FLD 对比结果

仿真方法	评估模式	平均 FLD/s	定位形式
本文方法	步骤级实时评估	8	步骤级配置、业务分析对比
EVE-NG	仿真结束后整体评估	93	业务 ping 包+人工分析
NS3	仿真结束后 KPI 曲线输出	68	业务丢包 KPI 曲线+人工分析

3.3.3 大规模性能分析

为评估本文方法在不同网元节点与业务规模下的运行效率与推演精度, 基于扩容后的虚拟化资源池, 设计双因素性能测试实验。实验以网元规模 $N \in \{24, 128, 256\}$ 与业务规模 $B \in \{200, 1000\}$ 为变量, 在所有组合配置 $\{N, B\}$ 上进行测试, 对于每组配置, 记录仿真网元开销与性能类相关指标。网元 CPU 负载与内存负载评估当前规模下虚拟化资源利用率; 单步推演耗时记录节点类或链路类原子化操作执行时间, 每组执行 10 次变更; 路径还原准确率统计本规模下所有业务的转发路径是否准确计算与还原; 全量评估耗时为执行一次全量评估所需的时间, 每组执行 3 次评估。

节点与业务规模双因素性能测试对比如表 6 所示。由表 6 可以看出, 平均网元 CPU 与内存负载均与网元

规模呈正相关, 但在固定 N 值时, 对于 K 级的业务规模变化不敏感, 这一特性源于数模双驱动仿真引擎的分层建模设计。与此同时, 各配置下的单步推演耗时保持稳定, 未表现出随 N 或 B 显著增长, 说明原子化操作具有良好的效率与可扩展性。精度方面, 无论网元负载水平如何, 只要路由能够正常收发且完成收敛, 路径还原准确率始终保持为 100%, 路径计算具备可靠性与结果一致性。虽然全量评估耗时随 $\{N, B\}$ 增大呈上升趋势, 但其增长幅度处于可控范围内, 仍可满足工程场景下响应时效要求。

表 6 节点与业务规模双因素性能测试对比

网元规模	业务规模	平均网元 CPU 负载	平均网元内存负载	平均原子化变更耗时/s	路径还原准确率	平均全量评估耗时/s
24	200	25%	38%	1	100%	3
24	1000	26%	37%	1	100%	6
128	200	42%	49%	2	100%	11
128	1000	43%	47%	3	100%	16
256	200	55%	53%	3	100%	23
256	1000	57%	54%	3	100%	30

4 结束语

本文面向 IP 承载网规模与复杂度持续增长的场景, 研究了传统网络仿真流程存在的过程黑盒化、操作不可逆、结果反馈滞后等问题。首先, 提出支持分步推演与状态回溯的仿真机制, 通过原子化验证、单步操作隔离与状态快照管理, 实现仿真步骤的实时跟踪与灵活回溯。随后, 围绕该机制的技术支撑体系展开设计, 提出基于数模双驱动仿真引擎的仿真范畴划分方法, 设计了约束算路的业务路径还原算法及多维度仿真评估方法。最后, 通过网络故障模拟、业务割接预演场景与横向对比实践验证, 证明该仿真流程能为 IP 承载网运维提供可验证的决策支撑, 有效提升网络规划与优化的可靠性。在未来的工作中, 将聚焦超大规模网络与动态复杂业务场景, 进一步研究数模双驱动协同优化与高效融合机制, 持续提升数字孪生网络仿真的实时性与规模化推演能力。

参考文献:

- [1] Pan Y D, Lei L, Shen G Q, et al. A survey on digital twin networks: architecture, technologies, applications, and open issues[J]. IEEE Internet

- of Things Journal, 2025, 12(12): 19119-19143.
- [2] Liu G Y, Zhu Y H, Kang M C, et al. Native design for 6G digital twin network: use cases, architecture, functions, and key technologies[J]. IEEE Internet of Things Journal, 2025, 12(15): 29135-29151.
- [3] Zhao J, Xiong X M, Chen Y M. Design and application of a network planning system based on digital twin network[J]. IEEE Journal of Radio Frequency Identification, 2022, 6: 900-904.
- [4] Frömmgen A, Stohr D, Koldehofe B, et al. Don't repeat yourself: seamless execution and analysis of extensive network experiments[C]//Proceedings of the 14th International Conference on Emerging Networking EXperiments and Technologies. New York: ACM Press, 2018: 20-26.
- [5] Almasan P, Ferriol-Galmés M, Paillisse J, et al. Network digital twin: context, enabling technologies, and opportunities[J]. IEEE Communications Magazine, 2022, 60(11): 22-27.
- [6] Kim E, Riley G. Automatic state saving and rollback in ns-3[C]//Proceedings of the 2017 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation. New York: ACM Press, 2017: 263-266.
- [7] ITU-T Y.3090: 2022 Digital twin network-requirements and architecture [S].
- [8] IETF. Digital twin network: concepts and reference architecture: draft-irtf-nmrg-network-digital-twin-arch-04[S]. 2025-07-19.
- [9] ETSI. Unlocking digital transformation with autonomous networks: ETSI white paper No.48[R]. 2023.
- [10] 中国移动研究院. 数字孪生网络(DTN)白皮书[R]. 2021. China Mobile Research Institute. Digital twin network (DTN) white paper[R]. 2021.
- [11] 中国联合网络通信有限公司研究院. 中国联通数字孪生网络白皮书[R]. 2023. Research Institute of China United Network Communications Co., Ltd. China Unicom digital twin network white paper[R]. 2023.
- [12] Chen M, Miao Y M, Humar I. OPNET IoT simulation[M]. Singapore: Springer Singapore, 2019.
- [13] Xian X D, Shi W R, Huang H. Comparison of OMNET++ and other simulator for WSN simulation[C]//Proceedings of the 2008 3rd IEEE Conference on Industrial Electronics and Applications. Piscataway: IEEE Press, 2008: 1439-1443.
- [14] Campanile L, Gribaudo M, Iacono M, et al. Computer network simulation with ns-3: a systematic literature review[J]. Electronics, 2020, 9(2): 272.
- [15] Chen J, Zheng X Q, Zhang L, et al. WLAN simulation experiment based on ENSP[J]. Journal of Physics: Conference Series, 2019, 1325(1): 012046.
- [16] 邱鹏, 霍瑛, 蒋悦. 基于H3C Cloud Lab的企业网络设计与仿真[J]. 实验室研究与探索, 2020, 39(10): 118-125. Qiu P, Huo Y, Jiang Y. Design and simulation of enterprise networks based on H3 C cloud lab[J]. Research and Exploration in Laboratory, 2020, 39(10): 118-125.
- [17] Harahus M, Čavojský M, Bugár G, et al. Interactive network learning: an assessment of EVE-NG platform in educational settings[J]. Acta Electrotechnica et Informatica, 2023, 23(3): 3-9.
- [18] Ajiardiawan G, Suryanto Y. Performance analysis of segment routing on MPLS L3VPN using PNETLAB[C]//Proceedings of the 2021 International Conference on Artificial Intelligence and Computer Science Technology (ICAICST). Piscataway: IEEE Press, 2021: 185-190.
- [19] Kormiyenko B, Galata L, Ladieva L. Research of information protection system of corporate network based on GNS3[C]//Proceedings of the 2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT). Piscataway: IEEE Press, 2019: 244-248.
- [20] Wu Y W, Zhang K, Zhang Y. Digital twin networks: a survey[J]. IEEE Internet of Things Journal, 2021, 8(18): 13789-13804.
- [21] Ali Madni M A, Iranmanesh S, Raad R. DTN and Non-DTN Routing Protocols for Inter-CubeSat Communications: a comprehensive survey[J]. Electronics, 2020, 9(3): 482.
- [22] Nguyen H X, Trestian R, To D, et al. Digital twin for 5G and beyond[J]. IEEE Communications Magazine, 2021, 59(2): 10-15.
- [23] Hui L B, Wang M W, Zhang L, et al. Digital twin for networking: a data-driven performance modeling perspective[J]. IEEE Network, 2023, 37(3): 202-209.
- [24] Yang L, Xu Z. Research on industrial digital twin technology system and key technologies[J]. Information and Communications Technology and Policy, 2021, 47(1): 8-13.
- [25] Jia W J, Wang W, Zhang Z Z. From simple digital twin to complex digital twin Part I: a novel modeling method for multi-scale and multi-scenario digital twin[J]. Advanced Engineering Informatics, 2022, 53: 101706.
- [26] Dai Y Y, Zhang K, Maharjan S, et al. Deep reinforcement learning for stochastic computation offloading in digital twin networks[J]. IEEE Transactions on Industrial Informatics, 2021, 17(7): 4968-4977.
- [27] 石鸿伟, 苏琛, 倪中阳, 等. 面向IP承载网的数模双驱动孪生网络系统架构研究[J]. 电信科学, 2025, 41(9): 1-15. Shi H W, Su C, Ni Z Y, et al. Research on hybrid-driven digital twin network system architecture for IP-based carrier network[J]. Telecommunications Science, 2025, 41(9): 1-15.
- [28] Moura L D, Björner N. Z3: an efficient SMT solver[C]//International conference on Tools and Algorithms for the Construction and Analysis of Systems. Berlin: Springer, 2008: 337-340.
- [29] 石鸿伟, 倪中阳, 陆干沂, 等. 高性能网络数字孪生仿真引擎研究[J]. 物联网学报, 2026, 10(1): 161-171. Shi H W, Ni Z Y, Lu G Y, et al. Research on high-performance network digital twin simulation engine[J]. Chinese Journal on Internet of Things, 2026, 10(1): 161-171.

[作者简介]



石鸿伟 (1982-), 男, 东南大学博士生, 紫金山实验室高级工程师, 主要研究方向为未来网络体系架构、软件定义网络、网络智能控制、网络数字孪生等。



倪中阳 (1987-), 男, 紫金山实验室研究员, 主要研究方向为新型网络协议、数字孪生网络、网络管控技术等。



黄韬 (1980-), 男, 博士, 北京邮电大学教授、博士生导师, 主要研究方向为路由与交换、软件定义网络、内容分发网络、确定性网络、算力网络等。