

• 计算机工程与应用 •



# 面向多用户的微观交通仿真实验系统设计

吴春江, 周世杰\*, 陈鹏飞

(电子科技大学信息与软件工程学院 成都 610054)

**【摘要】**针对现有交通虚拟仿真系统既无法支持多用户同时在线仿真, 又无法满足实际的虚拟仿真教学实验需求, 也难以大规模扩展等问题, 该文首次从系统架构和仿真引擎方面设计并实现了一种面向多用户的微观交通仿真实验系统。并且从单仿真引擎仿真车辆数变化、单仿真引擎仿真加速比变化和多仿真引擎并发执行3个性能指标对该系统进行测试。测试结果表明, 该文所实现的交通虚拟仿真系统能够支持多仿真引擎、多仿真任务的并发执行, 满足多用户的同时在线仿真需求。

**关键词** 并发执行; 分布式; 多用户; 仿真引擎; 交通仿真

中图分类号 TP311 文献标志码 A doi:10.12178/1001-0548.2020446

## Design and Research of Multi-User Oriented Micro Traffic Simulation Experiment System

WU Chun-jiang, ZHOU Shi-jie\*, and CHEN Peng-fei

(School of Information and Software Engineering, University of Electronic Science and Technology of China Chengdu 610054)

**Abstract** In view of the problems that the existing traffic virtual simulation systems have, such as they can neither simultaneously support real-time online simulation for multiple users, nor meet the practical requirements of virtual simulation teaching experiment and even be difficult to expand on a large scale. In this paper, from the aspects of system architecture and simulation engine, a multi-user-oriented micro-traffic simulation experiment system is for the first time investigated and devised. Furthermore, this paper tests the designed system from three performance indicators including the vehicle number change of single simulation engine simulation, the acceleration ratio change of single simulation engine simulation, and the concurrent execution of multiple simulation engines. The experimental results indicate that the traffic virtual simulation system devised in this paper can support the concurrent execution of multiple simulation engines and multiple simulation tasks as well as meet the simultaneous online simulation needs of multiple users.

**Key words** concurrent execution; distribution; multi-user; simulation engine; traffic simulation

随着大数据、人工智能、物联网等新兴技术的发展, 智慧交通已成为智慧城市建设的关键环节, 面向智慧交通的虚拟仿真实验也成为国家虚拟仿真实验教学项目的重要组成部分。在城市管理、交通工程、交通运输等专业的教学实验中, 往往需要开展单路口、单道路、区域路网甚至城市路网的复杂交通实验。然而, 通过在实际交通环境中调整交通信号灯、改变路网结构、增减车道数量等路网参数来开展以上实验, 成本高、影响大, 绝大多数情况下不可行。为此, 往往需要

借助道路交通虚拟仿真系统, 运用计算机仿真技术, 模拟道路交通复杂状况, 揭示交通流时空变化规律, 有利于用户掌握道路交通设计、预测、控制、诱导等知识与技术。

目前的交通仿真软件如 Aimsun next<sup>[1]</sup>、Trans Modeler<sup>[2]</sup>、Paramics<sup>[3]</sup>、Vissim<sup>[4]</sup>、CORSIM<sup>[5]</sup>、Matsim<sup>[6]</sup>、MITSim<sup>[7]</sup>等都只是对具体场景进行仿真, 不能满足虚拟仿真教学实验需求, 即使是开源的 SUMO<sup>[8]</sup>也不支持实时在线仿真以及在指定路网中进行不同类别的仿真实验。

收稿日期: 2020-12-21; 修回日期: 2021-04-24

基金项目: 四川省科技厅重大专项(2018GZDZX0033); 四川省科技厅重点研发项目(2020YFG0143)

作者简介: 吴春江(1982-), 男, 博士生, 高级工程师, 主要从事交通仿真及人工智能方面的研究。

通信作者: 周世杰, E-mail: sjzhou@uestc.edu.cn

# 1 研究进展

交通仿真作为计算机仿真的一个分支, 起源于 20 世纪 50 年代<sup>[9]</sup>, 该技术能够在计算机上实现对真实路网及车流状况的模拟, 用以对交通态势随时间和空间的变化进行跟踪描述。

本文对 10 个具有代表性的交通仿真系统从仿真类型、代码开源、仿真功能、仿真地图、车辆行为模型、在线仿真、仿真控制等几个方面进行比较, 结果如表 1 所示。目前的交通仿真软件<sup>[10-20]</sup>都只是对具体场景进行仿真且功能较少, 并不能满足虚拟仿真教学实验的要求, 不支持实时在线仿真以及在指定路网中进行不同类别的仿真实验, 并且这些仿真系统操作起来都非常复杂。

针对以上这些问题, 本文首次研究并设计了一

种面向多用户的微观交通仿真实验系统 (<http://its.uestc.cn>), 具体贡献如下:

1) 设计并实现了一种微观交通仿真引擎, 该仿真引擎具有消耗低、易扩展等特点, 支持独立运行和分布式通信;

2) 设计并实现了一种面向多用户的微观交通仿真实验系统, 该系统采用分布式计算、分布式消息通信等技术, 支持多个仿真引擎、多个仿真任务并发执行;

3) 本文对面向多用户的微观交通仿真实验系统进行了性能测试, 测试结果表明微观交通仿真引擎的性能与仿真车辆数和仿真加速比成正比, 并且在单台服务器上, 该仿真实验系统能够支持多个仿真引擎、多个仿真任务的并发执行。

表 1 交通仿真系统对比

| 仿真系统                         | 仿真类型     | 代码开源 | 仿真功能                               | 仿真地图            | 车辆行为模型                           | 在线仿真 | 仿真控制 |
|------------------------------|----------|------|------------------------------------|-----------------|----------------------------------|------|------|
| Aimsun <sup>[10]</sup>       | 微观/介观/宏观 | 否    | 评估交通仿真、交通需求管理策略评估                  | 支持路网导入并提供路网编辑功能 | Gipps的安全距离模型改进<br>Gipps的换道模型改进   | 否    | 是    |
| TransModeler <sup>[11]</sup> | 微观/介观/宏观 | 否    | 交通仿真和交通规划, 与交通预测软件结合等              | 支持路网导入并提供路网编辑功能 | 心理-生理跟驰模型基于规则的换道模型               | 否    | 是    |
| Paramics <sup>[12]</sup>     | 微观       | 否    | 交通仿真模拟, 协助交通评估等                    | 支持路网导入并提供路网编辑功能 | 基于规则的换道模型                        | 否    | 是    |
| PTV Vissim <sup>[13]</sup>   | 微观       | 否    | 交通分析和预测, 信号优化等                     | 支持路网导入并提供路网编辑功能 | 基于规则的换道模型                        | 否    | 是    |
| SUMO <sup>[14]</sup>         | 微观/介观/宏观 | 是    | 提供自定义模型, 离散时间下的微观交通仿真              | 支持路网导入并提供路网编辑功能 | Stefan Krauß模型的拓展<br>SUMO自带的换道模型 | 否    | 是    |
| Corsim <sup>[15]</sup>       | 微观/介观/宏观 | 否    | 交通仿真, 提供多种车道方案                     | 支持路网导入并提供路网编辑功能 | Pitt跟驰模型Pitt换道模型                 | 否    | 是    |
| Matsim <sup>[16]</sup>       | 微观       | 是    | 需求建模、公共交通、多功能分析, 提供基于代理的动态交通仿真等    | 开源api, 提供路网编辑功能 | 多代理交通模型                          | 否    | 是    |
| MITsim <sup>[18]</sup>       | 微观       | 是    | 交通仿真可模拟监视能力, 支持需求建模、公共交通、多功能分析等    | 提供路网编辑功能        | 心理-生理模型与GHR模型基于规则的换道模型           | 否    | 是    |
| TJTS <sup>[18]</sup>         | 微观       | 否    | 微观交通仿真, 基于仿真数据的评估功能, 提供城市道路实验和测试平台 | /               | /                                | 否    | 是    |
| SUTSS <sup>[19]</sup>        | 微观/介观/宏观 | 否    | 提供交通规划、交通建设、运行管理等                  | /               | /                                | 否    | 是    |
| 本系统                          | 微观       | 否    | 在线交通微观仿真                           | 提供路网编辑模块        | 自带的换道模型/换道模型                     | 是    | 是    |

# 2 系统设计

## 2.1 系统架构设计

本文所设计的一种面向多用户的微观交通仿真实验系统的架构如图 1 所示。

在面向多用户的微观交通仿真实验系统中, 每个交通仿真引擎执行独立的仿真任务, 各个仿真引擎之间相互独立、互不干扰。单台仿真引擎服务器可以创建出多个仿真引擎, 支持多个仿真任务的并发执行。再通过分布式计算技术和分布式消息通信

机制，将多台仿真引擎服务器进行统一管理，实现对每台服务器上的每个仿真引擎的统一管理。通过仿真引擎服务器数量的增加，可以实现更多仿真引擎的管理与并行计算，达到面向多用户的交通仿真任务并发执行的目的。

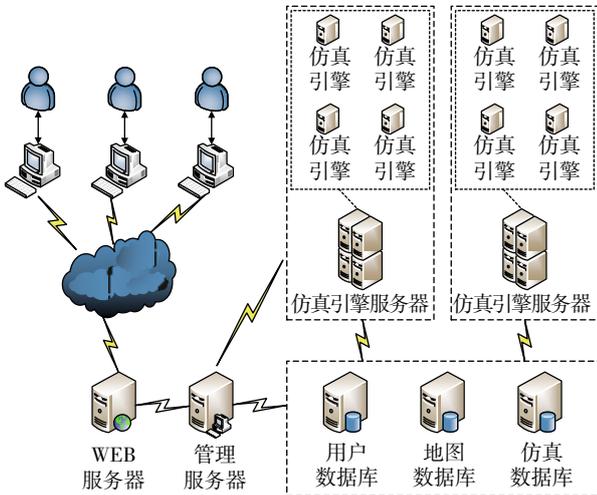


图 1 面向多用户的微观交通仿真实验系统架构图

### 2.2 仿真引擎调度

本文所设计的面向多用户的微观交通仿真实验系统，以仿真任务为功能单元，每个仿真引擎执行独立的仿真任务，各个仿真引擎之间相互独立、互不干扰，再通过微观交通仿真实验系统统一的任务调度和仿真引擎管理，创建多个仿真引擎执行不同的仿真任务，从而实现面向多用户的并发仿真任务。

如图 2 所示，在面向多用户的微观交通仿真实验系统中，每个仿真引擎都是一个独立的进程或安装在 Docker 容器中，仿真引擎管理模块负责仿真引擎进程或 Docker 容器的创建及回收。

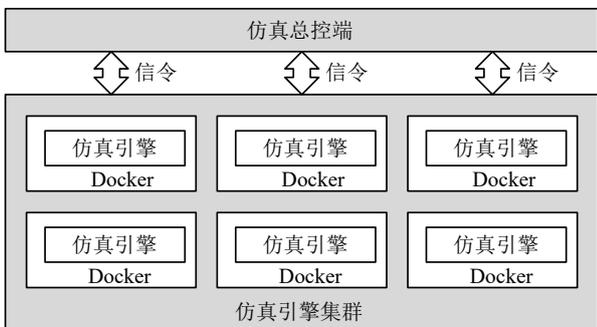


图 2 仿真引擎集群示意图

当有新的仿真任务创建时，仿真引擎管理模块会根据仿真任务的规模来预估仿真任务所需要的计算资源。如果系统中空闲的计算资源满足仿真任务

的需要，仿真引擎管理模块则会分配一个仿真引擎去执行仿真任务。如果系统中没有足够的计算资源，仿真引擎管理模块首先会将仿真任务放到等待队列中，等待其他仿真任务执行结束释放出足够的计算资源，再分配一个仿真引擎去执行该仿真任务。等待队列中的仿真任务按照先来先得的顺序分配仿真引擎和计算资源。并且，根据前面所设计的系统架构，本文设计的实验系统可以通过增加仿真引擎服务器的数量来扩展计算资源，用以满足多个不同规模仿真任务的运行需求。

### 2.3 仿真任务通信

本文所设计的面向多用户的微观交通仿真实验系统采用 kafka 作为任务管理端和仿真引擎之间的通信方式，以满足高吞吐量的分布式消息通信。

当仿真引擎和计算资源分配成功后，任务管理模块会将创建仿真任务需要的参数数据放入 kafka 消息队列当中，仿真引擎会从 kafka 消息队列中获取参数数据，进行任务创建。同时，仿真引擎在执行完一定步数的仿真计算后，也会将仿真结果放入 kafka 队列中，以供仿真总控端提取仿真数据，如图 3 所示。

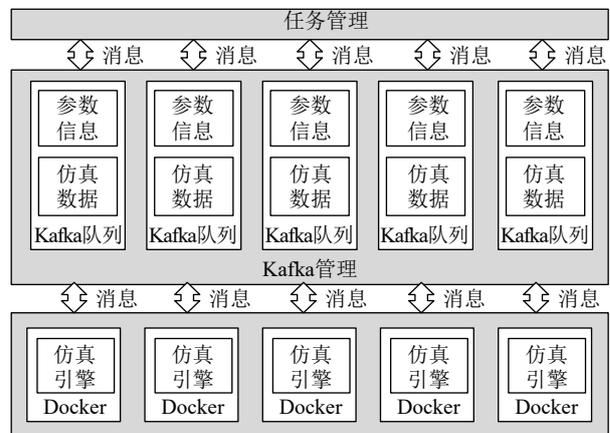


图 3 分布式交通仿真引擎消息通信

在面向多用户的微观交通仿真实验系统中，仿真数据分为车辆仿真数据和信号灯仿真数据。为了区分这两种实验数据，仿真引擎在发送到 kafka 队列的时候，会对数据加以处理，对于车辆仿真数据，数据是以“vehicle\_”加上 16 位的仿真 ID 号组成；对于信号灯数据，数据是以“phase\_”加上 16 位的仿真 ID 号组成。当 kafka 队列收到这些数据之后，可以通过数据的标志进行车辆仿真数据和信号灯仿真数据的判断。

### 3 性能测试

根据本文所提出的微观交通仿真实验系统设计思路, 需要达到面向多用户并发执行仿真任务的目的, 为此, 本文首先对该微观交通仿真实验系统进行性能分析。

根据所描述的仿真引擎设计, 在本文微观交通仿真引擎设计中, 所有仿真车辆的每一步仿真计算方式都是相同的。由此可以分析出, 本文所设计的微观交通仿真引擎的计算消耗与仿真车辆数强相关, 仿真车辆数越大, 每一步仿真计算的消耗就越大, 其时间复杂度为  $step \times O(n)$ , 其中,  $step$  为仿真加速比 (即仿真步数),  $n$  为仿真车辆数。同时, 在本文所设计的微观交通仿真引擎中, 每个仿真车辆占用的内存为 100 B 左右, 100 万仿真车辆占用的内存仅 100 M。

根据以上分析, 本文所设计的微观交通仿真引擎对 CPU 的计算消耗需求大, 对内存的消耗可忽略不计。同时, 本文所设计的微观交通仿真引擎的计算时间复杂度为  $step \times O(n)$ , 即与仿真加速比和仿真车辆数相关。因此, 本文设计了单仿真引擎仿真车辆数变化、单仿真引擎仿真加速比变化和多仿真引擎并发执行 3 种实验来测试性能。

本文所设计的微观交通仿真引擎性能测试的实验环境如表 2 所示。

表 2 微观交通仿真引擎性能测试实验环境

| 实验环境       | 实验环境描述                                     |
|------------|--|
| CPU        | Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz |
| 内存/GB      | 64   |
| 操作系统       | Windows 专业版                                |
| 空闲CPU占用率/% | 6~7  |

#### 3.1 单仿真引擎仿真车辆数变化性能测试

单仿真引擎仿真车辆数变化主要测试仿真车辆数对计算资源的消耗。本文选取双向 8 车道的 9 个十字路口地图作为仿真地图, 该仿真地图的最大容量为 2400 辆仿真车辆。在单仿真引擎仿真车辆数测试实验中, 车辆生成速率设置为 50, 仿真加速比固定设置为 20, 通过仿真车辆数的变化观察单仿真引擎的 CPU 资源消耗变化。测试结果如表 3、图 4、图 5、图 6 所示。

可以看出, 随着仿真车辆数的增加, 单仿真引擎所占用的 CPU 与仿真计算耗时也随之增加, 增长趋势基本相符, 与仿真引擎的性能分析结果一致。

表 3 单仿真引擎仿真车辆数性能测试

| 仿真步数 | 仿真车辆数 | CPU 占用率/% | 仿真计算耗时/ms |
|------|-------|-----------|-----------|
| 10   | 213   | 10.0      | 438       |
| 30   | 404   | 10.5      | 609       |
| 50   | 596   | 10.9      | 719       |
| 70   | 778   | 12.5      | 828       |
| 90   | 920   | 12.8      | 906       |
| 110  | 1071  | 14.3      | 1016      |
| 130  | 1214  | 14.2      | 1125      |
| 150  | 1370  | 15.3      | 1234      |
| 170  | 1471  | 15.8      | 1375      |
| 190  | 1589  | 16.6      | 1344      |
| 210  | 1669  | 17.0      | 1453      |
| 230  | 1744  | 16.9      | 1485      |
| 250  | 1813  | 17.1      | 1546      |
| 270  | 1875  | 17.3      | 1515      |
| 290  | 1940  | 17.7      | 1594      |
| 310  | 1992  | 18.3      | 1657      |
| 330  | 2027  | 18.6      | 1703      |
| 350  | 2051  | 18.6      | 1766      |
| 370  | 2089  | 18.7      | 1703      |
| 390  | 2125  | 18.6      | 1672      |
| 410  | 2136  | 18.8      | 1703      |
| 430  | 2164  | 19.4      | 1750      |
| 450  | 2158  | 19.5      | 1766      |
| 470  | 2174  | 19.4      | 1796      |

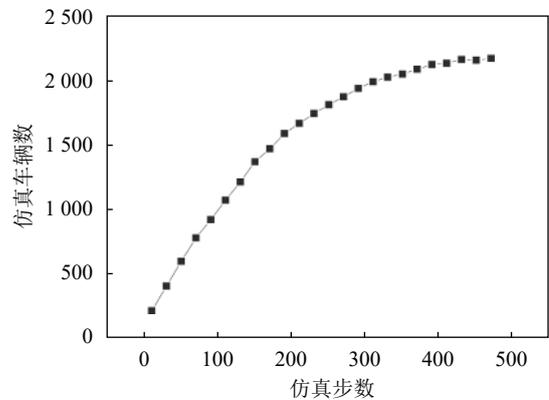


图 4 仿真车辆数变化

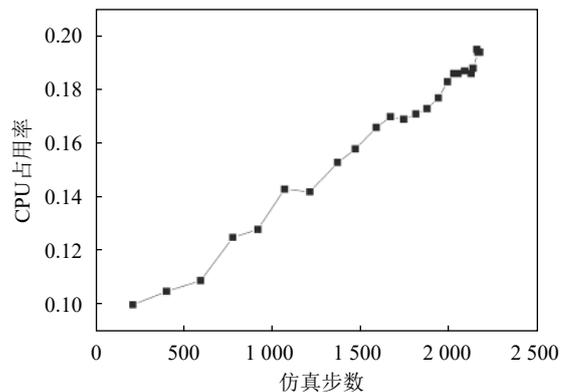


图 5 随仿真车辆数的 CPU 占用率

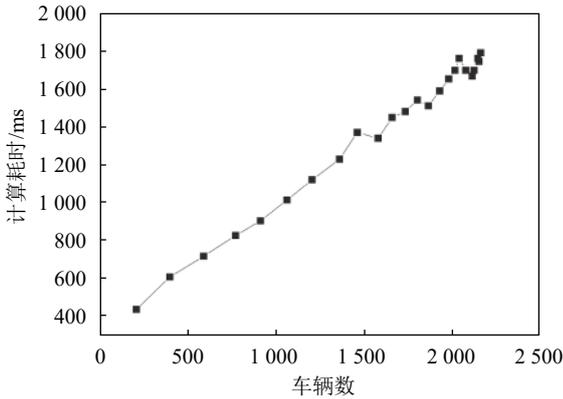


图 6 随仿真车辆数变化的计算耗时

### 3.2 单仿真引擎仿真加速比变化性能测试

单仿真引擎仿真加速比主要测试仿真加速比对仿真计算资源的消耗。本文选取双向 8 车道的十字路口地图作为仿真地图，该仿真地图的最大容量为 380 辆仿真车辆。实验中，仿真车辆生成速率设置为 50，当仿真车辆数达到仿真地图最大容量时，通过调整仿真加速比来观察 CPU 资源消耗的情况。测试结果如表 4、图 7、图 8 所示。

表 4 单仿真引擎仿真加速比变化性能测试

| 仿真加速比 | 仿真车辆数 | CPU 占用率/% | 仿真计算耗时/ms |
|-------|-------|-----------|-----------|
| 10    | 363   | 8.5       | 203       |
| 20    | 366   | 9.1       | 360       |
| 30    | 364   | 10.4      | 578       |
| 40    | 362   | 11.6      | 781       |
| 50    | 370   | 12.4      | 906       |
| 60    | 373   | 14.0      | 1218      |
| 70    | 374   | 14.8      | 1375      |
| 80    | 368   | 16.2      | 1578      |
| 90    | 363   | 18.2      | 1844      |
| 100   | 365   | 18.0      | 1984      |

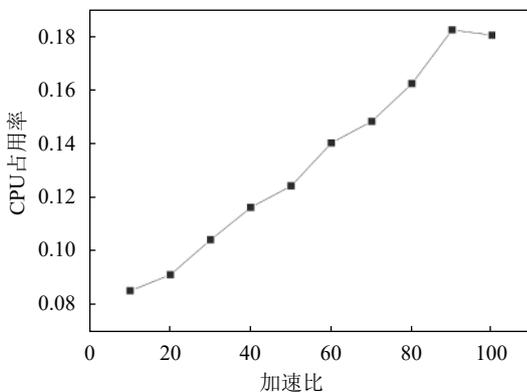


图 7 随仿真加速比变化的 CPU 占用率

从表 4、图 7、图 8 所展示的结果可以看出，随着仿真加速比的增加，单仿真引擎所占用的

CPU 与仿真计算耗时也随之增加，与仿真引擎的性能分析结果一致。

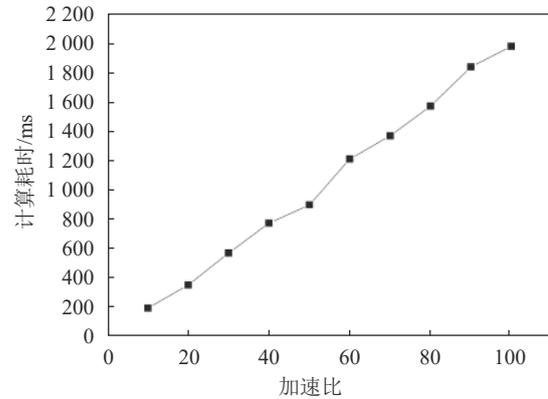


图 8 随仿真加速比变化的计算耗时

### 3.3 多仿真引擎并发执行性能测试

本文在如表 2 所示的单台服务器上，对面向多用户的微观交通仿真实验系统进行了多仿真引擎并发执行的性能测试。本文选取双向 8 车道的单十字路口地图作为仿真地图，该仿真地图的最大容量为 380 辆仿真车辆，通过对仿真引擎数的增加来观察仿真任务并发执行数对服务器 CPU 资源消耗的影响，测试结果如表 5、图 9 所示。

表 5 多仿真引擎并发执行性能测试

| 并发仿真引擎数 | CPU 最高占用率 (无任务)% | CPU 最高占用率 (任务并发执行)% |
|---------|------------------|---------------------|
| 0       | 4                | 18                  |
| 1       | 11               | 25                  |
| 2       | 18               | 36                  |
| 3       | 25               | 44                  |
| 4       | 32               | 51                  |
| 5       | 38               | 59                  |
| 6       | 45               | 68                  |
| 7       | 52               | 79                  |
| 8       | 60               | 87                  |
| 9       | 67               | 93                  |
| 10      | 76               | 100                 |

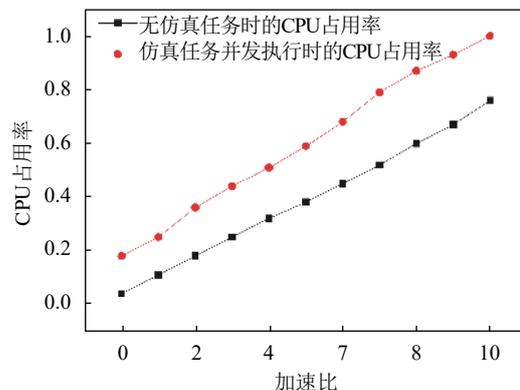


图 9 多仿真引擎并发执行的 CPU 占用率

从表5、图9所展示的结果可以看出, 在无仿真任务和仿真任务并发执行的情况下, CPU 占用率与仿真引擎的数量成线性比例增长。在单台服务器上, 本文所实现的一种面向多用户的微观交通仿真系统可以支持多个仿真引擎、多个仿真任务的并发执行。并且, 还可以通过增加服务器的数量来支持更多的仿真任务并发执行。

同时, 由于仿真引擎只有在请求仿真任务的时候才会增加对 CPU 的消耗, 而且仿真任务的执行时长是根据仿真请求步数确定的, 不会无限地执行下去, 会有一定的空闲期。因此, 在实际的应用过程中, 出现多个仿真任务并发执行的情况相对较少, 从理论上讲, 单台服务器可以支持更多的仿真引擎和仿真任务执行。

## 4 结束语

本文从仿真类型、代码开源、仿真功能、在线仿真、仿真控制等几个方面对多种交通仿真系统进行了横向比较。其次, 针对道路交通虚拟仿真实验的多用户、多并发、可操作等特性, 提出了一种面向多用户的微观交通仿真实验系统, 并从系统架构和仿真引擎方面对该仿真系统进行了设计。最后, 对面向多用户的微观交通仿真实验系统的性能进行了分析, 并从单仿真引擎仿真车辆数变化、单仿真引擎仿真加速比变化和多仿真引擎并发执行3个方面对该系统进行了测试。测试结果表明, 本文所实现的一种面向多用户的微观交通仿真系统可以支持多个仿真引擎、多个仿真任务的并发执行, 并且可以通过增加服务器的数量来支持更多的仿真任务并发执行, 能够满足多用户的同时在线仿真需求。

## 参 考 文 献

- [1] ANYA A R, ROUPHAIL N M, FREY H C, et al. Application of AIMSUN microsimulation model to estimate emissions on signalized arterial corridors[J]. *Transportation Research Record Journal of the Transportation Research Board*, 2014, 2428: 75-86.
- [2] 韦栋, 郑淑莹, 余文晟. 基于 TransModeler 的中观仿真模型研究与应用[J]. *交通科技与经济*, 2015, 17(2): 50-55.  
WEI Dong, ZHENG Shu-jian, SHE Wen-sheng. Research and application of microscopic traffic simulation model based on trans modeler software[J]. *Technology & Economy in Areas of Communications*, 2015, 17(2): 50-55.
- [3] CAMERON G, WYLIE B J N, MCARTHUR D. PARAMICS-moving vehicles on the connection machine[C]//Proceedings of the 1994 ACM/IEEE Conference on Supercomputing. Washington: IEEE, 1994: 291-300.
- [4] 盖春英. VISSIM 微观仿真系统及在道路交通中的应用[J]. *公路*, 2005(8): 118-121.  
GAI Chun-ying. Microscopic simulation system VISSIM and its application to road and transportation[J]. *Highway*, 2005(8): 118-121.
- [5] 吴亦政, 孙健, 彭春露, 等. Vissim 和 Corsim 在城市道路微观仿真中的对比研究[J]. *交通信息与安全*, 2012, 4: 116-121.  
WU Yi-zheng, SUN Jian, PENG Chun-lu, et al. Comparative study of micro-simulation performances for urban road network between Vissim and Corsim[J]. *Journal of Transport Information and Safety*, 2012, 4: 116-121.
- [6] SERRAS J. The multi-agent transport simulation MATSim[M]. London: Ubiquity Press, 2016.
- [7] CHEN Chen-yi, HU Jian-ming, GENG Chen-yao. Calibration of MITSIM and IDM car-following model based on NGSIM trajectory datasets[C]//Proceedings of 2010 IEEE International Conference on Vehicular Electronics and Safety. [S.l.]: IEEE, 2010: 48-53.
- [8] 韩光辉, 陈笑蓉, 俞洋, 等. 基于 SUMO 平台的微观交通仿真研究[J]. *计算机工程与科学*, 2012, 7: 195-198.  
HAN Guang-hui, CHEN Fu-rong, YU Yang, et al. A study of microscopic traffic simulation based on the SUMO platform[J]. *Computer Engineering & Science*, 2012, 7: 195-198.
- [9] 刘运通. 交通系统仿真技术[M]. 北京: 人民交通出版社, 2002.  
LIU Yun-tong. Traffic system simulation technology[M]. Beijing: China Communications Press, 2002.
- [10] 申丽君, 董悦, 周超雄, 等. 基于 AIMSUN 的城市交叉口优化方法仿真研究[J]. *公路交通科技*, 2010, 10: 265-268.  
SHEN Li-jun, DONG Yue, ZHOU Chao-xiong, et al. Simulation research on urban intersection optimization method based on AIMSUN[J]. *Journal of Highway and Transportation Research and Development*, 2010, 10: 265-268.
- [11] 杨慧, 成卫, 肖海承, 等. 基于 TransModeler 的拥堵区域交通流量调控方法研究[J]. *科学技术与工程*, 2011, 8: 1746-1750.  
YANG Hui, CHENG Wei, XIAO Hai-chen. Study of traffic flow adjustment methods to congestion area based on trans modeler[J]. *Science Technology and Engineering*, 2011, 8: 1746-1750.
- [12] 李伦. 微观交通仿真软件 Paramics 的工程应用实例[J]. *交通世界*, 2018, 9: 42-44.  
LI Lun. Engineering application examples of microscopic traffic simulation software Paramics[J]. *Transpoworld*, 2018, 9: 42-44.
- [13] 虞笑晨, 陈怡立, 王崇伟. 基于 Vissim 微观交通仿真的交叉口优化分析与研究[J]. *交通世界*, 2019, 2: 12-15.  
YU Xiao-cheng, CHEN Yi-li, WANG Chong-wei. Analysis and research on intersection optimization based on vissim microscopic traffic simulation[J]. *Transpoworld*, 2019, 2: 12-15.
- [14] ERDMANN J. Lane-changing model in SUMO[C]//Proceedings of the SUMO2014 Modeling Mobility with

- Open Data. Berlin: [s.n.], 2014, 24: 77-88.
- [15] OWEN L E, ZHANG Yun-long, RAO Lei, et al. Traffic flow simulation using CORSIM[C]//Winter Simulation Conference. [S.l.]: IEEE, 2000: 1143-1147.
- [16] 潘莹, 陈家琪. 基于 Multi-Agent 仿真的动态车辆路径算法研究[J]. 信息技术, 2015, 5: 121-124.
- PAN Yin, CHEN Jia-qi. Dynamic vehicle routing algorithm based on Multi-Agent simulation[J]. Information Technology, 2015, 5: 121-124.
- [17] 谢寒. 换车道模型研究[J]. 知识经济, 2010, 17: 5.
- XIE Han. Research on lane change model[J]. Knowledge Economy, 2010, 17: 5.
- [18] 邹智军, 杨东援. 城市道路交通仿真系统软件设计[J]. 交通运输工程学报, 2001, 1(3): 86-88.
- ZHOU Zhi-jun, YANG Dong-yuan. Software design of urban road traffic simulation system[J]. Journal of Traffic and Transportation Engineering, 2001, 1(3): 86-88.
- [19] CHEN Jian-kai, GAO Zuo-gang. Urban transport simulation system based shenzhen integrated traffic information services research[J]. Traffic & Transportation, 2013, 1: 62-64.
- [20] 邹智军. 新一代交通仿真技术综述[J]. 系统仿真学报, 2010, 9: 2037-2042.
- ZHOU Zhi-jun. Journal of system simulation[J]. Journal of System Simulation, 2010, 9: 2037-2042.

编辑 叶芳

(上接第 513 页)

- [8] PANG Ya-chuan, GAO Bin, WU dong, et al. A reconfigurable rram physically unclonable function utilizing post-process randomness source with  $<6 \times 10^{-6}$  native bit error rate[C]//2019 IEEE International Solid-State Circuits Conference. San Francisco: IEEE, 2019: 402-403.
- [9] MAHMOODI M R, NILI H, STRUKOV D B. RX-PUF: Low power, dense, reliable, and resilient physically unclonable functions based on analog passive rram crossbar arrays[C]//2018 IEEE Symposium on VLSI Technology. Honolulu: IEEE, 2018: 99-100.
- [10] YANG Jian-guo, LI Xiong, WANG Tao, et al. A physically unclonable function with BER  $< 0.35\%$  for secure chip authentication using write speed variation of RRAM[C]//2018 48th European Solid-State Device Research Conference. Dresden: IEEE, 2018: 54-57.
- [11] WEI Z, KANZAWA Y, ARITA K, et al. Highly reliable TaOx ReRAM and direct evidence of redox reaction mechanism[C]//2008 IEEE International Electron Devices Meeting. San Francisco: IEEE, 2008: 293-296.
- [12] LI Dai, YANG Kai-yuan. A  $562F^2$  physically unclonable function with a zero-overhead stabilization scheme[C]//2019 IEEE International Solid-State Circuits Conference. San Francisco: IEEE, 2019: 400-401.
- [13] WU Meng-yi, YANG T H, CHEN Lun-chun, et al. A PUF scheme using competing oxide rupture with bit error rate approaching zero[C]//2018 IEEE International Solid-State Circuits Conference. San Francisco: IEEE, 2018: 130-132.

编辑 叶芳