支持广义多方案分析仿真的异地克隆研究*

周 云[1],乔海泉[2],黄柯棣[1],胡德文[1]
(1. 国防科技大学 机电工程与自动化学院,湖南 长沙 410073;
2. 北京清河大楼子6,北京 100085)

摘 要:仿真克隆可以提高并行离散事件仿真的效率和并行性,方便快捷地对仿真中的多种可能性进行分析、比较和评估。给出广义多方案分析仿真概念,介绍仿真克隆的相关概念,分析当前仿真克隆方法的不足,提出提高广义多方案分析仿真整体运行效率的异地克隆概念。基于检查点与恢复方法,在动态并行仿真引擎中实现了异地克隆。利用Phold测试程序对异地克隆方法的效率进行测试,测试结果表明,异地克隆可以提高多方案分析仿真系统整体运行效率。

关键词:广义多方案分析仿真;仿真克隆;异地克隆;状态保存;状态恢复;持久化
中图分类号:TP391.9 **文献标志码**:B **文章编号**:1011-2486(2012)04-0074-05

Research on out-place cloning of performing generalized multi-scenario analysis simulation

ZHOUYun[1], *QIAO Haiquan*[2], *HUANG Kedi*[1], *HU Dewen*[1]
(1. College of Mechatronics Engineering and Automation, National University of Defense Technology, Changsha 410073, China;
2. Da Lou Zi, 6, Qinghe, Beijing 100085, China)

Abstract: The efficiency and the parallelism of parallel discrete event simulation can be improved through simulation cloning. Analysis, comparison and evaluation of what-if can be processed quickly and conveniently. Generalized Multi-Scenario Analysis Simulation (GMSAS) concept was given. Based on the introduction of simulation cloning and the analysis of the deficiency on current simulation cloning, the out-place cloning was proposed to advance the efficiency of GMSAS. It was implemented on checkpoint/recover in dynamic parallel simulation engine. An experiment was carried out by using Phold testing program. The results of experimental research demonstrate that the whole efficiency of GMSAS system is improved through out-place cloning.

Key words: generalized multi-scenario analysis simulation; simulation cloning; out-place cloning; state saving; state recovering; persistence

在传统的仿真中,每次仿真运行的输出仅对应一个结果集,当需要对仿真中的多种可能性(what-if)进行分析、比较和评估以作出决策时,就需要针对不同的方案,使用不同的决策规则和参数,多次重复运行仿真。在大规模并行与分布仿真中,仿真模型复杂且数目众多,反复从头重新配置和执行仿真,资源与时间花费巨大。从效率方面来说,这种重复运行的方式对于在线仿真就更不可取。如何进行多方案并发仿真,对提高仿真效率具有重要意义。仿真克隆思想的提出[1-3],为解决该问题找到了一条可行的途径。

本文首先给出广义多方案分析仿真概念,并介绍仿真克隆的相关概念,然后针对当前仿真克隆方法的不足,提出提高广义多方案分析仿真整体运行效率的异地克隆概念,并在动态并行仿真引擎中基于检查点/恢复方法实现异地克隆。最后,利用Phold测试程序对基于检查点/恢复方法实现的异地克隆方法进行测试。

1 问题的提出

1.1 广义多方案分析仿真

作战仿真通常包括三个大类,按其应用领域的不同可分为分析仿真、训练仿真和采办仿真。基于对仿真程序设计与运行的不同影响,又可以将分析仿真分为以下三种情况:

(1)多样本统计。采用概率分布和随机数产生方式对关键决策建模的仿真通常需要大量重复运行同一仿真,利用大量结果统计分析复杂系统的效能,提高仿真结果的准确性。在多样本统计仿真中,各次仿真有相同的输入,仅随机数不同,通过多次仿真减小随机因素的影响,不需要考虑

* **收稿日期**:2012-01-12
基金项目:国家自然科学基金资助项目(61074108,91024030/G30)
作者简介:周云(1965—),女,湖南郴州人,副教授,博士,E-mail: zhouyun8007@126.com

参数变化问题,是最简单的一类。

(2)多参数探索。多参数探索是指在参数的合理离散值或感兴趣的离散值确定的整个空间内运行模型,多次重复运行同一仿真,研究不同参数对问题的影响。在多参数探索仿真中,各次仿真的参数不同。需要在仿真建模时考虑将参数变量化,并在各次仿真中作为数据输入。

(3)多方案分析。利用多假设 what-if 分析探索各种方案可行性的分析仿真需要多次重复运行同一仿真,分析不同关键决策选项下的各种可能,从多种可能性中选择最佳方案。它们经常与指挥自动化系统以及决策支持系统等结合在一起,共同完成作战决策支持一类的应用,提高决策的正确性。多方案分析仿真中,不同的方案不仅可以设置不同的参数,也可设置不同的流程、不同决策等,可能引起模型结构、运行流程的改变,是最复杂的一类。

从软件程序的角度来说,"同一仿真"指的是同一个仿真程序。例如作战仿真中对多种战法方案进行比较分析时,运行同一个仿真程序,仅仅输入的是不同的方案想定。从比较分析的角度来说是不同的仿真,但从研究提高仿真效率的角度来说是没有区别的。因此,从多次重复执行的角度,我们将上述三种分析仿真归结为一种,统称为广义多方案分析仿真,并给出如下定义[4]:

定义 1　广义多方案分析仿真(Generalized Multi-Scenario Analysis Simulation, GMSAS):广义多方案分析仿真是一种需多次重复执行的分析仿真,可用二元组表示如下:

$$\mathrm{GMSAS}::=(S,\ T)$$

其中:

S 表示分析仿真;

T 表示广义多方案分析的层次类型,$T=\{Tmsa,\ Tmpa,\ Tmsc\}$;其中:$Tmsa$ 表示多样本统计,$Tmpa$ 表示多参数探索,$Tmsc$ 表示多方案分析。

广义多方案分析仿真的层次类型不同,其多次重复仿真计算的差异粒度也不同。假设多次重复仿真计算的差异粒度表示为 g,则有:

$$g(Tmsa) < g(Tmpa) < g(Tmsc)$$

传统仿真中,每次仿真运行的输出仅仅对应着一个结果集,进行广义多方案分析仿真时,通常需要多次重复运行仿真,计算量大成为突出问题。仿真克隆思想的提出,为提高广义多方案分析仿真的速度找到了一条可行的途径。

1.2　仿真克隆及其相关概念

美国 Georgia 理工大学的 Hybinette 和 Fujimoto 于 1997 年首先将仿真克隆技术作为一种并发评估机制应用于并行仿真,目的是为利用并行离散事件仿真探索多种可能性方案提供简单有效的支持。

仿真克隆就是在仿真过程中对仿真进行动态拷贝,在内存中创建新的仿真进程,形成一个或多个新的仿真。简而言之,仿真克隆就是创建运行仿真的拷贝。在并行与分布仿真中,通过对逻辑进程的复制实现仿真的克隆。每个克隆所产生的仿真称为仿真的一个版本,原始仿真为版本 1,克隆产生的第一个仿真为版本 2,以此类推[5]。

物理逻辑进程(简记为 P)是指一个具有物理实现的逻辑进程。虚拟逻辑进程(简记为 V) 是指以虚拟形式存在并映射到一个物理逻辑进程的逻辑进程。虚拟逻辑进程映射到物理逻辑进程记为 $V\rightarrow P$[6]。

在支持克隆的仿真环境中,仿真由一组虚拟逻辑进程和一组物理逻辑进程组成。对于具有 A、B、C 三个逻辑进程的原始仿真 Sim^1,可用二元组记为:

$$Sim^1:=(<V_A^1,V_B^1,V_C^1>,<P_A^1,P_B^1,P_C^1>)$$

其中:$V_A^1\rightarrow P_A^1$,$V_B^1\rightarrow P_B^1$,$V_C^1\rightarrow P_C^1$,上标表示仿真的版本,区分不同的仿真;下标表示逻辑进程,区分同一仿真中的不同逻辑进程。

完全克隆:只要到达决策点就对整个仿真内的所有物理逻辑进程进行克隆。

递增克隆:利用虚拟逻辑进程的思想,仅克隆在决策点状态发生变化的物理逻辑进程,状态不受影响的物理逻辑进程以共享方式运行于新旧仿真之中。

仿真克隆从两个方面提高了仿真执行的性能:①克隆得到的多个仿真在克隆动作发生之前共享同一执行路径;②虚拟逻辑进程的思想避免了仿真克隆中不必要的重复计算,使多个仿真能够在决策点之后进一步共享计算。

1.3　当前仿真克隆方法的不足

当前的完全克隆和递增克隆没有考虑多处理器计算机节点的负载平衡问题,存在下列不足:

(1)节点负载不均,延长了运行时间。当前克隆方法克隆的仿真与原仿真在同一组节点上(即同一个物理空间内)同时运行,分享同一组节点的 CPU 时间,过多的克隆仿真聚集于相同的节点,使节点 CPU 的负载过重,将延长原仿真和克隆仿真的运行时间,同时浪费了空闲的节点资源。

(2)物理内存有限,制约了克隆次数。当克隆仿真的次数较多时,由于节点的物理内存有限,可能出现物理内存与磁盘空间的交换,这将极大地降低仿真运行速度。

2 原地克隆和异地克隆

针对当前克隆方法的不足,从负载平衡的角度出发,本文提出原地克隆和异地克隆的概念,在共享相同计算的前提下,既解决节点的负载平衡问题,又解决节点的内存限制问题,进一步提高广义多方案分析仿真的整体运行效率。下面分别给出定义[4]:

定义2 原地克隆(In-place Cloning):克隆仿真并使仿真克隆与原始仿真在同一组计算节点上运行的克隆称为原地克隆。如图1所示,Hybinette 和 Fujimoto 提出并实现的完全克隆和递增克隆方法都属于原地克隆。

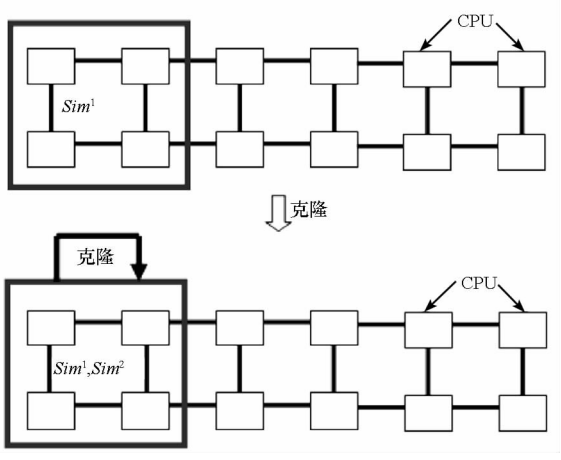

图1 原地克隆示意图

Fig. 1 In-place cloning

定义3 异地克隆(Out-place Cloning):克隆仿真并使仿真克隆迁移至原始仿真所处计算节点之外的计算节点运行的克隆称为异地克隆。如图2所示,异地克隆主要以完全克隆的方式出现。

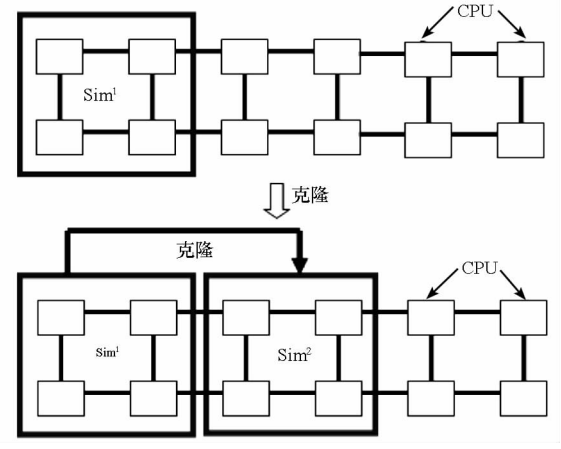

图2 异地克隆示意图

Fig. 2 Out-place cloning

与原地克隆相比,异地克隆方法的优点是:

- 不同方案间的消息被自然分离,不需要物理逻辑进程区分来自不同仿真的物理逻辑进程的消息;
- 允许仿真克隆应用不同的时间管理机制,可以对不同时间推进方式进行并行管理;
- 仿真克隆与原始仿真处于不同的计算节点上运行,可以有效解决负载平衡问题;
- 异地克隆与原地克隆方法中的递增克隆方法有机结合,可以充分提高系统整体运行效率。

3 异地克隆的实现

3.1 异地克隆的实现原理

为使并行仿真引擎支持仿真的异地克隆,必须在并行仿真引擎中增加支持异地克隆的功能。异地克隆方法的实现基于以下原理:仿真程序定义了一组状态变量的集合,这些状态变量按一定的规律随仿真时间的推进而变化,只要状态变量的变化轨迹相同,就说明仿真是相同的。检查点是指保存的某一时刻的仿真状态,恢复是指用检查点恢复仿真的运行。本文使用检查点与恢复方法实现异地克隆,将异地克隆分解为状态保存与迁移、状态恢复与修改2个步骤。其中,状态保存与状态恢复需要持久化方法的支持,迁移地的选择需要负载平衡机制的支持。

3.1.1 状态保存与迁移

并行离散事件仿真系统的状态保存包括仿真对象的状态保存以及事件列表、仿真时钟变量、其他相关数据结构的保存。状态保存与迁移过程由交互进程 ExtProc、仿真管理进程 SimMgrProc 和仿真内核协同完成。

交互进程 ExtProc 周期性地推进时间,通过用户输入、条件判断、定时触发三种方式启动克隆过程。克隆过程启动时,ExtProc 首先解析启动命令,生成相应的“命令事件”列表,并发送给 SimMgrProc,为改变克隆仿真的状态提供信息;然后生成“仿真保存事件”,并发送给 SimMgrProc,通知各节点进行仿真保存与迁移。

仿真管理进程 SimMgrProc 分两步处理“命令事件”列表。克隆之前,原仿真的 SimMgrProc 在自己内部保留一份“命令事件”副本;克隆启动之后,通过处理“启动事件”,将“命令事件”发送到它的目标节点。克隆启动前后之间,SimMgrProc 从“仿真保存事件”中提取信息,传递给内核,并通知内核开始保存仿真。

仿真内核收到克隆通知后,首先生成“启动事件”,并发送给 SimMgrProc;然后利用持久化方

法将所有仿真对象的状态及其目标节点的拓扑结构信息保存在缓冲区中,并将这些数据发送至目标节点。为了保证仿真状态的一致性,仿真保存的前后必须进行仿真同步。

3.1.2　状态恢复与修改

每个迁移的目标节点上都驻留了守护进程,负责接收数据和启动仿真程序。仿真程序启动之后,首先调用持久化方法的接口函数读取数据,恢复仿真状态,并根据节点拓扑结构信息建立节点间的通信链接;然后执行 SimMgrProc 发送过来的"命令事件",改变仿真状态。

3.2　持久化方法的用户接口

仿真持久化即仿真对象的持久化,是指将仿真对象保存在文件、缓冲区或数据库中,可以用这些保存的内容恢复原来的对象。为了在动态并行仿真引擎中增加支持异地克隆的功能,必须为每个需要持久化的对象设计持久化的用户接口,即 C + +实现的序列化接口。本文设计并实现的动态并行仿真引擎的静态数据结构如图 3 所示:

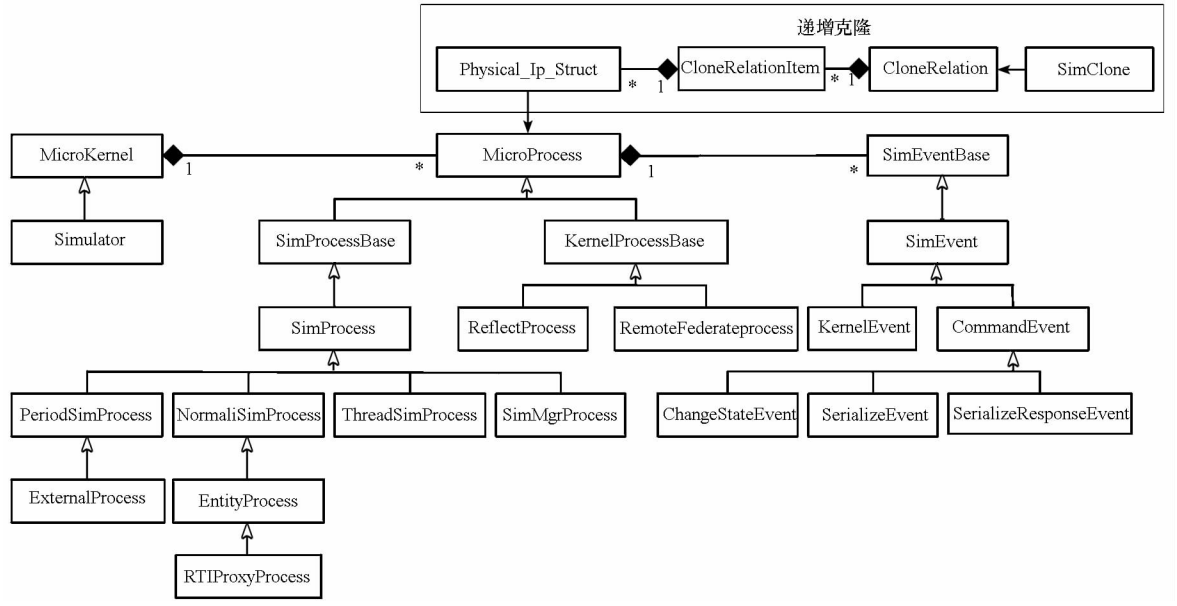

图 3　动态并行仿真引擎的数据结构

Fig. 3　Data structure of the dynamic parallel simulation engine

我们为动态并行仿真引擎数据结构中的每个类设计了持久化方法的用户接口,图 4 所示为仿真管理进程类 SimMgrProcess 的序列化接口实现。其中,save 为序列化接口,load 为反序列化接口。

```
class SimMgrProcess : public SimProcess
{

#if ENABLE_PS
    friend class boost::serialization::access;
    template<class Archive>
    void save(Archive & ar, const unsigned int ) const {
        ar & BOOST_SERIALIZATION_BASE_OBJECT_NVP(SimProcess);
        ar & BOOST_SERIALIZATION_NVP(chgsta_event_list);
        bool is_clone = true;
        ar & boost::serialization::make_nvp("i_am_cloned_sim", is_clone);
    }

    template<class Archive>
    void load(Archive & ar, const unsigned int ) {
        ar & BOOST_SERIALIZATION_BASE_OBJECT_NVP(SimProcess);
        ar & BOOST_SERIALIZATION_NVP(chgsta_event_list);
        ar & BOOST_SERIALIZATION_NVP(i_am_cloned_sim);
    }
    BOOST_SERIALIZATION_SPLIT_MEMBER()
#endif

    ......

};
```

图 4　SimMgrProcess 类的序列化接口

Fig. 4　Serialize interface of SimMgrProcess

3.3　异地克隆与原地克隆的综合集成

基于异地克隆的实现,我们提出异地克隆与原地克隆的综合集成方法,支持广义多方案分析,形成粗粒度下广义多方案分析的异地克隆方式以及细粒度下广义多方案分析的递增原地克隆方式,进一步提高系统效率。

图 5 给出了利用综合集成克隆方法提高分析

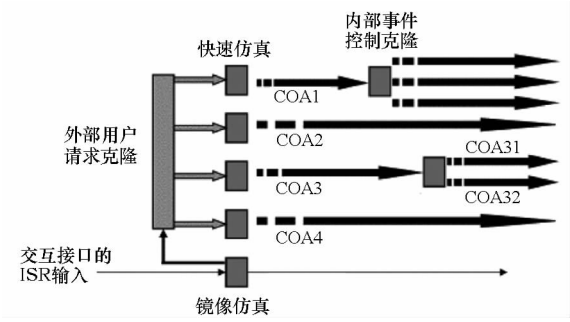

图 5　异地克隆与原地克隆方法的综合集成

Fig. 5　The integration of out-place cloning and in-place cloning

4 种不同行动方案(Course of Action,COA)的广义多方案分析效率的示意。针对 4 种 COA,分别采用

异地克隆方式克隆原始镜像仿真,获得 4 个快速仿真。COA1 中采用原地递增克隆方式产生蒙特卡洛仿真,而 COA3 中的 COA31 和 COA32 仿真是采用异地克隆还是原地递增克隆方式,则视系统负载情况而定。也可以通过人机交互接口,人工选择采用异地克隆方式还是原地克隆方式[7]。

4 测试与结果

为了检验本文研究的异地克隆方法的效率,测试克隆次数和异地克隆的比例对仿真运行效率的影响,本文采用经典的对称负载基准测试模型 Phold 对系统进行测试。

(1)测试环境:配置为 Intel 双核 P8600 @ 2.4GHz 处理器、4G 内存、Win 7 操作系统的 DELL 笔记本一台,配置为 Intel 四核@2.66GHz 处理器、2G 内存、Win XP 操作系统的联想 ThinkCentre M8000T 台式机一台,两者之间使用千兆网络互联。

(2)测试模型与设置:Phold 模型中共有 6 个逻辑进程,平均分配到 2 个仿真节点上。仿真初始化时,每个仿真节点拥有 100 个初始事件,按平均分布从 6 个逻辑进程中随机选择一个作为发送目标。仿真过程中,逻辑进程每收到一个事件就再产生一个新的事件,向随机选择的目标发送。在仿真中的任一时刻,所有仿真节点上当前未处理事件和正在处理事件之和保持不变。仿真时间设为 100s。

(3)测试方法:在笔记本上运行仿真模型,在仿真时间为 T 时,进行 n 次克隆,异地克隆的比例为 r。将其中的 nr 个仿真迁移到台式机上运行,产生 nr 个异地克隆,记录原仿真的运行时间 t。改变 n 和 r,研究 t 的变化趋势和规律。

(4)具体方案:启动仿真,在仿真时间 $T=20$s 时启动仿真克隆,依次进行 0, 2, 4, 6, 8, 10 次克隆,测试 $r=0$、$r=0.5$ 和 $r=1$ 三种情形下的原仿真运行时间。其中,每种情形分别进行 3 次测试,取其平均时间。测试结果如图 6 所示。

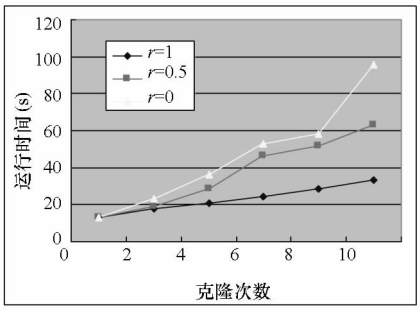

图 6 系统运行效率测试结果

Fig. 6 Efficiency of the system

从图 6 中的测试结果可以看出:①随着克隆次数的增多,克隆需要的开销增大,原仿真的运行时间将增加;②异地克隆数目占总克隆数目的比例越高,原仿真运行时间的增加越少,异地克隆可以有效均衡负载,提高仿真系统整体运行效率。

5 结 论

针对当前仿真克隆方法的不足,本文提出了原地克隆和异地克隆的概念,基于检查点与恢复方法实现了异地克隆方法。与原地克隆相比,异地克隆方法的迁移行为虽然产生了时间延迟,但是,由于异地克隆能够有效均衡负载,合理地使用原地克隆和异地克隆,将使广义多方案分析仿真的整体运行效率得到显著提高。当然,要实现复杂大系统的实时在线广义多方案分析仿真,还需进一步优化综合使用原地克隆和异地克隆的算法,完善动态并行仿真引擎的克隆机制。

参考文献(References)

[1] Hybinette M, Fujimoto M. Cloning: a novel method for interactive parallel simulation[C]// Proceedings of the Winter Simulation Conference, Atlanta, Georgia, USA, 1997: 444-451.

[2] Hybinette M, Fujimoto M. Cloning parallel simulations [J]. ACM Transactions Modeling and Computer Simulation (TOMACS), 2001, 11(1): 378-407.

[3] Hybinette M. Just-in-time cloning[C]// Proceedings of the Eighteenth Workshop on Parallel and Distributed Simulation. Kufstein, Austria, 2004: 45-51.

[4] 周云. 面向实时作战决策支持的动态数据驱动仿真理论和方法研究[D]. 长沙:国防科技大学,2010.
ZHOU Yun. Research on the theory and methods of dynamic data driven simulation for real-time combat decision support[D]. Changsha: National University of Defense Technology, 2010. (in Chinese)

[5] 乔海泉. 并行仿真引擎及其相关技术研究[D]. 工学博士学位论文, 国防科技大学. 2007.
QIAO Haiquan. Research on parallel simulation engine and the relevant techniques [D]. Changsha: National University of Defense Technology, 2007. (in Chinese)

[6] 周云, 韩守鹏, 刘焱, 等. 支持多剧情并发执行的仿真克隆中间件机制研究[J]. 国防科技大学学报, 2008, 30(5): 108-113.
ZHOU Yun, HAN Shoupeng, LIU Yan, et al. Mechanism on simulation cloning middleware of performing concurrent multiple scenarios [J]. Journal of National University of Defense Technology, 2008, 30(5): 108-113. (in Chinese)

[7] Gilmour D A, Hanna J P, Blank H C. Dynamic resource allocation in an HPC environment[C]// Proceedings of the HPCMP Users Group Conference, 2004.