

doi: 10.7690/bgzdh.2023.09.017

# 基于任务状态图的仿真推演引擎设计方法

闫强强, 何伟

(中国人民解放军陆军炮兵防空兵学院研究生大队, 合肥 230031)

**摘要:** 针对仿真推演引擎设计的现实需求, 提出一种基于任务状态图的仿真推演引擎设计方法。设计仿真推演引擎的结构, 构建对任务和规则的形式化描述。结果表明: 该方法能增强模型的重用性, 为方案的推演提供技术支持。

**关键词:** 任务状态图; 规则条件; 仿真推演

**中图分类号:** TP391.9 **文献标志码:** A

## Design Method of Simulation Engine Based on Task State Diagram

Yan Qiangqiang, He Wei

*(Brigade of Graduate, PLA Army Academy of Artillery and Air Defense, Hefei 230031, China)*

**Abstract:** According to the practical requirements of simulation engine design, a design method of simulation engine based on task state diagram is proposed. The structure of the simulation deduction engine is designed, and the formal description of tasks and rules is constructed. The results show that the method enhances the reusability of the model and provides technical support for the scheme deduction.

**Keywords:** task state diagram; rule conditions; simulation deduction

### 0 引言

仿真引擎, 是支撑仿真应用系统、驱动模型运行的核心组件, 为模型提供输入数据、控制模型运行、推进仿真时间, 为各类分析和表现工具提供输出数据和控制信号<sup>[1]</sup>等基本功能, 对系统开发和扩展具有重大影响。根据不同仿真应用系统的需求, 仿真引擎还具备特殊功能。

就军事方案推演仿真系统而言, 通常以实体、任务和交互组织军事推演方案, 仿真引擎应具备识别、解析军事推演方案的特殊功能。一般的推演仿真引擎<sup>[1-5]</sup>只能解析推演方案中“任务”这一层次, 不能定制任务, 反映不出实体执行任务的特殊性。为实现行动任务可定制、体现推演任务的特殊性, 笔者采用面向对象技术、任务状态机和产生式规则, 结合图形化手段, 设计一种基于任务状态图控制仿真实体行动的仿真推演引擎。

### 1 仿真推演引擎设计的需求分析

军事推演方案由多个实体的任务在时间上聚合而成。推演过程中, 一个实体只能同时执行一个任务, 即实体不能并行执行任务。每个实体维护一个任务队列, 并为队列中的任务执行提供上下文环境。

当军事业务逻辑需要并行执行任务时, 将其转换设计成为单任务。推演方案中的任务组织形式如图 1 所示。为体现实体的决策风格, 各任务的执行程序应可定制; 即使对相同任务, 不同实体执行都可按照完全不同结构的程序执行。

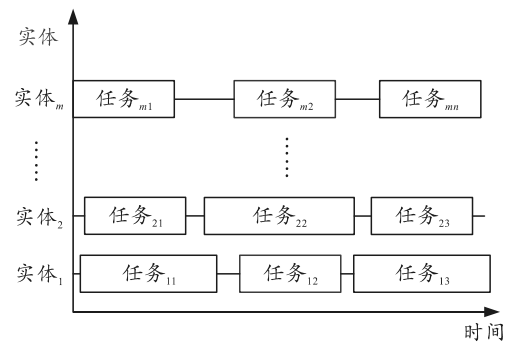


图 1 实体任务的组织方式

仿真引擎的主要工作, 是协调一致地推进各实体执行任务, 包括: 1) 接收用户输入并转换为结构化的任务表示, 向指定实体增加任务, 其来源是用户于推演前预设的计划任务、推演中临时设置或处置情况的随机任务; 2) 从实体任务队列移除任务并调用模型推进相应任务执行; 3) 处理实体间的交互, 构建实体及其任务相互依赖相互影响的综合环境。

收稿日期: 2023-05-07; 修回日期: 2023-06-05

作者简介: 闫强强(1988—), 男, 山东人, 硕士。

## 2 仿真推演引擎的结构设计

仿真推演引擎的核心模块，包括实体管理、数据处理、时间推进管理、环境(地形)处理、事件分发与处理、日志管理、行动规则解析等模块。它依赖于通信服务和数据库服务等公共服务，与导调、推演、显示等子系统密切相关，其结构如图2所示。

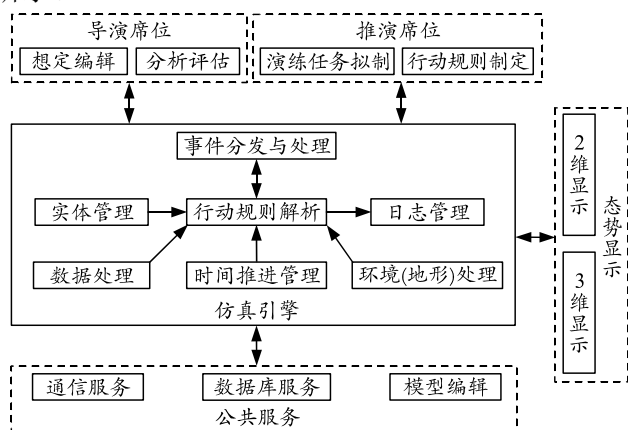


图2 仿真推演引擎的结构

实体管理模块主要负责管理各类实体的实体管理，可控制实体对象的创建及删除，也可管理实体各项配置及有关数据。除了能维护自身状态信息外，还能维护任务列表，是仿真模型执行的环境基础<sup>[6]</sup>。

数据处理模块主要负责管理仿真中的各种数据信息，例如对想定中任务数据进行识别，或将战场环境数据加载到环境(地形)处理模块，还可以统计并存储仿真过程中的数据等。

时间推进管理模块提供时间推进管理功能，在仿真引擎内部，采用统一的仿真逻辑时间，按照一定步长推进仿真时钟，根据实体的运行状态来决定时间推进<sup>[1]</sup>。

环境(地形)处理模块主要负责管理仿真所需的各类天候、地形和战场设备等数据，还可提供接口，进而支持各类实体在任务需要时调用战场数据。

事件分发与处理模块负责管理仿真引擎之间的消息通信，利用特定的消息传输机制实现服务之间的调用和相互通信，确保各模块之间的互联。

日志管理模块主要负责提供日志管理功能，能与态势显示模块进行通信，展示实时态势；也可记录方案推演的过程，进而对数据进行分析或回放。

行动规则解析模块主要接收由事件分发与处理模块递交的仿真任务，调用相关数据与模型进行计算，推进仿真时间，并把仿真结果传递级日志管理模块进行展示和记录。

## 3 基于状态图的任务形式化表示方法

从系统设计的角度来看，用户下达的任务，暗含对知识的抽象和表达，是仿真推演的基本依据。而知识的表示方法，对于用户理解和使用以及系统的设计和实现至关重要。笔者结合产生式规则、状态空间和数学模型3种形式化描述方法，设计了一种基于规则条件和状态图的任务表示方法。

### 3.1 基本概念

1) 任务状态：为保持任务流程良好的可配置性，将任务划分为若干个阶段，每个任务阶段对应一个任务状态。执行某类任务的过程，就是任务在不同状态之间转换的过程，也是从任务初始状态推进到任务结束状态的过程；而且，任务结束状态可以是任务执行成功的状态，也可是任务执行失败的状态。用任务状态图表示任务流程。

2) 规则：概括了在某任务状态内，实体可能面临的情况及其处置对策。一个任务状态内，存在一条或多条作战规则，具体运行哪条规则取决于相应条件是否成立。如果规则条件成立，则按次序执行规则内的若干行动。若存在多条规则的条件成立，则优先运行排序靠前的规则<sup>[7-8]</sup>。随着仿真时间推移和演练态势发展，各规则的成立条件可能会变化。

3) 行动：仿真过程中实体的最小行为单元，具有不可分割性和特定目标性，也称为原子行动。若干个具体的原子行动队列组合成任务，可由各类作战仿真系统按照其内部标准规则完成相关动作过程。

### 3.2 概念的形式化描述

任务表示为：

$$T = \langle S, T_{\text{trans}}, s_0, S_{\text{final}} \rangle。$$

式中： $S$ 为任务状态集合； $T_{\text{trans}}$ 为状态集内不同状态之间的转移集； $s_0$ 为起初状态； $S_{\text{final}}$ 为任务结束状态集。另外， $s_0 \in S$ ； $S_{\text{final}} \subset S$ 。

任务状态表示为：

$$s = (r_1, r_2, \dots, r_m)。$$

每个任务状态由若干规则 $r_i (r_i \in R, R$ 为规则集)有序组合而成。

规则表示为：

$$r = \langle c, A, s_{\text{trans}} \rangle。$$

式中： $c$ 为规则条件； $A$ 为若干原子行动 $(a_1, a_2, \dots, a_n)$ 的有序组合； $s_{\text{trans}}$ 为此规则执行成功后转移到的任务状态；转移集 $T_{\text{trans}} = \{ \langle r, s_{\text{trans}} \rangle | r \in R, s_{\text{trans}} \in S \}$ 。

### 3.3 规则表达方式

军事中的规则，可认为是在一定背景下的战术规则，表示仿真实体针对特定的态势作出特殊反应，具备较强的逻辑关系。参考一般类型作战的特点和事件处置的原则，建立可重用、可扩展的规则库。因此，采用产生式规则表示，其基本形式如下：

```
if 条件1 and 条件2 and ... or 条件m;
then 结论。
```

规则由条件和结论 2 部分构成。条件部分包含一个或者多个前提条件，多个条件通过“与”“或”“非”运算连接起来，通过计算可获取每个条件的布尔值，可用括号改变多个条件的运算优先级；结论部分表示规则的一个动作序列，即只有满足规定的条件，才能产生相应的动作操作列表。因此，规则结构可设计为如图 3 所示。其中，规则条件和行动序列是规则设计的核心。同时，为了保证规则设计的封装性，对外提供条件判断、规则执行及其状态等接口，以便仿真引擎调用。

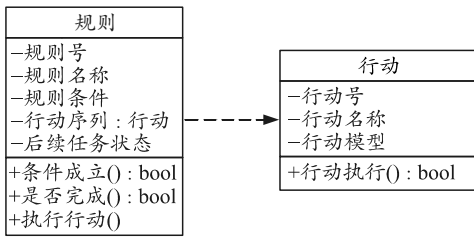


图 3 规则结构定义

### 3.4 条件判断

在进行条件判断时需要 3 个步骤：

#### 1) 构建条件树。

条件常以字符串的形式整体存储，为实时取得其布尔值，需将其从字符串的形式转换成条件树。如将条件“接到冲击命令&&(当前毁伤值≤毁伤阈值)”转换条件树，如图 4 所示。其中，叶子节点是操作数，中间节点为操作符。

接到冲击命令&&(当前毁伤值≤毁伤阈值)

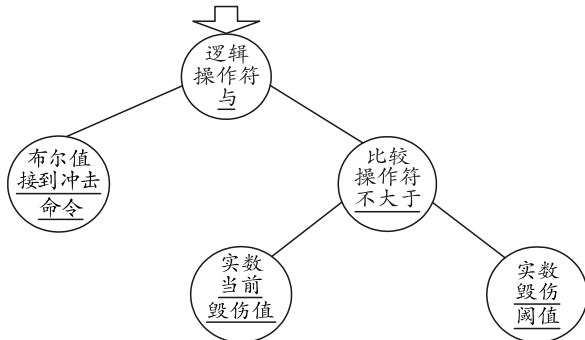


图 4 条件树构建示例

#### 2) 统一操作数类型。

各操作数的原始存储类型为字符串，但其逻辑类型却不一定是字符串，需根据其含义还原成实际类型才能进行计算或比较；因此，按照第 1 个操作数的实际类型，统一 2 个操作数的类型。为简化转换过程的复杂性，不管基本数据类型还是复合数据类型，都必须实现从字符串到相应类型的转换函数，便于程序统一调用。

#### 3) 按照深度优先原则，判定条件树的布尔值。

具体到单个原子条件的判断，先取得各操作数的当前值，再根据操作符进行比较，即得出单原子条件的布尔值，其中操作数有 3 类：① 普通常量，当条件形成时就不再变化；② 任务常量，不同任务具有不同的常量，需要在任务初始时进行初始化；③ 实体变量，与实体的某个属性对应。从条件树最底层的原子条件开始，由下向上不断取求各子树的布尔值，直到得出根节点的布尔值，即整个条件树的值，亦即条件的布尔值。

### 3.5 行动

规则的结论，通常为指示下级实体采取某种行动的命令、请求上级给予支持的请示或情况通报。为便于仿真执行，将实体发出的信息及其执行统称为“行动”，而不仅指狭义的上级对下级的指示<sup>[9]</sup>；因此，规则的结论部分为行动序列。

在系统设计上，行动是成熟的、计算机可执行的数学模型。这类模型已基本形成规模，通常以服务接口的形式，黑盒执行，即给定模型所需参数，经模型执行后得到解算结果通知。根据性质不同，行动区分为时间型行动和任务量型行动。其中，时间型行动是已知时长的行动模型，其结果包括在该时长内所完成的任务量和实体状态；任务量型行动是已知工作量的行动模型，其结果包括完成该工作量所耗时间和实体状态。

### 3.6 任务、状态、规则、条件和行动的组合关系

根据具体的任务需求，组合条件和行动，生成仿真推演系统可模拟仿真的规则，若干规则组合成要素实体的某一个状态，若干状态及其转移组合，生成任务状态，由任务状态和连接线聚合形成任务状态图。其中，某一层次的修改维护，对其他层次的功能影响有限，从而提高了模板的重用性。其组合关系如图 5 所示。

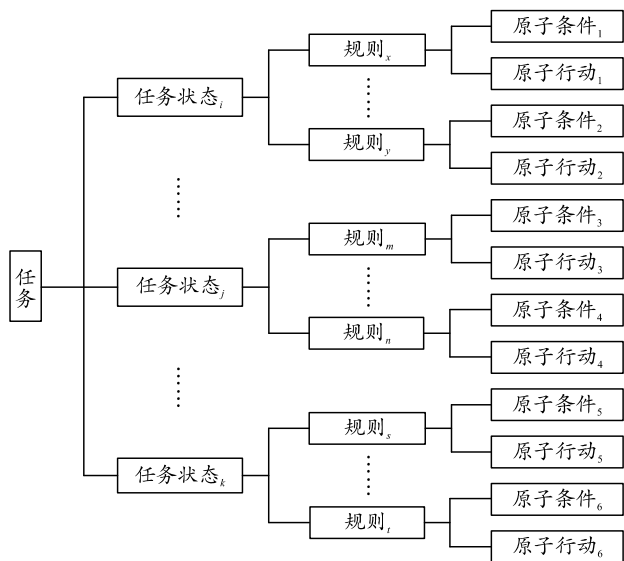


图 5 任务、任务状态、规则、条件和行动的组合关系

#### 4 仿真引擎的任务调度与推演过程

基于任务状态图控制任务流程，其核心思想是以状态图为蓝本、以任务当前状态为基础，结合实体状态、战场态势和环境等影响因素，选择规则、执行相应行动，并根据其结果处理任务状态跳转，实质就是任务在不同状态之间转换的过程，也是从任务初始状态推进到任务结束状态的过程(如图 6 所示)。

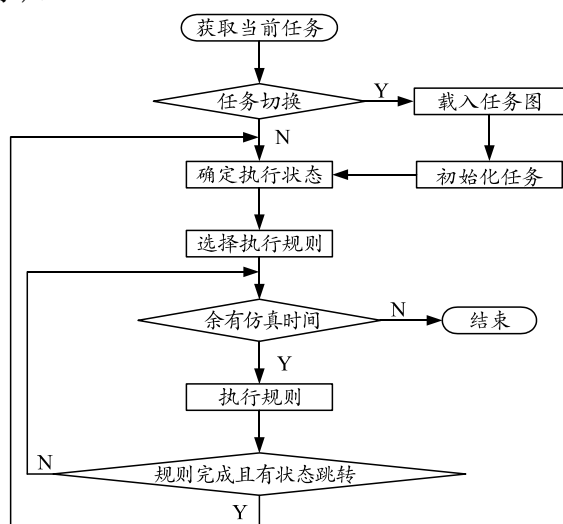


图 6 任务状态调度流程

系统数据库中的行动库和规则库是构建仿真引擎的基础，任务状态图模板存储在规则库中，任务流程的运转依赖于规则条件。具体地讲，一个任务状态内，存在一条或多条作战规则，随着仿真时间推移和演练态势发展，各规则的成立条件可能会变

化，决定着可执行的规则。在规则库与行动库之间通过仿真引擎的解释程序，产生驱动模型运转的输入条件，具体运行哪条规则取决于相应条件是否成立。如果规则条件成立，则按次序执行规则内的若干行动。若存在多条规则的条件成立，则优先运行排序靠前的规则，以此促使任务流程按照合理的方向执行(如图 7 所示)。

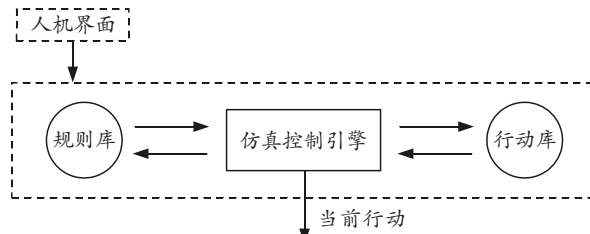


图 7 引擎规则控制方式

#### 5 结束语

笔者构建任务和规则的形式化描述，提出基于任务状态图的仿真引擎设计方法，描述通过引擎完成任务调度解析和推演过程，使实体对象能按任务状态图执行相应行动模型，提高了模型的重用度，解决了行动任务可定制、体现演练任务的特殊性问题，能够为军事方案的推演提供相应的技术支撑。

#### 参考文献：

- [1] 杨雪生, 关永, 张昕, 等. 联合作战仿真引擎分析与设计初探[J]. 军事运筹与系统工程, 2008, 22(3): 8-11.
- [2] 王晨. 基于多核的并行仿真引擎的研究[D]. 北京: 北京理工大学, 2015.
- [3] 彭勇. 作战仿真模型体系分析及其模型设计与实现关键技术研究[D]. 长沙: 国防科学技术大学, 2011.
- [4] 李泽民, 王小振. 基于实体模型的通用作战仿真引擎设计[J]. 电脑知识与技术, 2010, 6(3): 677-680.
- [5] 彭英武, 司光亚, 梅彦铮, 等. 基于HLA的战争分析仿真引擎研究[J]. 装备指挥技术学院学报, 2006, 17(3): 9-13.
- [6] 杨光. 基于面向对象的电子战仿真系统设计与实现[D]. 西安: 西安电子科技大学, 2018.
- [7] 王佳胤, 张宏军, 程凯, 等. 基于有限状态机的作战仿真实体状态转移规则建模[J]. 信息系统工程, 2019(9): 121-123.
- [8] 尹星, 孙鹏, 韩冰. 基于决策树的作战实体行为规则建模[J]. 指挥控制与仿真, 2020, 42(1): 15-19.
- [9] 樊延平, 马亚龙, 柏杰. 基于规则预置的应急事件指挥控制建模研究[J]. 指挥控制与仿真, 2009, 31(3): 9-13.