

doi: 10.7690/bgzd.2016.07.001

面向人为干预的作战计划仿真相关问题研究

王伟¹, 赵晓哲², 王勃³

(1. 海军大连舰艇学院研究生管理大队, 辽宁 大连 116018; 2. 海军大连舰艇学院作战软件与仿真研究所, 辽宁 大连 116018; 3. 中国人民解放军 92189 部队, 辽宁 大连 116021)

摘要: 将人在回路应用到作战计划仿真中, 可达到深入分析问题的目的。从作战计划仿真引入人为干预的需求入手, 在介绍作战计划仿真流程基础上, 对仿真中的可干预内容进行分析, 分别对不同类型的干预内容进行详细阐述, 借鉴程序调试中的断点概念设计不同时机下的人为干预操作方式, 描述加入人为干预后的仿真运行过程。结果表明: 该研究能调整作战计划仿真中的事件及事件处理参数, 为人为干预在作战计划仿真中的应用提供方法。

关键词: 作战计划仿真; 人为干预; 断点

中图分类号: TJ02 **文献标志码:** A

Related Studies on Operation Plan Simulation for Man Intervention

Wang Wei¹, Zhao Xiaozhe², Wang Bo³

(1. *Administrant Brigade of Postgraduate, Dalian Warship Academy of PLA Navy, Dalian 116018, China;*
2. *Operational Software & Simulation Research Institute, Dalian Warship Academy of PLA Navy, Dalian 116018, China;*
3. *No. 92189 Unit of PLA, Dalian 116021, China*)

Abstract: The man in the loop simulation is applied to the operation plan simulation, can achieve the purpose of thorough analysis problems. From operation plan simulation is introduced into the demand of human intervention, on the basis of introduced operation plan simulation process, analysis of simulation can intervene in the content, respectively in detail, and the content of the different types of intervention program debugging breakpoints in the conceptual design for reference under the different timing of intervention operation mode, describe human intervention after the simulation operation process. Results show that the research can adjust events and event processing parameters in the simulation of the operation plan, and provide the method for the application of human intervention in the operation plan simulation.

Keywords: operation plan simulation; man intervention; breakpoint

0 引言

作战计划仿真以作战计划为输入, 以评估结果为输出, 达到对作战计划进行评判的目的。现实中很少有作战过程完全按照作战计划发展的情况。战场环境变化迅速, 需要不断地调整作战计划以适应态势的变化。为了在作战计划仿真过程中, 尽可能帮助熟悉作战行动中的各种情况和应对措施, 将单纯的检验作战计划向深刻理解作战过程转变。为此, 可使决策人员参与到仿真进程中, 实现由仿真前的设置想定到对仿真过程进行临机调整的角色转变。

在仿真进程中加入人的因素, 是一种人在回路的仿真方法。目前, 作战仿真根据用途可分为 3 类: 分析类仿真、训练类仿真和采办类仿真^[1]。人在回路的仿真方法多用于训练类仿真。在仿真的模拟环境中, 人作为受训对象接受训练。作战计划仿真属于分析类仿真, 多采用人不在回路的仿真方法^[2]。把人在回路应用到作战计划仿真中, 其实质就是在原来封闭运行的仿真过程中, 使用人为干预的手段对仿真进程施加影响, 达到深入分析问题的目的。

在此过程中, 参与者的角色能够激发人的创造性思维, 利于更好地分析问题。笔者从人为干预的角度出发, 对作战计划仿真所涉及到的干预内容、干预方式和干预后的仿真运行流程等方面进行分析, 给出作战计划仿真使用人为干预手段的基本思路。

1 作战计划仿真中的可干预内容

1.1 干预内容的分类

作战计划仿真是典型的离散事件仿真^[3-4], 所以笔者从离散事件仿真的角度入手, 分析作战计划仿真中可干预的内容, 将事件调度法作为作战计划仿真的仿真策略。其基本的仿真流程如图 1 所示。

离散事件仿真流程可以简要概括: 建立/更新事件表→时间推进→事件处理(隐含产生新事件)→更新事件表的循环过程。其中, 事件的产生来自于 2 个方面: 1) 系统初始化。系统根据想定对实体、仿真状态初始化, 产生用于建立事件表的初始事件; 2) 事件处理过程。利用相应事件的处理方法对系统状态进行设置, 并可能产生新的事件。

从以上仿真流程可以看出, 可以对事件表中事

收稿日期: 2016-03-26; 修回日期: 2016-04-28

基金项目: 中国博士后科学基金(2014M562557)

作者简介: 王伟(1982—), 男, 内蒙古人, 博士, 助理工程师, 从事水面舰艇建模与仿真研究。

件信息和事件处理过程进行干预。为了后续说明的简便性，将对事件表的干预称为类型 I，对事件处理过程的干预称为类型 II。因为事件处理是推动仿真进程发展的关键环节，所以类型 II 在干预中占据较大的比例。

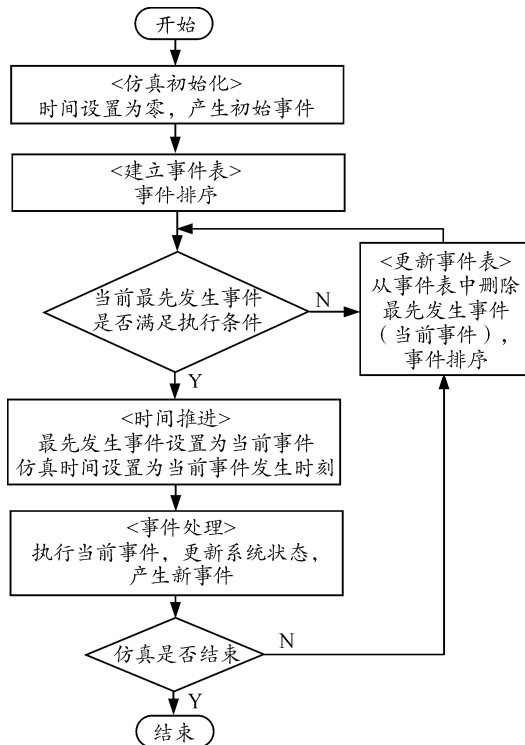


图 1 离散事件仿真流程

1.2 类型 I 的干预内容

事件表是一个存储事件信息的线性表，由仿真系统自动生成，存储的事件信息一般包含事件的类型、事件发生时间和事件的产生者，有时也可以包含其他一些相关信息，但不宜太多^[5]。事件表存储的事件是在当前仿真时间之后发生的，所以事件表也可以叫做未来事件表。类型 I 主要是对事件表进行插入、删除和修改等人为干预操作。可以表示为：

$$TYPE\ I = \{insert(Event), delete(Event), modify(Event)\}$$

其中 Event 表示事件表中的事件信息。Event 的数据结构由事件表的数据模型决定，根据具体实现方式的不同而不同，但基本结构可以表示为四元组：

$$Event = \langle Time, Name, Entity1, Entity2 \rangle$$

其中：Time 表示事件发生的时间；Name 表示事件名称，指明事件的含义；Entity1 表示事件的执行者；Entity2 表示事件的接受者或被影响者。

insert 表示插入操作，即把一个事件插入事件表，相当于在仿真过程中临时加入新的任务。插入操作时，需要人工填写 Event 信息，Time 表示人为确定的事件发生时刻，Name、Entity1、Entity2 从

系统提供的选项中进行选择。插入的事件并非一定能够被执行。因为事件在执行之前，会判断是否满足执行的条件，如事件的执行者是否存在、事件的发生与资源是否存在冲突等。如果不满足执行条件，事件将被忽略。

delete 表示删除操作。将事件从事件表中删除，该事件在仿真中将不再被执行。

modify 表示修改操作。将事件表中事件信息的属性值根据需要进行修改。除人为插入的事件，其他事件都是仿真系统运行时动态生成，事件间具有一定的关联性；所以修改事件信息时，注意不要引起事件之间时间、逻辑上的混乱。

1.3 类型 II 的干预内容

类型 II 是对事件处理过程进行修改。事件处理是在事件表中的事件确定执行后，调用相应的事件处理方法。类型 II 可表示为：

$$TYPE\ II = \{modify(EventParaSet_i) | i=1, \dots, EventNum\}$$

其中：EventParaSet_i 表示与事件相对应的事件处理参数集，不同的事件，在事件处理过程中所使用到的参数也不同；EventNum 表示事件的个数；modify 表示修改操作，在经过对仿真进程的观察，了解战场态势的变化后，可根据新的想法，在事件发生前重新对事件处理过程中的参数进行设置。

为了更好地阐述问题，选取水面舰艇编队作战过程中几类比较典型的事件进行举例说明：

1) 指挥控制类事件。指挥控制类事件是上级对下级进行作战指挥，以及下级对上级指令反馈所对应的事件类型。在仿真进程暂停后，可以修改此类事件中的指挥关系及指令内容。由于上级一般不直接参与具体任务的执行，改变上级对任务的完成影响不大，故较少对其进行修改。通过改变下级，即改变指令的接收者，可将任务重新分配至新的实体。修改后下级实体会根据自身的状态来决定是否执行指令。当仿真继续运行时，新的下级实体代替原有下级实体执行任务。除修改下级实体外，还可修改指令内容，实现对下级任务的重新安排。

2) 武器系统类事件。此类事件的执行者是平台上装备的武器系统。以武器发射事件为例，在发射前仿真系统暂停，可以对与发射相关的状态、属性重新进行设置，如改变武器发射的数量，随后观察修改后打击的效果。

3) 机动类事件。海上编队作战时，舰艇一般都处于运动状态。舰艇起航、返航都属于机动类事件。这类事件涉及航向、航速、航线和活动区域等参数。

以上参数的修改将对后续事件产生影响，如对舰艇航速的修改，会改变接敌时间，进而影响作战事件。

通过以上分析可看出：类型Ⅱ的干预内容直接由干预的事件决定，每个事件的处理过程又有若干参数，所以类型Ⅱ的干预内容十分丰富。在进行类型Ⅱ干预时需要注意，修改的内容应在一定的合理范围内，不能任意设置。

2 基于断点的干预方式

在明确了干预内容后，需采取一定方法来实现干预操作。在仿真进程中进行人为干预，将仿真进程暂停，以便执行干预操作。为使仿真过程在合适的位置暂停，笔者以计算机编程中的“断点”为参考，提出在仿真运行中设置断点，对作战仿真进行分析。断点是程序调试中的术语。设置断点是调试器的功能之一，可让程序在需要的地方中断，方便对程序进行分析；也可在一次调试中设置断点，下一次当程序运行到断点时，便会在断点的位置处停止运行。将仿真进程看作程序调试过程，针对仿真进程设置断点。当仿真运行至断点处时仿真暂停，在干预完成后恢复运行，运行后的系统状态将被人干预所影响。为了对作战计划仿真设置断点，笔者将设置断点的方式分为静态设置和动态设置。

2.1 静态设置

静态设置是在仿真运行前进行断点设置。此时仿真运行过程是未知的，所以不能确定具体的断点位置，而是通过设置断点的触发条件，以达到在适当位置暂停仿真进程的目的。根据作战计划仿真的特点，将触发条件表示为：

$Trigger_condition = \langle state, time, event_condition \rangle$

其中： $state$ 表示仿真系统满足某种状态时暂停； $time$ 是满足特定时间时仿真系统暂停； $event_condition$ 表示与事件相关的触发条件。

$event_condition = \langle name, type, entity \rangle$

$event_condition$ 中的元素以事件数据为参考。

$name$ 表示事件名，当将要执行的事件与 $name$ 具有相同名称时仿真暂停，这是针对某一具体事件进行干预； $type$ 表示事件类型，事件类型是针对一类事件进行干预； $entity$ 表示事件执行者，一旦设置的实体将要执行事件，则仿真过程暂停。需要注意：可供选择的事件执行者是在仿真运行前设置好的实体，仿真中动态产生的实体无法进行设置。以上是基本的触发条件，在实际应用中根据需要进行修改。

2.2 动态设置

动态设置是在仿真运行过程中根据需要设置断

点。直观的方式是通过观察仿真进程的态势变化，在需要使用类似“暂停键”的方式中断仿真，除直接进行人为干预外，还可在事件表中的事件序列前设置断点标志。在后续仿真中，遇有断点标志的事件时仿真暂停。

由于离散事件仿真系统的仿真速度较快，时钟是以事件发生时刻不等间隔地向前推进，从确认需要暂停到具体操作，仿真进程可能已超过原暂停位置。为增加操作的便捷性，将“单步执行”的概念引入仿真系统。“单步执行”是一种程序开发的调试手段，一步一步跟踪程序执行的流程。将单步执行的“步”理解为事件，即按照事件序列依次推进仿真进程，也可称为单事件中断。对处于单步执行过程中的事件来说，相当于在每个事件前都设置了断点标志。单步执行的目的并非使仿真进程在每个事件前暂停，这样会妨碍仿真进程的连贯性，也失去了断点的意义；所以单步执行不能应用于仿真的全过程，而是在适当的时机使仿真系统进入单步执行。从单步执行开始，一个事件周期完毕后仿真将暂停，由人掌握推进到下一事件的时机。在此期间可以设置断点或是直接进行人为干预，最后退出单步执行，恢复仿真的连贯运行。以上操作可重复进行。

无论是静态设置还是动态设置，都是使仿真进程暂停，方便进行人为干预的手段。主要的区别在于：设置操作是在仿真运行前还是仿真运行中。由于设置的时机不同，将静态和动态设置两者结合使用，可以更好地实现人为干预效果。

3 加入人为干预后的仿真流程

在仿真进程中加入人为干预将使原来的仿真流程发生改变。为了在仿真流程中重点突出人为干预的环节，建立如图2所示的仿真运行示意图。

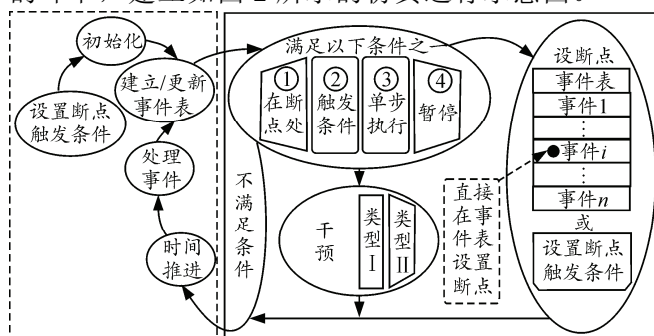


图2 加入人为干预后的仿真运行

- 1) 在仿真开始前，可以通过设置断点触发条件的方式，使仿真运行时在设定的位置暂停。
- 2) 仿真开始后，系统初始化，建立按时间顺序排列的事件表。