

基于改进分层任务网的指控行为模型

孙 林 焦 鹏 许 凯

(国防科学技术大学信息系统与管理学院 长沙 410073)

摘 要 指挥控制行为模型是军事分析仿真系统的主要建模对象之一。通过改进的分层任务网对军事分析仿真系统的指控行为进行建模。指控行为模型由统一的任务描述规范和通用任务管理器组成,可以克服现有模型中存在的建模难度大、模型扩展性弱和使用灵活性差的缺点。最后通过一个空海对抗的实例说明了该指控模型的应用。

关键词 指控行为,军事分析仿真,作战管理语言,分层任务网

中图法分类号 TP391.9 文献标识码 A

Command and Control Behavior Model Based on Improved Hierarchical Task Network

SUN Lin JIAO Peng XU Kai

(College of Information System and Management, National University of Defense Technology, Changsha 410073, China)

Abstract The command and control (C2) behavior model is one of the major modeling objects in military analytical simulation system. The improved hierarchical task network (HTN) was proposed to build the C2 behavior model. It consists of the mission description specification and the general mission manager. The C2 behavior model overcomes the weaknesses of hard modeling, poor expansion and little flexibility in current command and control behavior models. The example of Air-Sea combat illustrates the application of the proposed approach.

Keywords C2 behavior, Military analytical simulation, Battle management language, HTN

1 引言

军事分析仿真系统是支持联合作战辅助决策的重要手段^[1-3]。根据模型功能的不同,可以将军事分析仿真系统的模型分为以下类型:通信模型、武器装备模型、毁伤模型、感知模型、运动模型和指控行为模型。指控行为模型是军事分析仿真系统的核心,发挥着模拟指挥决策行为、协调和运用其他模型的作用。目前,外军多个军事分析仿真系统都建立了以指控行为模型为核心的模型体系^[4-5]。

联合作战的指挥是基于作战任务的指挥,通过作战任务的层层分解与执行来完成上层指挥人员的作战意图^[6]。当前的指控行为模型由于没有采用以任务为核心的指控模型建模框架,因此存在以下不足。

(1)模型建模难度大:不同指控行为模型间相同任务的执行逻辑不能灵活重用,建模人员需要为不同的复杂任务设计相同的子任务管理逻辑,增加了建模负担。

(2)模型可扩展性差:复杂的任务模型开发完成后,用户不能变更作战实体所能执行的任务或任务的执行逻辑。

(3)使用灵活性差:针对具有层次关系的复杂任务,现有的模型不支持系统最终用户自行调整子任务及子任务间的关系。

针对上述不足,设计了基于改进分层任务网的指控行为模型。鉴于现代联合作战对于作战力量快速反应、高效协同和精确打击的需求^[2],军事分析仿真系统的指控行为模型应该能够无歧义地理解 C4ISR 系统或其他仿真系统传递的命

令、报告和请求等信息。联合作战管理语言(Coalition Battle Management Language, CBML)^[7]是由仿真互操作标准组织 SISO 组织制定的一种公开的互操作协议,以支持仿真系统、指挥控制系统和无人作战系统之间传递标准化的信息,其传递的信息主要包括计划、命令、请示和报告等。目前 CBML 已经得到了初步应用^[8-9]。为了保证指控行为模型能够准确理解其他系统的命令,需要为指控行为模型提供基于 CBML 的命令解释机制,使其能够将其他系统下达的命令映射成自身使用的任务描述的数据结构,驱动任务执行。

本文首先设计了基于改进分层任务网的任务描述规范,并给出了基于 CBML 的作战任务数据结构,然后提供了通用的任务管理方法,最后提供了空海对抗的说明实例。

2 基于改进分层任务网的任务描述规范

作战任务具有典型的层次结构,且相同的任务有多种执行方案。分层任务网^[10]是一种合适的作战任务描述方法,其将任务分为两类:基本任务和组合任务,具体定义如下。

定义 1(基本任务) 基本任务指不需要进行分解即可直接执行的作战任务。

定义 2(组合任务) 组合任务指可分解为多个子任务的作战任务。组合任务的结果依赖于分解得到的任务结果及任务间的关系。

由于传统的分层任务网模型对子任务间的约束关系和环境的动态性考虑得较少,难以满足作战任务描述的需求,因此在分层任务网的基础上增加了阶段、关键任务和退出条件 3

本文受国家自然科学基金项目(6147330)资助。

孙 林(1987—),男,博士生,主要研究方向为作战仿真,E-mail: mksl163@126.com。

个概念来增强指控行为模型对作战任务的描述能力,以提高模型的适应性。

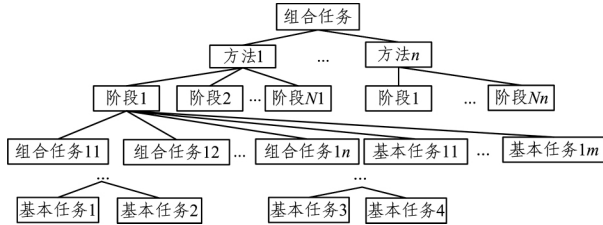
2.1 改进的分层任务网模型及形式化描述

基本任务由一个六元组表示:

$$\text{prim-T} = \{ \text{MissionID}, \text{Attr}, \text{owed phase}, \text{preconditions}, \text{Entity}, \text{Tasker} \}$$

其中, *MissionID* 是任务的标识符; *Attr* 表示任务的属性,如开始时间、任务状态、优先级、是否为关键任务等; *owed phase* 表示任务所属的阶段; *preconditions* 表示执行任务的前提条件; *entity* 表示执行任务的指控实体; *Tasker* 表示下达任务的实体。

组合任务的结构如图 1 所示。



组合任务的形式化描述是在基本任务的基础上增加了一个方法集合 *M* 并去掉了执行条件集合。原因在于同一个组合任务会存在多个前提条件不同的执行方案,任意一个方案被成功执行就认为组合任务被成功执行。将每个执行方案称为方法,对其定义如下。

定义 3(方法) 完成任务的子任务执行序列。

方法用于描述组合任务的子任务集以及子任务间的关联关系^[11-12],其形式化描述如下:

$$\text{Method} = \{ \text{ID}, \text{attr}, \text{phases}, \text{preconditions}, \text{ownedTask} \}$$

其中, *ID* 是方法的标识符; *attr* 表示方法的属性,如状态、优先级等; *phase* 表示方法阶段队列,当队列中的所有阶段都成功执行完毕时,就认为当前的方法被成功执行,反之队列中任意阶段失败,可以判定当前方法失败; *preconditions* 表示方法的执行条件集合; *ownedTask* 表示方法所属的任务。

由于组合任务通常有多种方法,因此在仿真运行过程中,指控实体根据执行条件和优先级选择一种方法执行。执行条件的定义如下。

定义 4(执行条件) 指控实体应用方法完成组合任务的前提条件。只有满足所有执行条件且具有最高优先级的方法才会被执行。

组合任务的子任务间存在 4 种关联关系:顺序关系、并发关系、与关系和或关系^[13]。与关系和或关系表述了子任务的结果对父任务的影响,顺序关系和并发关系描述了子任务的时序关系,为了描述子任务间的顺序关系和并发关系,给出阶段的定义。

定义 5(阶段) 具有并发关系的子任务集合。同一阶段的子任务是并发关系,不同阶段的子任务是顺序关系。指控实体通过阶段的状态变化来控制子任务的时序关系。

阶段的形式化描述如下:

$$\text{phase} = \{ \text{ID}, \text{status}, \text{missions}, \text{exitconditions}, \text{ownedMethod} \}$$

其中, *ID* 是阶段的标识符; *status* 表示阶段状态; *missions* 表示属于阶段的子任务; *exitconditions* 表示退出条件集; *ownedMethod* 表示所属的方法。

由于作战环境具有动态性,存在一些条件使得指控实体不能执行后续的任务,如时间、位置等,因此为了描述作战环境的影响,给出退出条件的定义。

定义 6(退出条件) 指控实体结束当前阶段,并开始执行下一阶段必须满足的条件。退出条件分为充分退出条件、必要退出条件和充要退出条件。

定义 7(充分退出条件) 当退出条件满足时,指控实体必须中止执行当前阶段,并判断当前阶段是否执行成功。

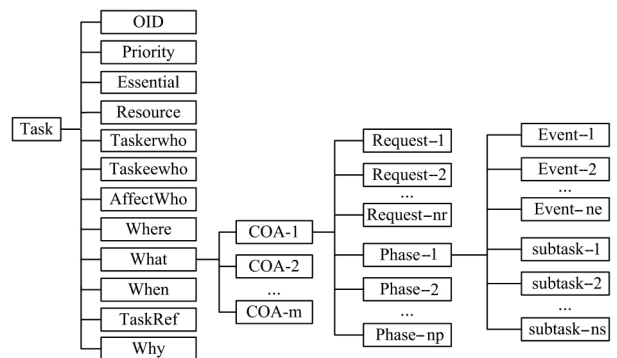
定义 8(必要退出条件) 在退出条件没有满足前,即使阶段中所有的子任务都完成,指控实体也不能结束当前阶段。

充要退出条件同时满足上述定义。阶段的完成情况判定受退出条件和子任务属性两类因素的影响,其判定过程比较复杂,将在下文详细说明。

2.2 基于 CBML 的任务参数描述

在 CBML 中,作战任务是由 5W 原则来指导描述的: Who, When, Where, What 和 Why。Who 表示任务中涉及的作战实体,如任务的下达者和任务的执行者; When 表示任务的时序关系,如任务的开始执行时间、任务的结束时间等; Where 表示任务中的空间关系描述,如运动的路径、目标的区域等; What 表示具体执行的任务或行动; Why 表示任务执行的原因。

在此基础上,结合对任务描述规范的分析可知,在本文提出的任务描述规范中存在一部分信息是目前的 CBML 规范中未涉及的,需要进行扩充,例如,任务的优先级信息在 CBML 中是不涉及的,但是在仿真过程中,任务优先级是指控过程中必须使用的信息。由于组合任务描述的要素包括基本任务的描述要素,因此以组合任务为例设计基于 CBML 的作战任务描述方法。组合任务描述模板如图 2 所示。



其中, *Priority* 和 *Essential* 分别表示任务的优先级和是否为关键任务。由于可以将任务的方法的执行条件看作作战实体对当前战场态势是否符合任务执行需求的一种判断,因此采用 What, Coa, Request 来表示组合任务方法的执行条件。作战实体通过发出 Request 来查询当前态势,并判断是否满足方法的执行条件。任务在各个阶段的退出条件在本质上是预先设置的突发情况,因此采用事件来描述阶段退出条件。

3 通用任务管理器

指控行为模型的可扩展性和使用灵活性主要体现在 3 个方面:1)可以调整作战实体的基本任务;2)可以为作战实体增加新的基本任务;3)可以调整组合任务的子任务集和子任务

间的时序和逻辑关系。

为了满足上述需求,设计了组合任务插件和基本任务插件。在想定编辑阶段,用户为指控实体添加不同的基本任务插件以使其具备执行相应的基本任务的能力。指控行为模型的结构如图 3 所示,每个实体由一个组合任务插件和多个基本任务插件组成。无论是基本任务插件还是组合任务插件,均由两部分组成:任务执行模块和任务参数。每个任务参数代表一个具体的作战任务,同一个任务插件可以同时拥有多个任务参数。任务执行模块负责存储、管理和使用任务参数。组合任务插件的任务执行模块被称为通用任务管理器,为指控行为模型提供了任务状态监控与维护方法。在指控行为模型的开发过程中,建模人员只需设计底层的基本任务模型,而由通用任务管理器提供任务的管理及复杂任务的执行逻辑控制算法,从而降低了模型开发难度。组合任务插件提供了描述任务层次关系的模型框架。在想定编辑中,系统的最终用户可以编辑组合任务的子任务和子任务间的时序关系及逻辑关系。在仿真运行过程中,指控行为模型通过调用组合任务插件和基本任务插件来执行作战任务。由于用户可以根据组合任务描述规范自行修改组合任务的子任务和子任务间的关系,并实现任务的重用,因此组合任务插件增强了指控行为模型的扩展性,提高了模型的灵活性。

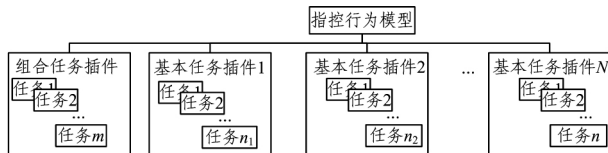


图 3 指控行为模型结构

3.1 任务执行算法

仿真过程中,任务来源于想定、知识库^[14]和通信消息。如果任务源于想定,指控实体会在初始化阶段生成任务执行事件^[15]。仿真引擎在任务开始时自动调用任务执行算法。如果任务源于知识库或通信消息,那么指控实体会生成任务执行事件并将其交由仿真引擎处理。任务执行算法的描述如算法 1 所示。

算法 1 任务执行算法

已知:指控实体 e 开始执行任务 t

1. if t 是基本任务 then
2. 获得与 t 冲突的正在执行的任务集 T'
3. for $t'' \in T'$ do
4. if t 的优先级小于 t'' then
5. 中断 t
6. return
7. end if
8. end for
9. 中断 T' 中所有任务,执行 t
10. else
11. 获得 t 的方法集 M,方法按优先级排序
12. for all $m \in M$ do
13. if m 的所有执行条件都满足 then
14. 从 M 中移出 m
15. 获得 m 的最先执行的阶段 p
16. 生成 p 的所有退出条件实例
17. 获得 p 的任务集 T
18. for all $t' \in T$ do

19. 向 t' 的执行者发送任务执行消息
20. end for
21. 修改 p 的状态为正在执行
22. else if M 中不存在可执行方法 then
23. 执行算法 5,状态为失败
24. end if
25. end if
26. end for
27. end if

如果同时执行的基本任务存在冲突,指控实体将中断低优先级任务。当高优先级的任务完成、取消或失败后,被中断的任务才会恢复执行。

3.2 任务响应算法

任务开始执行后,通用任务管理器需要及时对引起任务状态变化的情况进行响应,有 3 种情况需要处理:1)退出条件满足;2)收到子任务失败消息;3)收到子任务完成消息。任务响应算法如算法 2 所示,阶段完成处理算法如算法 3 所示,阶段失败处理算法如算法 4 所示。

算法 2 任务响应算法

已知:退出条件满足或子任务 t 产生反馈

1. 获得 t 所属的阶段 p
2. if p 的一个充分条件满足 then
3. if p 的所有必要条件都满足 then
4. if p 的所有关键任务都完成 $\vee$ (p 没有关键任务 $\wedge$ p 所有任务都返回结果 $\wedge$ p 有任务成功) then
5. p 完成处理
6. else p 失败处理
7. end if
8. else p 失败处理
9. end if
10. else if p 有失败的关键任务 $\vee$ (p 没有关键任务 $\wedge$ p 所有任务都失败)
11. p 失败处理
12. else if p 的所有必要条件都满足 $\wedge$ (p 没有关键任务 $\vee$ p 的关键任务都完成) $\wedge$ (p 没有关键任务 $\wedge$ p 所有任务都返回结果 $\wedge$ p 有任务成功) then
13. p 完成处理
14. end if
15. end if
16. end if

算法 3 阶段完成处理算法

已知:子任务 t 所属阶段完成

1. 获得 t 所属的阶段 p
2. 向 p 的所有未执行、中断和正在执行的任务的执行者发送任务取消消息
3. 修改 p 状态完成
4. 获得 p 所属的方法 m
5. if m 有未执行的阶段 then
6. 执行 m 中下一阶段
7. else t 的上级任务状态改成完成
8. 向命令执行上级任务的指控实体发送任务完成消息
9. end if

算法 4 阶段失败处理算法

已知:子任务 t 所属阶段失败

1. 获得 t 所属的阶段 p

2. 向 p 的所有未执行、中断和正在执行的任务的执行者发送任务取消消息
3. 获得 p 所属的方法 m
4. 修改 p 和 m 的状态为失败
5. 获得 t 上级任务的方法集 M
6. 从 M 中选择下一方法
7. end if

3.3 任务状态变化处理算法

仿真运行过程中,通用任务管理器能够应对任务状态变化,保证指控实体能够及时对引起任务状态变化的作战环境做出响应。任务状态变成完成、失败和取消时的处理方法如算法 5 所示。

算法 5 任务状态变化处理算法

已知:实体 e 收到任务 t 的状态变化消息、新状态 s

1. if t 是基本任务 then
2. 释放 t 使用的资产
3. t 的状态改成 s
4. 恢复被 t 中断的任务
5. else 获得当前正在执行的方法 m
6. 获得 m 的未执行、中断和正在执行的阶段集合 P
7. forall $p \in P$ do
8. 获得 p 所有未执行、中断和正在执行的任务集 T
9. for all $t' \in T$ do
10. 向 t' 的执行者发送任务取消消息
11. end for
12. 修改 p 的状态为 s

13. end for
14. t 的状态改成 s
15. end if
16. if s 是失败或完成 then
17. 向命令 e 执行 t 的实体发送状态变化消息,新状态为 s
18. end if

4 仿真实例

以空海对抗为实验背景,实验初始设置如表 1 所列。红方指控实体是指挥所及战斗机 1 架,蓝方指控实体是舰艇 1 艘。各指控实体承担的作战任务如表 2 所列,以仿真过程中的时间约束作为任务的执行条件和退出条件。在仿真过程中,通过设置不同的仿真实验初始态势(以任务开始时间和任务执行时间为例)来影响红、蓝双方的对抗结果。仿真实验初始态势如下。

(1)红方:空中进攻任务开始时间为 14 时,侦察任务持续时间为 5 小时,机动任务执行时间为 4.5 小时。

(2)蓝方:对地攻击任务开始时间为 0 时,机动任务执行时间为 8 小时。

表 1 初始设置

所属方	实体	基本任务插件
红方	战斗机	打击任务
		机动任务
	指挥所	侦察任务
蓝方	舰艇	机动任务
		打击任务

表 2 任务设置

实体	组合任务	方法	执行条件	阶段	子任务	充分退出条件	必要退出条件
指挥所	空中进攻	方法一	时间小于 15 小时	阶段一	侦察任务	时间到达 20 时	
				阶段二	机动任务		
				阶段三	打击任务		
		方法二	时间大于 15 小时	阶段一	机动任务	时间到达 20 时	
				阶段二	打击任务		
舰艇	对地攻击			阶段一	机动任务		时间到达 16 时
				阶段二	打击任务		

仿真过程中,作战实体的任务状态变化如图 4 所示。

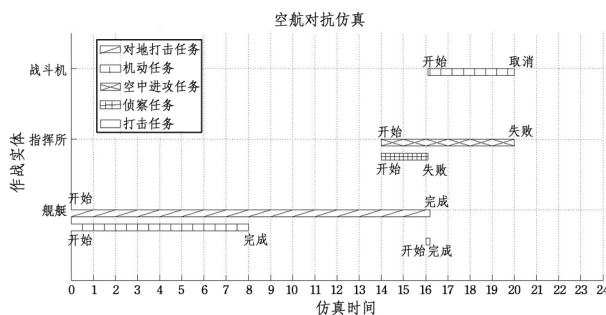


图 4 任务状态图

仿真时间推进到 8 小时,蓝方机动任务结束,但是由于必要退出条件的限制,蓝方维持阶段一,不会执行打击任务。仿真时间推进到 14 小时,指挥所开始执行空中进攻任务,由于方法一的执行条件满足,因此执行方法一的子任务。仿真时间推进到 16 小时,蓝方执行打击任务,红方指挥所损伤导致侦察任务失败,进而导致方法一失败,红方选择方法二执行。仿真时间推进到 20 小时,充分退出条件满足而机动任务未执行完,因此空中进攻任务失败,机动任务取消。

在仿真实验中,由于战斗机和舰艇都要执行打击任务和机动任务,因此为其开发了打击任务插件和机动任务插件。在编辑想定时,通过添加任务插件来使得实体具备执行相应任务的能力。在仿真过程中,指挥所和舰艇执行了不同的组合任务,通过初始输入和仿真结果可知,用户只需要输入子任务和控制条件,模型会自行管理组合任务的执行逻辑。因此,指控行为模型的灵活性和扩展性得到了提高。同时,由于建模人员不开发具体的组合任务,因此建模难度得到了降低。

结束语 基于改进分层任务网的指控行为模型对于军事分析仿真系统具有重要意义。指控行为模型由统一的任务描述规范和通用任务管理器组成。改进的分层任务网模型中增加了对子任务间的时序和逻辑关系的描述,支持用户调整子任务和子任务间的时序及逻辑约束,增强了模型的可扩展性和使用灵活性。通用任务管理器提供了统一的任务管理算法,降低了建模难度。在后续工作中,针对不同类型的指控行为模型进行细化,通过建立多层次多分辨率的指控行为模型,形成更加完善的模型体系。

(下转第 550 页)

参 考 文 献

- [1] LANDAUER R. Irreversibility and heatgeneration in the computing process[J]. IBM Journal of Research and Development, 2000,44(1/2):261-269.
- [2] BARCNO A,BENNETT C,et al. Elementary gates for quantum computation[J]. Physical Review A, 1995, 52(5):3457-3467.
- [3] LI Z Q,LI W Q,CHEN H W. Algorithm of optimizing quantum reversible logic synthesis[J]. Journal of Software, 2009, 20(9):2332-2343.
- [4] LI Z Q,CHEN H W,XU B W,et al. A fast algorithm for synthesis of quantum reversible logic circuits[J]. Chinese Journal of Computers, 2009, 23(7):1291-1303.
- [5] 王冬,陈汉武,安博,等. 基于矩阵初等变换的四量子比特可逆电路综合算法[J]. 电子学报, 2010, 38(11):2561-2565.
- [6] 王冬,陈汉武,安博,等. 量子可逆电路综合的启发式快速匹配算法[J]. 东南大学学报(自然科学版), 2009, 39(5):900-903.
- [7] WANG D,CHEN H W,AN B,et al. A Novel RM-Based Algorithm for Reversible Circuits[C]//LNCS 5821. 2009:63-69.
- [8] WANG D,CHEN H W,ZHU W N. Bidirectional Matrix-based Algorithm for 4-qubit Reversible Logic Circuits Synthesis[C]//2010 IEEE Congress on Evolutionary Computation (IEEE CEC 2010). Barcelona, Spain, 2010:1325-1329.
- [9] 王冬,陈汉武,朱晓宁,等. 多值逻辑量子置换门的酉矩阵表示[J]. 计算机学报, 2012, 35(3):639-644.
- [10] 李志强,陈汉武,徐宝文,等. 基于 Hash 表的量子可逆逻辑电路综合的快速算法[J]. 计算机研究与发展, 2008, 45(12):2162-2171.
- [11] PERES A. Reversible logic and quantum computers[J]. Physical Review A, DEC, 1985, 32(6):3266-3276.
- [12] KHAN M H A. Design of full-adder with reversible gates[C]//Processings of the 5th International Conference on Computer and Information Technology (ICCIT' 2002). Dhaka, Bangladesh, 2002:515-519.
- [13] THAPLIYAL H, SRINIVAS M B. Novel Reversible TSG gate and its application for designing reversible carry look ahead adder and other adder architectures[C]//Processings of the 10th Asia-Pacific Computer System Architecture Conference, LNCS, 3740. Springer-Verlag, Singapore, 2005:775-786.
- [14] HAGHPARAST M, NAVI K. A Novel Reversible BCD Adder For Nanotechnology Based Systems[J]. American Journal of Applied Sciences, 2008, 5(3):282-288.
- [15] HAGHPARAST M, JASSBI S J, NAVI K, et al. Design of a Novel Reversible Multiplier Circuit Using HNG Gate in Nanotechnology[J]. World Applied Sciences Journal, 2008, 3(6):974-978.
- [16] ISLAM M S, RAHMAN M M, BEGUM Z, et al. Low Cost Quantum Realization of Reversible Multiplier Circuit[J]. Information Technology Journal, 2009, 8(2):208-213.
- [17] BANERJEE A, PATHAK A. An analysis of reversible multiplier circuits[J]. arXiv:0907.3357, 2009:1-10.
- [18] CEHNG S T, WANG C Y. Quantum switching and quantum merge sorting [J]. IEEE Transactions on Circuits and Systems, 2006, 53(2):316-325.

(上接第 522 页)

参 考 文 献

- [1] JOHN R S, KEVIN K. Deep Green: Commander's tool for COA's Concept [C] // Proceedings of the 2008 Computing, Communications and Control Technology Conference (DDT). 2008.
- [2] 黄柯棣,赵鑫业,杨山亮,等. 军事分析仿真评估系统关键技术综述[J]. 系统仿真学报, 2012, 24(12):2439-2447.
- [3] 周云,黄教民,黄柯棣. 美国“深绿”计划对指挥控制的影响[J]. 火力与指挥控制, 2013, 38(6):1-5.
- [4] GEORGE F S, GREGORY A M. The Joint Warfare System (JWARS): A Modeling and Analysis Tool for the Defense Department[C]//Proceedings of the 2001 Winter Simulation Conference, 2001.
- [5] JTLS Executive Overview [EB/OL]. [2013-04-12]. http://www.rolands.com/jtls/j_over.html.
- [6] MARK G H, TANIA E R. Using Modeling and Simulation to Represent Commander's Intent [C] // European Simulation. 2008.
- [7] Standard for coalition battle management language (C-BML) phase 1; SISO-STD-011-2014[S]. 2014.
- [8] ADAM B. UK Experiences and Lessons Identified Using C-BML in Practical Experiments [C]//Proceedings of 19th International Command & Control Research & Technology Symposium. 2014.
- [9] PULLEN J M, ABABNEH M, NICKLAS L, et al. An Open Source MSDL/C-BML Interface to VR-Forces [C] // Proceedings of Fall Simulation Interoperability Workshop 2012. 2012.
- [10] KUTLUHAN E, JAMES A H, DANA S N. UMCP: A sound and complete procedure for hierarchical task-network planning [C]//Proceedings of AIPS. 1994:249-254.
- [11] SANTIAGO O, MICHAEL B. Adversarial Hierarchical-Task Network Planning for Complex Real-Time Games [C] // Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence. 2015:1652-1658.
- [12] YIN Q J, SUN L, PENG Y. Research on dynamic task allocation of CGF under communication constraint[C]//Proceedings of IC-CWAMTIP. 2013.
- [13] 冯磊. CGF 协同行为建模关键技术研究[D]. 长沙:国防科技大学研究生院, 2011.
- [14] SUN L, JIAO P, XU K. Rule-based Expert System for Large Scale Analytical Simulation[C]//Proceedings of IEEE ICMA. 2015.
- [15] ZEIGLER B P, PRAEHOFFER H, KIM T G. Theory of Modeling and Simulation(2nd)[M]. New York: John Wiley, 2000.