

多架构体系建模与仿真联合平台^①



杨兆瑞¹, 于翔², 王坚¹, 鲁金直³, 兰小平⁴, 姚春波⁵

¹(电子科技大学 信息与通信学院, 成都 611731)

²(中国电子科技集团公司第二十九研究所, 成都 610036)

³(北京中科蜂巢科技有限公司, 北京 100088)

⁴(兵器工业信息中心, 北京 100094)

⁵(深圳市慧宇系统有限公司, 深圳 154100)

通讯作者: 兰小平, E-mail: xplan@126.com

摘要: 现代作战体系中通常涉及来自多个领域的复杂系统, 因而体系设计时会采用到多个不同的建模工具与仿真工具, 导致体系建模工具之间异构数据难以共享、体系建模工具与仿真工具联动困难等不足. 为解决上述问题, 提出了一种新型的多架构体系建模与仿真联合平台, 形成了“元模型设计-体系建模-仿真验证”的全覆盖设计能力. 在此基础上, 以空地协同防御体系中的人机协同防御场景为例, 验证了该平台对体系设计的有效性. 具体来说, 采用该平台设计了与案例场景相对应的元模型、建立了作战视图体系模型, 并对作战任务中的战机行为进行了仿真, 为战机出战方案设计提供了理论依据.

关键词: 体系设计; 多架构体系建模; 仿真验证; 复杂系统; 作战体系

引用格式: 杨兆瑞, 于翔, 王坚, 鲁金直, 兰小平, 姚春波. 多架构体系建模与仿真联合平台. 计算机系统应用, 2021, 30(12): 63-72. <http://www.c-s-a.org.cn/1003-3254/8226.html>

Systems Modeling and Simulation Joint Platform of Multi-Architecture System

YANG Zhao-Rui¹, YU Xiang², WANG Jian¹, LU Jin-Zhi³, LAN Xiao-Ping⁴, YAO Chun-Bo⁵

¹(School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China)

²(CETC Research Institute 29, Chengdu 610036, China)

³(Beijing ZK Fengchao Technology Co. Ltd., Beijing 100088, China)

⁴(National Defense Technology Industry Maritime Security System Innovation Center, Beijing 100094, China)

⁵(Shenzhen HY System Co. Ltd., Shenzhen 154100, China)

Abstract: Modern combat System of Systems (SoS) generally consists of complex systems across various fields. Accordingly, multiple modeling and simulation tools are used for SoS development. As a result, several problems are caused. For example, heterogeneous data is hard to share among modeling tools, and the collaboration between modeling and simulation tools is difficult. To address these problems, we propose a new multi-architecture SoS modeling and simulation platform with the development capability including meta-model development, SoS modeling, as well as simulation and verification. Then, we demonstrate the platform through a man-aircraft cooperative defense scenario in the air-ground cooperative defense SoS. Specifically, the meta-models for the case scene are designed, and the operational view models are built via the proposed platform. In addition, the simulation of aircraft behavior in combat missions provides a theoretical basis for the trade-off of combat plans.

Key words: system of systems development; multi-architecture system of systems modeling; simulation and verification; complex systems; combat system of systems

① 基金项目: 四川省科技厅重点研发计划 (2020YFG0142)

Foundation item: Key Research and Development Program of Science and Technology Bureau, Sichuan Province (2020YFG0142)

收稿时间: 2021-03-09; 修改时间: 2021-03-31; 采用时间: 2021-04-13

随着现代作战信息化与智能化演进,作战体系需要通信、机械、自动化等多个领域的复杂系统协同工作^[1],如何设计出高效能的作战体系成为各国研究的热点.美国最先于1996年发布了信息通讯指挥系统(Command, Control, Communications, Computers, Intelligence, Surveillance and Reconnaissance, C4ISR)^[2]支持作战体系开发,随后在此框架基础上于2003年颁布了美国国防部体系结构框架(Department of Defense Architectural Framework, DoDAF)^[3]指导美军作战体系设计.基于DoDAF的思想,英国于2005年颁布英国国防部体系结构框架(Ministry of Defence Architecture Framework, MoDAF)^[4]支持英军作战体系开发,北约于2007年发布北约结构框架(NATO Architectural Framework, NAF)^[5]支持作战体系开发.

然而,目前作战体系的设计过程中还存在一些挑战,比如:

(1) 体系建模工具之间异构数据难以共享.作战体系设计涉及到多个不同领域的复杂系统,需要用到多种建模工具,而这些工具建模语言不统一、数据结构不一致、模型组件库不兼容,这使得不同建模工具之间的异构数据难以共享.

(2) 体系建模工具与仿真工具联动困难.目前大多数建模工具只支持建模,缺乏与之配套的仿真工具.对于体系模型的仿真往往需要移植到其他平台中实现,导致体系建模与仿真联合工作难度大.

为解决上述问题,本文提出一种多架构体系建模与仿真联合平台,该平台采用多架构建模语言(Kombination of ARchitecture Model specificAtion, KARMA)^[6]支持元元模型、元模型与体系模型的形式化表达以及体系模型的仿真,实现元模型设计、体系模型设计以及体系模型仿真一体化设计能力.

1 相关工作

近年来,人们对体系设计展开了广泛的研究.体系设计相关研究工作主要分为两大块内容:体系建模和模型仿真.

1.1 体系建模

20世纪90年代后期,体系研究的重要性初步得到认可,体系工程研究开始起步^[7].2009年国际系统工程学会(International Council On Systems Engineering, INCOSE)^[8]提出了基于模型的体系工程(Model-Based

System of Systems Engineering, MBSOSE)概念后,MBSOSE成为了体系工程领域的研究热点.MBSOSE使用图形化的模型支持体系或其系统中要求、设计、分析、验证和确认等活动^[9],不断迭代这些活动完成整个体系的设计开发.MBSOSE主要由建模语言、建模方法和建模工具组成.

MBSOSE的建模语言有很多种,对象管理组织(Object Management Group, OMG)和国际系统工程学会(INCOSE)于2006年提出了统一建模语言(Systems Modeling Language, SysML)^[10]支持体系设计中系统建模,随后又开发了支持DoDAF与MoDAF的统一建模标准(Unified Platform for Defense Modeling, UPDM)^[11],此外还有针对特定领域模型描述的领域建模语言(Domain Specific Modeling Language, DSML)^[12].

MBSOSE的建模方法有很多种,Lutz等人提出了基于agent的体系建模方法^[13];美国国防部提出了基于能力规划(Capability-Based Planning, CBP)的体系建模方法^[14],以能力分析为核心对体系进行建模设计;美军开发了联合能力集成开发系统(Joint Capabilities Integration and Development System, JCIDS)^[15]支持体系建模分析.

1.2 模型验证

模型验证旨在突破体系模型的仿真验证技术,支持体系设计的概念仿真验证.对于这方面的研究工作,Clarke等人提出了Promela^[16]设计约束语言对系统设计进行约束,同时搭配SPIN^[17]验证工具对模型进行验证;瑞典的Uppsala大学与丹麦的Aalborg大学联合开发了一款自动验证工具UPPAALL^[18],实现实时系统的仿真验证;Cimatti等人开发了一款名为NuSMV^[19]的模型验证工具,实现模型符号验证.

尽管有许多的建模语言与建模方法,但多数建模工具不能支持所有的建模语言和建模方法的实现,导致体系建模工具之间异构数据难以共享.同时多数模型仿真验证方法只支持特定的验证工具,导致体系建模工具与仿真工具联动困难.

2 多架构体系建模与仿真联合平台

本文提出多架构体系建模与仿真联合平台,设计概念图如图1所示,该平台具体流程如下:首先基于GOPPRR(Graph, Object, Point, Property, Relationship, Role)元元模型^[20],采用GOPPRR建模方法建立特定

建模语言的元模型库;其次基于元模型库,按照特定体系设计框架进行体系建模;最后针对建立的体系模型,通过混合有限状态机仿真技术对体系行为进行仿真.

整个平台采用 KARMA 多架构建模语言实现,从而支持元元模型、元模型与体系模型的形式化表达与体系模型仿真的无缝衔接.

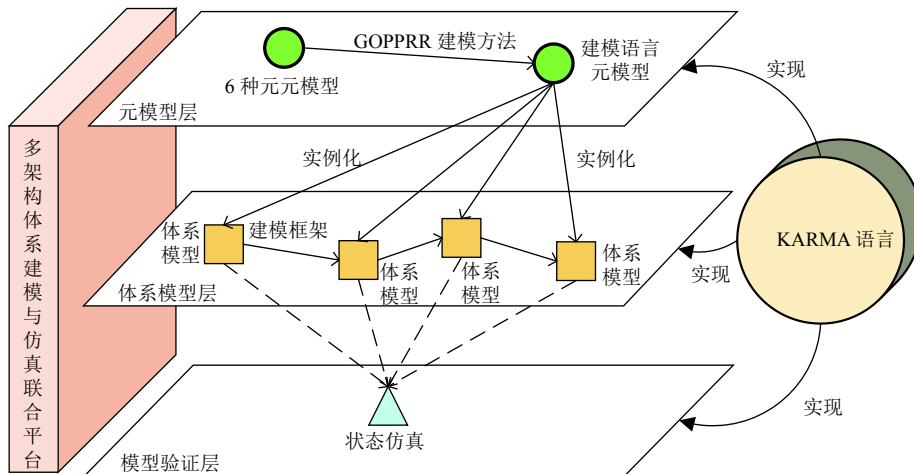


图1 多架构体系建模与仿真联合平台

2.1 模型设计

模型设计涵盖元模型层与体系模型层. 基于 M0-M3 建模层级^[21], 采用 GOPPRR 建模方法^[20] 建立元模型与体系模型. 如图 2 该建模层级将模型分为 4 个层次, 分别是 M3 (元元模型)、M2 (元模型)、M1 (体系模型) 和 M0 (真实视角), 其中 M3 与 M2 层为平台中元模型层, M1 与 M0 层为体系模型层. M3 层包括图、对象、点、属性、关系和角色 6 种元元模型^[17], 首先, 通过元元模型建立 M2 层的不同建模语言的元模型库; 其次, 基于建立的元模型库, 通过实例化元模型建立 M1 层中描述体系视角的模型; 最后, 通过 M1 层中的模型来表达 M0 层中真实体系的不同视角.

2.1.1 M3 元元模型层

M3 层为元元模型层, 代表最高抽象程度的模型, 是对元模型的抽象表达. 6 种元元模型分别是图、对象、属性、点、角色和关系, 其含义如下:

(1) 对象 (object): 可以单独存在的模型. 即可以独立于角色和关系而存在, 也能与其他对象进行连接.

(2) 属性 (property): 不能单独存在, 只能附属在其他元元模型上表达对应参数或特征值.

(3) 关系 (relationship): 用于表达两个对象之间的连接关系, 通过角色和对对象进行连接.

(4) 点 (port): 附属在对对象上的模型, 用于表示对象

上能够与关系连接的接口.

(5) 角色 (role): 存在于连接对象的关系两端的模型, 用来表示对象或对象的点之间连接.

(6) 图 (graph): 图由以上 5 种元元模型组成, 用于描述体系视角的模型.

通过 6 种元元模型的组合, 开发 M2 层特定建模语言的元模型.

2.1.2 M2 元模型层

M2 层为元模型层, 元模型由 6 种元元模型组合而成, 用于描述特定建模语言, 元模型设计流程如下:

(1) 定义属性元模型, 用于描述其他元模型的参数或特征值, 支持其他元模型的引用.

(2) 定义点元模型, 为其添加属性元模型, 并定义其方向, 包括输入、输出和无向 3 种, 支持附属在对对象元模型上, 描述对象上用于连接关系的接口.

(3) 定义对象元模型, 添加对象名称, 并对不同对象元模型添加特有的属性元模型, 同时包含接口的对象需要添加点元模型.

(4) 定义关系元模型, 添加关系名称, 并对不同关系元模型添加其特有的属性元模型, 支持连接两个对象, 描述对象之间的交互关系.

(5) 定义角色元模型, 添加角色名称, 并对不同角色元模型添加其特有的属性元模型, 支持对象与关系

的连接,描述对象的连接方式。

(6) 定义图元模型, 首先, 添加其含有的对象元模

型、关系元模型与角色元模型; 其次, 定义关系与角色的连接; 最后, 定义对象或对象上的点与角色的连接关系。

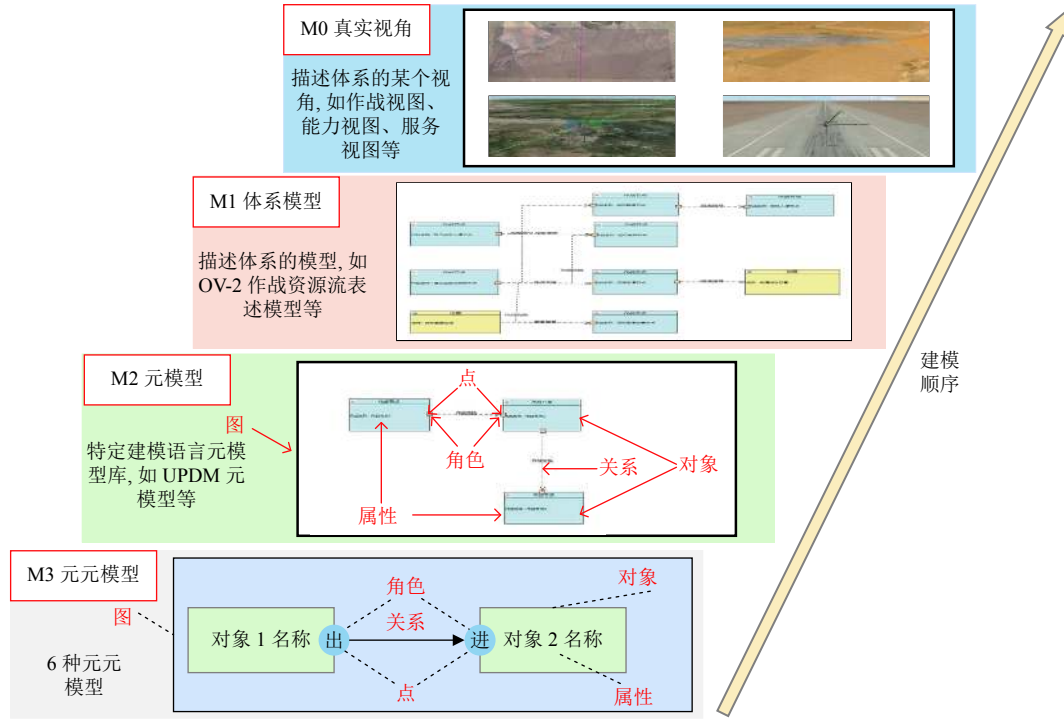


图2 M0-M3 建模层级

本文以 DoDAF 八个视角对应的 UPDM (Unified Platform for Defense Modeling) 标准下的元模型为例, 基于 GOPRR 元元模型相互组合, 建立 UPDM 标准的元模型库, 如对象元模型“作战节点”、属性元模型“作战名称”、点元模型“资源出口”和关系元模型“资源交互”等。

2.1.3 M1 体系模型层

M1 层为体系模型层, 基于特定建模语言的元模型库, 按照特定体系设计框架, 通过实例化 6 类元模型建立描述体系的模型。

以 DoDAF 框架下的作战视图为例, 作战视图所包含的模型图及描述如表 1 所示。基于 UPDM 标准元模型库, 首先实例化对象元模型、属性元模型、点元模型、关系元模型与角色元模型, 其次通过这 5 类元模型的组合实例化出描述体系的图模型, 例如图模型“OV-2 作战资源流描述”。图模型需要根据 DoDAF 框架的作战视图建模顺序进行建立, 作战视图建模顺序如图 3 所示。

表 1 作战视图模型图描述

模型	描述
OV-1: 高层作战概念图	对高层次作战概念的图形描述。
OV-2: 作战资源流描述	对作战活动间的资源交换流的描述。
OV-3: 作战资源流矩阵	对所交换的资源及该交换相关属性的描述。
OV-4: 组织关系图	组织背景、角色以及组织间的其他关系。
OV-5a: 作战活动分解树	以层次结构组织的能力和作战活动。
OV-5b: 作战活动模型	能力和作战活动的背景以及它们与活动、输入和使出间的关系。
OV-6a: 作战规则模型	用于描述作战活动的三模型之一, 标识了作战活动中需要满足的规则。
OV-6b: 状态转移描述	用于描述作战活动的三模型之一, 标识了与事件的业务过程。
OV-6c: 事件跟踪描述	用于描述作战活动的三模型之一, 追踪了一个场景中的各种活动。

2.1.4 M0 真实视角层

M0 层为真实的体系视角, 基于 M1 层建立的体系模型, 建立描述真实体系的不同视角, 如基于 M1 层建立的 OV-1、OV-2、OV-3、OV-4、OV-5b、OV-6a、OV-6b 及 OV-6c 共 8 张模型图, 组成 DoDAF 框架的

视角“作战视图”,支持作战任务使命、环境及资源交互的描述.此外还包括“能力视图”“服务视图”“全景视图”“标准视图”“数信视图”“系统视图”及“项目视图”共8个视角,支持从不同角度描述真实体系.

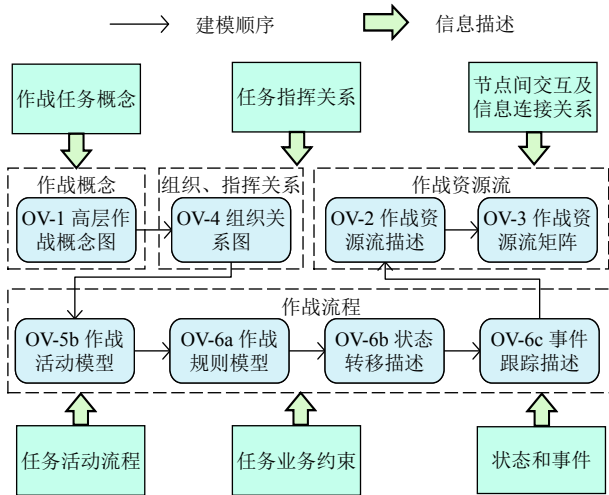


图3 作战视图建模顺序

2.2 模型验证

基于混合有限状态机的状态仿真技术,对描述体系动态行为的模型进行仿真,根据仿真结果对体系行为中的各对象参数与变量进行分析,验证其变化是否满足设计需求,同时支持根据仿真结果进行方案权衡.状态仿真技术的原理图如图4所示.

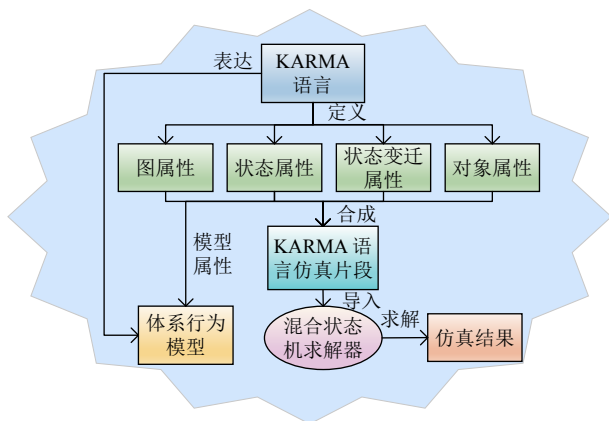


图4 状态仿真原理图

如图4所示,状态仿真技术基于KARMA语言,实现体系行为模型的形式化表达.体系行为模型包含图属性、状态属性、状态变迁属性以及对象属性4种模型属性,其含义如下:

(1) 图属性: 该属性描述状态机图中需要用到的初始变量与参数,如任务场景中的“飞机状态”初始值为“静止”及“雷达开启状态”初始值为“关闭”等,支持描述场景中的各种初始变量与参数.

(2) 状态属性: 该属性描述状态机图中每个状态的属性,包括状态名称,以及该状态下的环境参数,如飞机的“加速”状态下的环境风速参数等,支持描述体系行为中包含的状态.

(3) 状态变迁属性: 该属性描述状态机图中各个状态之间的变迁属性,包括状态之间变迁的条件,如两个对象之间的距离小于100时跳转至下一个状态等,以及变迁时的执行动作,如飞机跳转至“探测”状态时将飞机的探测雷达状态置为“开启”等,支持描述体系行为中对象的状态跳转.

(4) 对象属性: 该属性描述状态机图中的对象的属性,包括对象在整个行为中的参数,如飞机的长、宽、高等,以及参数变化的数学公式,如飞机的速度公式与耗油公式等,支持针对体系行为中对象的各种参数的动态变化进行分析.

首先,通过多架构建模语言定义4种模型属性;其次,基于4种模型属性合成语言仿真片段;最后,将语言仿真片段导入混合状态机求解器中求解得出仿真结果,实现针对描述体系动态行为的模型进行仿真.

3 案例分析

本文案例以空地协同防御体系中的人机协同防御场景为例,使用多架构体系建模与仿真联合平台对场景进行体系设计,验证该平台支持体系设计的有效性.

3.1 案例场景描述

空地协同防御体系中的人机协同防御场景的作战任务概念图如图5所示.敌军入侵我方领地被我方雷达检测信号捕获后,我方雷达控制中心指挥战机出击,寻找敌机并驱逐敌机.随后,战机寻找敌方战车并追踪敌方战车.敌方战车撤离后,雷达控制中心指挥战机前往指定区域收集地面图像信息.最后,战机前往指定区域肃清,保证区域内无敌军后返回基地.

针对作战场景的设计,存在战机的任务所需时间、耗油量及飞行高度等需求指标,设计中需要满足这些指标,如表2所示.本文给出假想的敌方侦察机以及我方几架备选战机的性能指标,如表3所示.

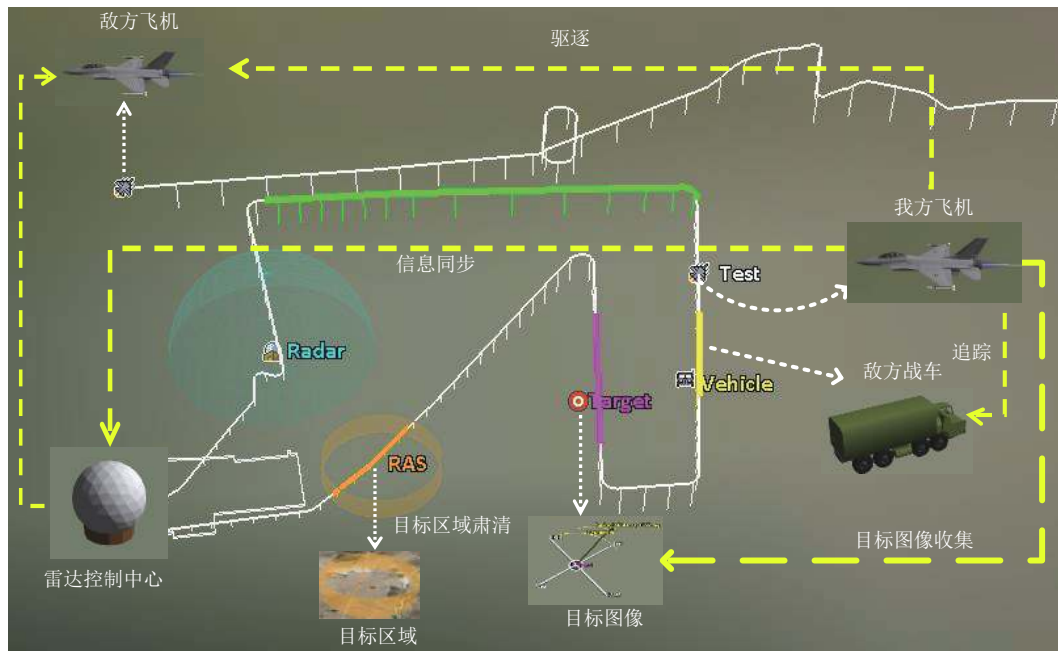


图5 任务概念图

表2 任务需求指标

需求应用于	需求内容	需求指标
	任务所需时间 (min)	<52
战斗机	耗油量 (kg)	<3000
	飞行高度 (m)	>8000 and <20000
雷达控制中心	探测距离 (km)	>1000

表3 战机性能指标

性能指标	敌机	备选战机1	备选战机2	备选战机3
机重(kg)	30600	25000	17000	9750
飞行重量(kg)	77000	37000	36000	15000
推力(kN)	144.57	158.86	129.78	132
最大速度(km/h)	3920	3062	3100	2448
巡航速度(km/h)	3672	2761	2658	2046

3.2 案例实现

首先, 基于 GOPRR 元模型建立 UPDM 标准元模型库; 其次, 基于 UPDM 标准元模型库, 按照 DoDAF 框架建立场景的作战视图模型; 最后, 基于作战视图模型, 通过混合有限状态机仿真技术对作战任务中的战机状态进行仿真, 并根据仿真结果与任务指标进行出击战机方案权衡。

3.2.1 UPDM 标准元模型建立

基于 GOPRR 元模型建立 DoDAF 八个视角对应的 UPDM 标准下的元模型库, 建立的元模型库包含 DoDAF 的 8 个视角共 52 张图所对应的所有 UPDM

标准元模型 (由于 UPDM 元模型库数量过多, 本文只列出 DoDAF 中的 OV-2 对应元模型库供参考), 本文提出的平台以 MagicDraw^[22] 工具中元素为参考进行元模型设计, 建立的 OV-2 元模型如表 4 所示。

3.2.2 UPDM 体系建模

基于 UPDM 标准下的作战视图元模型库, 根据 2.1.3 节介绍的作战视图的建模顺序建立体系模型, 建立人机协同防御场景作战视图模型如图 6 所示。

(1) 首先建立作战概念部分. 建立 OV-1 分析人机协同防御场景任务的高层作战概念图及功能需求, 描述作战任务的概念流程。

(2) 随后建立组织、指挥关系部分. 建立 OV-4 描述作战任务中我方的参与组织结构关系来明确作战中的指挥层级关系。

(3) 明确了作战概念、功能需求和组织结构后建立作战流程部分. 首先建立 OV-5b, 以活动图的方式建立出整个防御场景作战流程以及每个作战活动所需要的触发条件; 其次建立 OV-6a, 给出场景中的我方各要素的性能边界, 以此作为选择作战方案的指标; 最后建立 OV-6b 和 OV-6c, OV-6b 以状态机图的方式描述出各作战节点的状态变化, 描述出作战顺序, 以及作战之间的转换、OV-6c 以序列图的方式, 强调了各作战节点之间的信息交互事件和时间顺序。

(4) 最后建立作战活动间的资源和信息交互关系。分别建立 OV-2 和 OV-3。OV-2 描述了人机协同防御场景作战活动之间的信息交换和依赖关系,反映了作

战节点之间的信息需求,形成信息流动网络。OV-3 具体描述了每个节点之间交互所产生的资源和信息,反映作战节点之间的连通性。

表 4 多架构体系建模与仿真联合平台 OV-2 元模型 (MagicDraw OV-2 元素) 列表对比

对象	功能	关系
条件 (Condition)	功能 (Function)	预期效果 (Desired effect)
执行者 (Performer)	服务访问 (Service access)	执行活动所需能力 (Activity part of capability)
作战活动 (Operational activity)	度量 (Measure)	执行者的能力 (Capability of performer)
系统 (System)	度量类别 (Measure type)	关联 (Association)
软件 (Software)	位置 (Location)	泛化 (Generalization)
能力 (Capability)	位置类 (Location type)	执行 (Activity performed by performer)
能力配置 (Configuration)	使命 (Mission)	作战交换 (Operational exchange)
机构类 (Organization type)	交换元素 (Exchange element)	实现 (Implement)
人员类 (Person Type)	地缘政治范围类 (Geo Political extent type)	资源交互 (Resource interaction)
能量 (Energy)	物资 (Material)	
服务端点 (Service)	作战节点 (Node role)	
请求端点 (Request)		

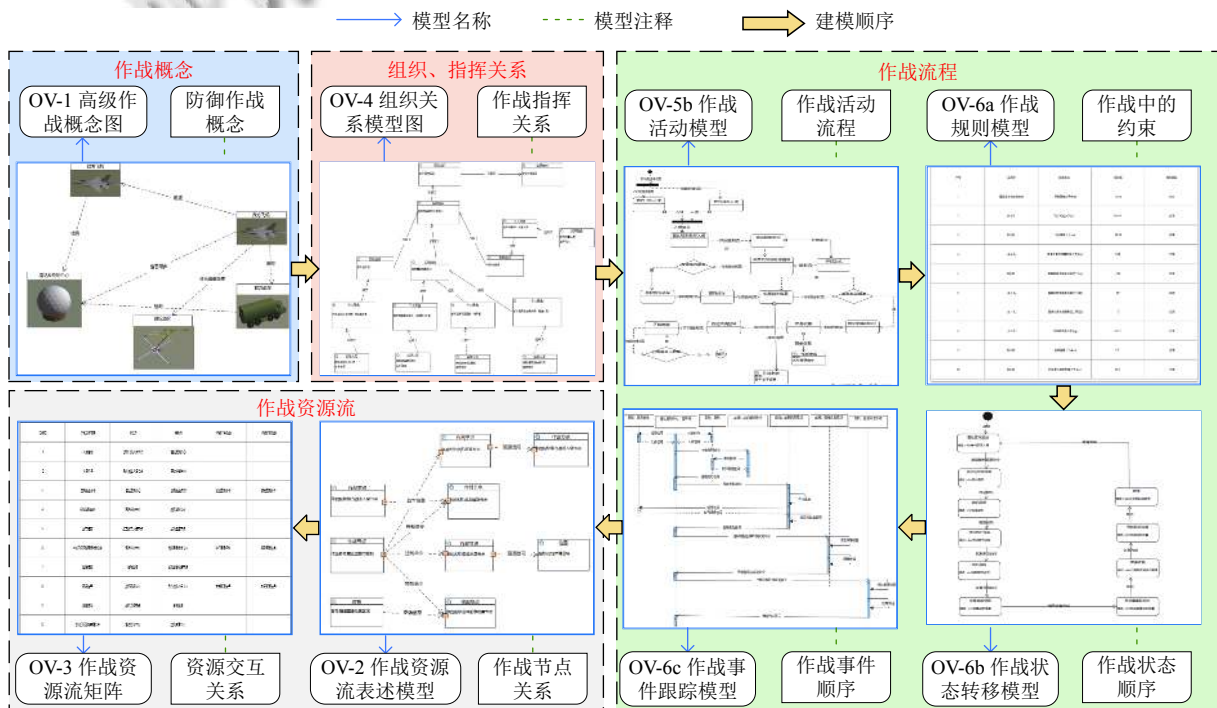


图 6 作战视图模型

基于图 6 中建立的体系模型进行仿真并通过仿真结果对我方战机指标进行评估实现出击战机方案权衡。

3.2.3 状态仿真

本文案例任务中出击战机有 3 架战机备选,存在战机的任务执行正确性、任务所需时间和耗油量这 3 个任务指标进行对比分析,选出最佳出击战机。本文

使用状态仿真技术对建立的体系模型中的状态机图“OV-6b 作战状态转换模型”进行动态仿真。首先根据仿真结果来验证作战任务中战机状态转换流程是否符合预期设计,其次根据仿真结果的战机执行任务时间和耗油量两个参数来对比分析选出最佳出击战机,状态仿真设计顺序图如图 7 所示。

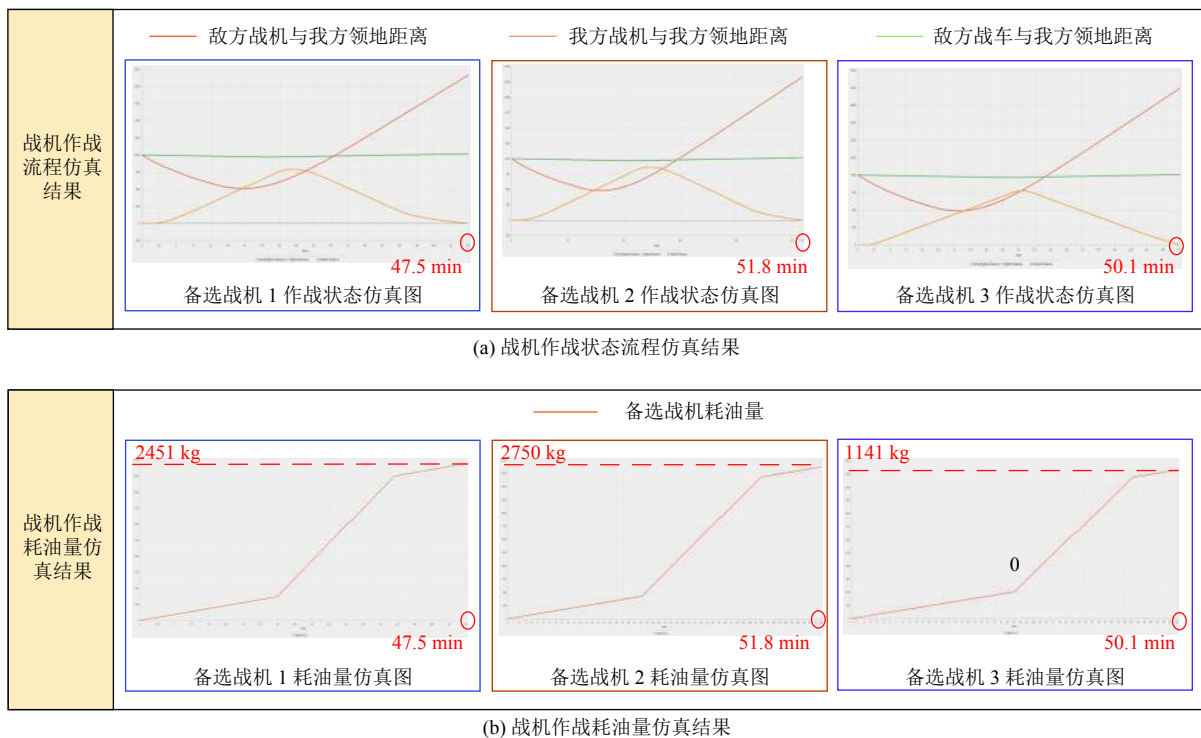


图8 备选战机状态仿真结果

参考文献

- 张毅, 詹武, 郭颖辉, 等. 基于信息流程仿真的作战体系结构设计验证方法. 指挥控制与仿真, 2019, 41(1): 131–135. [doi: 10.3969/j.issn.1673-3819.2019.01.026]
- C4ISR ITF integrated architectures panel. C4ISR architecture framework version 1.0. USA: Department of Defense, 1996.
- DoD Architecture Framework Working Group. DoD architecture framework version 1.0. USA: Department of Defense, 2004.
- Ministry of Defence. MoD Architecture Framework Version 1.0 [Report]. UK: Ministry of Defence, 2005.
- NATO. NATO architecture framework version 3.0. Brussels, Belgium: NATO C3 Board, 2007.
- Lu JZ, Wang GX, Guo JM, *et al.* A general modeling language supporting model-based systems engineering formalisms (Part 1). 30th Annual INCOSE International Symposium. 2020. 323–338.
- 王维平, 朱一凡, 王涛, 等. 体系视野下的 MBSE. 科技导报, 2019, 37(7): 12–21.
- Williamson RC. Model-based systems engineering for systems of systems. *Insight*, 2009, 12(4): 12–14. [doi: 10.1002/inst.200912412]
- 吴颖, 刘俊堂, 郑党党. 基于模型的系统工程技术探析. 航空科学技术, 2015, 26(9): 69–73. [doi: 10.3969/j.issn.1007-5453.2015.09.016]
- Hause M. The SysML modelling language. Proceedings of 15th European Systems Engineering Conference. Edinburgh, Scotland: INCOSE, 2006. 1–12.
- Object Management Group. Unified profile for DoDAF and MODAF (UPDM). <http://www.omg.org/spec/UPDM/2.0/PDF/>. (2012-01-03)[2021-03-01].
- Roques P. Systems Architecture Modeling with the Arcadia Method: A Practical Guide to Capella. London: Elsevier, 2017.
- Acheson P, Dagli C, Kilicay-Ergin N. Model based systems engineering for system of systems using agent-based modeling. *Procedia Computer Science*, 2013, 16: 11–19. [doi: 10.1016/j.procs.2013.01.002]
- National Research Council, Division on Engineering and Physical Sciences, Naval Studies Board, *et al.* Naval Analytical Capabilities: Improving Capabilities-based Planning. Washington, DC: National Academies Press, 2005.
- Sharawi A, Sala-Diakanda SN, Dalton A, *et al.* A distributed simulation approach for modeling and analyzing systems of systems. Proceedings of the 2006 Winter Simulation Conference. Monterey: IEEE, 2006. 1028–1035.
- Clarke EM. Model checking. Ramesh S, Sivakumar G.

- Foundations of Software Technology and Theoretical Computer Science. Berlin: Springer, 1997: 54–56.
- 17 Inverardi P, Muccini H, Pelliccione P. Automated check of architectural models consistency using SPIN. Proceedings 16th Annual International Conference on Automated Software Engineering. San Diego: IEEE, 2001. 346–349.
 - 18 Behrmann G, Larsen KG, Moller O, *et al.* Uppaal-present and future. Proceedings of the 40th IEEE Conference on Decision and Control. Orlando: IEEE, 2001. 2881–2886.
 - 19 Cimatti A, Clarke EM, Giunchiglia F, *et al.* NuSMV: A new symbolic model verifier. Proceedings of the 11th International Conference on Computer Aided Verification. Berlin: Springer. 1999. 495–499.
 - 20 Wang HW, Wang GX, Lu JZ, *et al.* Ontology supporting model-based systems engineering based on a GOPRR approach. Proceedings of the 9th World Conference on Information Systems and Technologies. Cham: Springer. 2019. 426–436.
 - 21 Aßmann U, Zschaler S, Wagner G. Ontologies, meta-models, and the model-driven paradigm. *Ontologies for Software Engineering and Software Technology*. Berlin: Springer. 2006. 249–273.
 - 22 No Magic. “Magicdraw”, <https://www.nomagic.com/products/magicdraw>. [2020-08-06].