

doi: 10.7690/bgzd.2014.06.006

# 基于多智能体的战术仿真系统体系结构设计方法

柳玉, 文家焱

(海军陆战学院科研部, 广州 510430)

**摘要:** 针对传统模拟方法已经无法适应现代战场环境的问题, 提出一种基于多智能体的战术仿真系统体系结构设计方法。以完成单兵种战术模拟演习任务为牵引, 给出体系结构中单个智能体的仿真模型及形式化描述, 引入合作任务管理器实现多个智能体之间的协同通信。实验结果证明: 该方法有效且具有一定的可行性, 能够客观地反映军事演习的整个过程。

**关键词:** 智能体; 多智能体; 模拟训练; 通信协议

**中图分类号:** TJ03 **文献标志码:** A

## Architecture Design of Tactics Simulating System Based on Multi-Agents

Liu Yu, Wen Jiayan

(Department of Scientific &amp; Research, Naval Marine Academy, Guangzhou 510430, China)

**Abstract:** The architecture design of tactics simulating system based on multi-agent is proposed to problems of traditional simulation ways not to adapt to modern battlefield environment. According to the tasks of corps tactics simulation for single arms, the simulation model and formalization of single agent is given and co-task manager is imported to achieve collaborative communication between agents. The test result shows the method is effectively and demonstrates objectively the whole process of the military maneuver.

**Keywords:** agent; multi-agents; simulation training; communication protocol

### 0 引言

仿真技术是继科学理论、科学实验之后的第 3 种认识和改造世界的手段<sup>[1]</sup>。作战模拟(war game)是指通过建立系统、过程、现象和环境模型, 并在一段时间内运行, 用于分析、测试、人员训练和决策支持, 从而揭示作战过程的基本规律<sup>[2]</sup>。随着现代科学技术的不断发展, 战争的规模和形式越来越复杂, 对作战模拟提出了更高的要求, 传统的模拟方法, 如棋戏、沙盘、实兵演练等, 在真实、经济、主动性方面存在一些问题, 已经无法适应现代瞬息多变的战场环境。分布式交互模拟<sup>[3-4]</sup>、智能模拟<sup>[5]</sup>、神经网络模拟和模拟验证<sup>[6]</sup>等新涌现的新技术使得作战模拟的层次更高、应用范围更广。

基于 Agent 的战术模拟系统已有不少研究成果, 主要包括: 1) 侧重于系统的运行环境, 王义冬等<sup>[7]</sup>针对装备作战仿真应用系统, 设计 KQML-RTI 控制器, 通过 HLA/RTI 与 KQML 的集成实现了智能 Agent 之间的通信和合作, 作战 Agent 组件的实现及其组装以及仿真系统的装配; 2) 关注单个 Agent 的体系结构, 李彪等<sup>[8-9]</sup>从需求获取、系统分析和系统设计 3 个阶段阐述了基于 Agent 的作战仿真系统的建模方法, 着重研究作战 Agent 的命令规

则的动态变更机制, 结合可废止逻辑中优越关系的良构性和完备性, 给出了规则动态变更后维护系统无冲突性的具体方法, 并证明了维护方法的有效性; 3) 研究 Multi-Agent 系统在具体应用领域体系结构的实现, 彭辉等<sup>[10]</sup>在城市作战仿真中, 分析 Agent 路径规划过程中需要考虑的战术任务要求, 建立距离代价与战术代价相结合的综合路径评价模型, 设计基于改进概率路标图的战术路径规划方法。

虽然作战模拟、Multi-Agent 系统及其通信等已有大量研究成果, 但目前主要集中在 Agent 技术本身; 然而实际的模拟训练系统并不是孤立存在的, 时刻需要与模型库、环境库、CGF(computer generation forces)库等外部系统协同通信, 需要进一步研究 Multi-Agent 系统的高层框架, 从而消除系统间的异构性, 以便更好地处理各种分布式业务。此外, 改进 Agent 间的协商和交互策略是提高单兵种模拟训练系统性能的一个重要方面。

笔者首先介绍了现有 Agent 系统的种类、特点及发展现状, 通过分析总结战术模拟演习的一般内容、遵循 IEEE1516 相关标准<sup>[11]</sup>, 设计单兵种 Agent 模拟系统; 为了便于各兵种战术训练模拟系统进行综合集成, 采用高层体系结构<sup>[12]</sup>(high-level

收稿日期: 2014-01-09; 修回日期: 2014-02-19

作者简介: 柳玉(1982—), 男, 河南人, 博士, 讲师, 从事军事建模与仿真、Agent 应用技术研究。

architecture, HLA) 的标准进行设计, 确保后期实施子系统间互联、互通、互操作的便捷性, 同时规范地给出了单个智能体的表示方法。

## 1 Multi-Agent 战术仿真系统的体系结构

### 1.1 系统需求

一个基于 Agent 战术模拟系统应满足要求:

1) 位置透明。是指系统应该提供一种寻址机制, 可以动态地定位目标 Agent, 从而对用户屏蔽 Agent 的移动性;

2) 自适应。是指不同的应用会对 Agent 间的协作和通信产生不同的要求, 从而需要系统能够进行相应的调整;

3) 松散耦合。不同业务流程可通过不同 Agent 组合来实现。

### 1.2 Multi-Agent 系统模型

按照军事演习的 6 个一般活动过程(拟制想定、下达任务、作战准备、计划展开、组织实施、导调评估)以及参演兵力等要素之间的关系, 把模型分为 3 层, 分别是用户接口层、业务逻辑层和数据支持层。整个战术仿真系统的运行是靠 3 个不同层次子系统以及单个子系统内部个体之间共同协作来完成的, 见图 1。

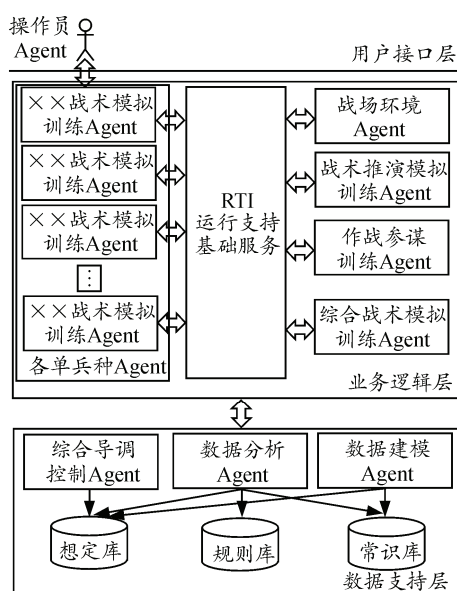


图 1 Multi-Agent 战术仿真系统体系结构

各层按系统内各要素之间的业务关联, 把其中某些功能模块抽象成软件 Agent, 通过 Agent 推理和自适应机制根据环境的变化动态地将相关活动和要素组织起来, 模拟人在回路中的战斗过程, 实现

在作战实验室内进行战术训练、武器操作、装备作战能力评估等, 减少实弹、实装训练带来的经费开支, 提高训练的效果和质量。

#### 1) 用户接口层。

该层主要用于处理与操作员或其他外部应用程序的交互问题, 根据要求传递信息给系统中其他 Agent, 或接收来自系统 Agent 的请求并通过移动 Agent 发送到适当的目的地, 该层的主要 Agent 有:

① 软件界面 Agent: 针对不同类型操作员或不同站位提供不同的显示界面, 同时记录用户在一段时期内的使用习惯和兴趣喜好, 通过分析向用户给出便捷的操作向导和功能接口。

② 移动 Agent: 根据外部环境的变化, 自主地在异构网络上按照一定规则移动, 寻找合适的计算资源、信息资源或软件资源, 完成特定的任务。

③ 信息处理 Agent: 负责接收来自操作员或其他外部应用程序的输入, 并转换成模拟训练系统内通用的数据格式, 供下层 Agent 处理。

#### 2) 业务逻辑层。

业务逻辑层几乎包含了模拟训练系统中所有的业务过程模块, 从军事演习的实施过程来看, 该层是作战实验室实现虚拟战争功能的核心部件。

① 用户 Agent: 是系统中用户持续、固有的表示。在某一时刻, 操作员自身可能不出现在共享工作空间中, 因此需要由用户 Agent 接收相关的消息、并负责传送, 作用于其他 Agent; 在某些情况下, 用户 Agent 可根据战场态势代表用户主动完成某些工作。

② 战场环境 Agent: 综合战场环境是指军事系统所处的包括地形、海洋、大气和空间 4 个领域以及它们之间内部动态的物理环境。战场环境 Agent 主要有 2 部分组成: 环境基本实况和军事系统模型, 前者为后者提供作战的仿真环境, 并为业务逻辑层中其他模拟训练 Agent 提供训练区域、装备性能试验场地以及其他一些战术演练场景。

③ 单兵种模拟训练 Agent: 是对实际的单一兵种计算机军事演习的抽象及封装, 可以在实验室里完成对某一具体兵种针对特定任务的战术演练; 该 Agent 通过运行支持及基础服务平台 Agent 与战场环境、战术推演及底层数据层 Agent 交互, 获得演练所需用的人员、地域、装备、弹药等实体资源, 以及计划组织实施过程。

④ 作战参谋业务应用 Agent: 实际完成具体作

战参谋业务工作的 Agent。它们被指明属于某一个应用，根据应用的需要开始或终止工作；在系统中相互提供和使用服务、相互通信，共同完成某一业务流程。

3) 数据支持层。

数据支持层的主要职责是为战术训练模拟提供各种内容的数据资源服务，主要包括 3 种 Agent：

① 综合导调控制 Agent：依照想定库向单兵种 Agent 传输导调信息，下达导调命令，控制仿真执行进程等功能确保演习顺利进行，在战场环境 Agent 的配合下，使受训者在一种近似实战的较为逼真的训练环境中进行综合有效训练。

② 数据分析 Agent：根据想定库、规则库和常识库向单兵向 Agent 传递模型校正、目标运动、战场环境等辅助参考信息，实现武器装备在特定战场环境下的智能性，增强操作员在实验室内战术模拟训练的沉浸感。

③ 数据建模 Agent：负责从常识库中提取武器弹药的性能指标、结合想定库，生成初始的目标行为规则，系统运行过程中通过与战场环境 Agent 的交互进行修正；同时监察决策支持的反馈结果，更新数据库中的过时信息和数据，并对提取的信息和知识进行完善。

1.3 单兵种 Agent 系统模型

一般来说，兵种战术模拟演习的基本任务是通过模拟演习训练，使受训者理解战术思想和原则、指挥决策原则和方法，体会战术决策的思维方法，提高对战场态势的敏感性，强化其战术意识和战术决策创新能力。根据 IEEE 1516 相关标准要求，设计单兵种战术训练模拟系统的体系结构，见图 2。

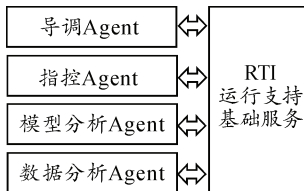


图 2 单兵种 Agent 战术仿真系统体系结构

在图 2 中，导调 Agent 负责训练想定和方案的选择、训练管理、训练过程的监控和态势调理，综合观摩完成战场态势的综合显示和各类战斗行动与结果报告，是综合导调控制 Agent 发送信息的主要接收方。指控 Agent 主要完成模拟训练过程中来自用户界面层转发的各种指挥命令和战场态势，情况报告、环境分析、辅助决策等输出。模型分析 Agent

主要包括针对单一兵种的各型、各类武器装备及其战斗使用模拟模型、战术行动模拟模型、各种战斗效果评估模型等。

在项目实施过程中，需要不断对上述体系结构中的模块进行细化、分解，达到目前现有工程条件、计算机性能可承受的粒度，下面给出抽象后兵种战术仿真系统中的单个 Agent 模型。

1) Agent 的框架结构。

根据基于人工生命作战仿真方法的特点，在行为主义人工智能的基础上，引入一种基于目标驱动的具有内部状态的混合 Agent 结构<sup>[13]</sup>，内部状态作为历史信息与当前感知共同决定下一步的动作，使 Agent 的反应更具理性；同时通过对目标完成程度的合理评价，实现 Agent 的自适应，见图 3。

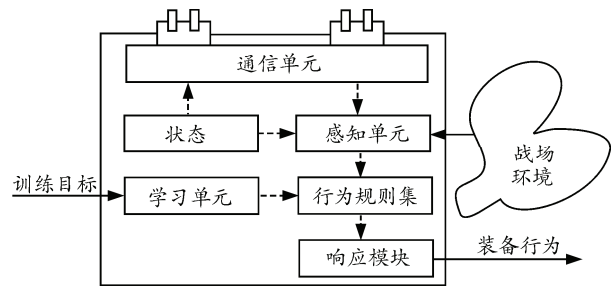


图 3 具有内部状态的混合 Agent 模型

从图 3 可以看出，兵种战术训练模拟系统中一个具体的 Agent 主要包括感知、通信、动作规则、学习、响应等几个组件，包含自身状态，同时通过与战场环境的交互对目标行为产生影响。

2) Agent 的形式化表示。

根据上述结构，参照 BDI 模型，将一个战术单元的 Agent 表示为三元组：

$$\begin{aligned}
 \text{Agent} &::= \langle b, d, i \rangle \\
 b &::= \langle \text{EI}, \text{EE} \rangle \\
 d &::= \langle se, de \rangle \\
 i &::= \langle \text{RULE}, \text{SA}, \text{GO} \rangle \\
 se &::= \text{EE} \rightarrow \text{SA} \ \&\& \ \text{Agent} \rightarrow \text{SA} \\
 de &::= \langle ac, le \rangle
 \end{aligned}$$

其中：*b* 为信念，代表了系统的规则知识、状态和发生的事件；*d* 为愿望，体现为各种潜在的可能战术行为规划，代表行为规划问题求解操作推理；*i* 为意图，体现为决定实施的计划行为，代表策略集合；其他参数具体含义如下。

EI：战场环境即仿真系统运行的内部环境，包含模拟训练所涉及区域中的已知威胁、地形、敌我双方目标等；

EE：外部环境，即其他 Agent 的状态；

*se*: 表示 Agent 感知环境能力的函数, 包含 2 部分:  $EE \rightarrow SA$ , 是指 Agent 自身对战场环境的反应, 表现在其状态集 SA 上;  $Agent \rightarrow SA$ , 是指其他 Agent 当前状态对自身的影响;

*de*: 表示 Agent 自适应能力的函数, 其中, *ac* 表示决策过程函数, 输入为当前感信息、自身状态及决策规则集, 即  $EI \&\& EE \&\& SA \&\& RULE$ , *le* 表示学习过程函数, 具体实现 Agent 的泛化性功能;

RULE: 规则集; SA: 表示 Agent 的状态集合;  
GO: 表示 Agent 待完成的任务集。

### 3) Agent 的推理决策模型。

推理与决策是 Agent 的内在行为, 体现了 Agent 的智能程度。为提高推理速度, 放弃传统通过定义任务和进行多目标优化的方法, 简化为设计 1 个单目标函数进行最优逼近。在具体应用时, 通过分析一般可以找出与完成任务和目标相对应的度量方式, 可将该度量稍作修改成性能函数。

令  $IS$  为 1 个 Agent 集合,  $u$  为自定义的性能函数, 则一定有以下式成立:

$$(\exists A \in IS \wedge (\forall G \in IS)(u(G) \leq u(A)))$$

即  $A$  为满足当性能目标的最优函数。

在此基础上, 可给出学习过程函数 *le* 的定义:

$$Le = IS \rightarrow A$$

具体实现时, 需要从当前规则库中找到环境状态与动作交替的有限序列, 使得 Agent 按照该规则库决策行为时, 性能函数  $u$  达到最优值。

## 2 Multi-Agent 战术仿真系统的协同通信

### 2.1 通信机制

在分布式仿真环境中, 需要底层网络通信模块的支持, 同时要实现 Agent 之间的交互, 发送方需要知道接收方的 ID, 位置以及功能等, 笔者通过建立基础服务、全局服务和局部服务 Agent 为战术仿真系统提供高效、透明的通信服务。

全局服务 Agent 提供全局唯一的 Agent 标识, 管理分布仿真中 Agent 的功能、位置等信息, 其他 Agent 可通过全局服务 Agent 查询其他 Agent 的物理位置、功能等, 监控分布仿真中结点的工作状态; 局部服务 Agent 在每个处理结点上且有且只有一个局部服务 Agent, 负责给本结点内部的 Agent 提供服务, 包括 Agent 注册, 每个 Agent 加入到系统中之前, 需要获得本 Agent 在系统中的唯一 ID 号, 同时, 将本 Agent 的位置和功能进行注册, 以获得标识。

RTI 是 HLA 体系结构中接口规范的软件实现, 为仿真运行提供必要的服务, 同时也成为智能 Agent 通信的基础, 基础服务 Agent 直接使用 HLA/RTI 提供的公共服务接口。

### 1) 通信模型。

KQML 是 Agent 之间通信的基本协议, 已经成为 Agent 之间通信的事实标准, Agent 通过 KQML 发送消息来宣布命令、通知事件发生、查询事件结果等。一个联邦成员可能没有 Agent 对象, 也可能有一个或多个; 因此, 需要对原有 Multi-Agent 系统模型的业务逻辑层进行修改, 在存在 Agent 的联邦成员和 RTI 之间嵌入 KQML 控制层, 见图 4。

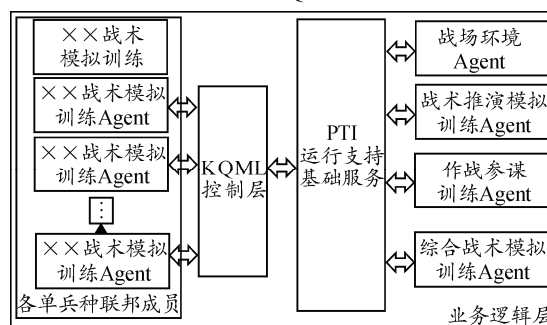


图 4 Multi-Agent 系统的通信模型

KQML 控制层为战术训练模拟系统联邦成员中所有 Agent 对象之间通信和合作提供支持。输入的消息通过控制层进行解析, 输出的消息在这一层编码为联邦中使用的 KQML 交互格式消息。对于任何要加入联邦执行的智能 Agent, 只需对它的通信接口进行修改, 而不需要对消息的格式进行任何改动, 从而大大提高了基于 Agent 的模型对于 HLA 体系结构的适应性<sup>[14-16]</sup>。

### 2) 通信过程。

Agent 间的消息格式采用标准 KQML 语言, 并通过 XML (Extensible markup language) 加以封装, 便于网络的高效传输和扩展, 具体实现过程如下:

Step 1: 初始化, Agent 通过局部服务 Agent 向全局服务 Agent 注册, 获得全局唯一 ID;

Step 2: Agent 登记其所能够完成的功能;

Step 3: Agent 发送 KQML 行为元语后, 通信控制层首先查询此元语的目的地 Agent 的 ID 及其位置信息, 如果不属于本联邦成员, 则转向 Step 6, 否则继续;

Step 4: 将此元语传送到指定 ID 的 Agent;

Step 5: 转向 Step 9;

Step 6: 通信控制层对待传输的行为元语进行相

应的交互编码；

Step 7: 通过局部服务 Agent 发送到目的联邦成员；

Step 8: 目的联邦成员收到此交互后，经 KQML 解析后传递给相应的 Agent；

Step 9: 通信结束。

### 2.2 协作模型

在单兵种模拟训练系统中设立子任务管理器 S-CoAgent，在导调台设立总任务管理器 T-CoAgent，以便了解所有 Agent 成员和环境的状态、识别和协调各 Agent 成员之间的观点和意图的不一致和冲突，保证 Agent 的局部决策或行为与整体目标相适应，并避免损害其他 Agent 成员的正常活动。

为保证 CoAgent 不会束缚系统资源的有效利用、避免系统行为的不稳定性对任务进行分解时，要考虑各子任务之间的关联性和各自的独立性<sup>[12]</sup>。根据分布式计算和多 Agent 协同工作的特点，在进行任务分解时要遵循以下原则：

- 1) 独立性，生成的子任务尽量相互独立，减小 Agent 之间的耦合度。
- 2) 层次性，任务划分应根据完成目标的要求来进行，从高层到低层、复杂到简单依次进行，直到单兵种战术模拟 Agent 能够顺利完成任务为止。
- 3) 均衡性，尽量使生成的子任务集合在规模和难易程度上保持相当，避免各子任务在执行过程中因时间悬殊导致系统整体效性的下降。

设  $P$  表示待完成的任务，经 CoAgent 分解后得到子任务序列  $P_1, P_2, \dots, P_n$ ，对任意的  $i, j \in [1 \dots n]$  且  $i \neq j$ ， $P_i, P_j$  存在 3 种时序关系，分别是顺序、并行、互斥，见图 5。

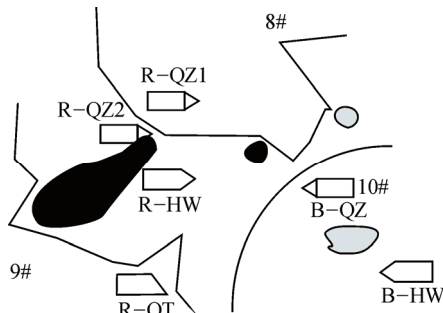


图 5 作战想定

### 3 应用实例

作战想定：我方观通站发现敌方一个舰艇编队 (1 艘基德级驱逐舰 B-QZ, 1 艘诺克斯级护卫舰

B-HW) 向我方 10 号海域运动，上级要求我方贴近该海域的一个战术群进行拦截。1) 兵力编成，红方：一个舰艇编队战术群，包括 2 艘驱逐舰 R-QZ1、R-QZ2，1 艘护卫舰 R-HW，一艘常规潜艇 R-CQ，其中，R-QZ1 为指挥节点；2) 组织计划，R-QZ1 抄近道穿越航道危险的 8 号海域，R-QZ2 从 9 号海域继续追击，R-QZ1 穿越 8 号海域后，由 R-HW、R-CQ 负责接应，R-HW 与 R-CQ 相互协同作战，直到重创敌方；3) 战场态势如图 5 所示。

通过分析发现，上述参予的所有战斗实体都是智能 Agent 实体，其中，甲方 R-QZ2、R-HW 及 R-CQ 属于同一联邦成员。R-QZ1 通过自身的感知以及另外 2 个友邻 (侦察机) 上报的态势进行决策，向另外 2 个编队成员下达命令，实现战场情报、知识的共享以及战斗的协调。模拟系统运行于上述支持 Multi-Agent 的 HLA/RTI 仿真环境中，系统结构如图 6 所示。

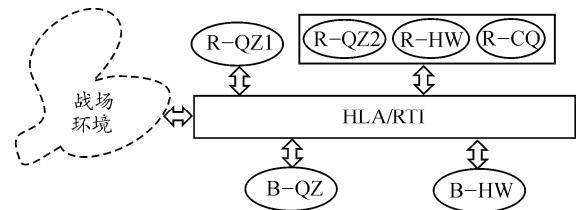


图 6 原型系统 Agent 结构

在本系统中，编队成员对战场知识和并发联合行为采用规范一致的描述，甲方的 3 个 Agent 都面向特定目标，即击沉或重创敌编队，Agent 采用规则演绎的推理机制，利用通信 Agent 上报战场态势，同时通过 KQML 语言进行知识共享、计划和行动协调。而乙方编队属于反应式 Agent，对感知到的战场环境做出应急反应，选择火力射程范围内的目标进行反击。图 7 是某一时刻仿真系统的战场态势。

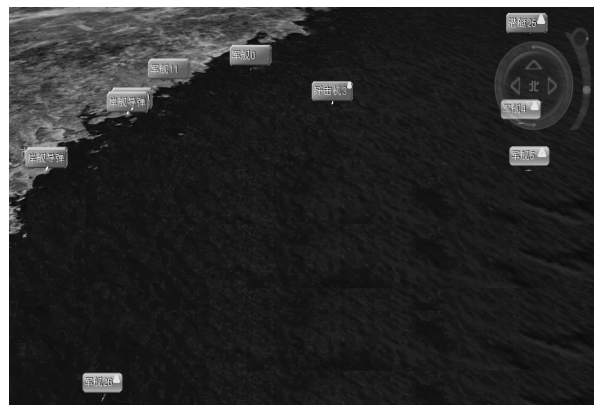


图 7 仿真系统的战场态势