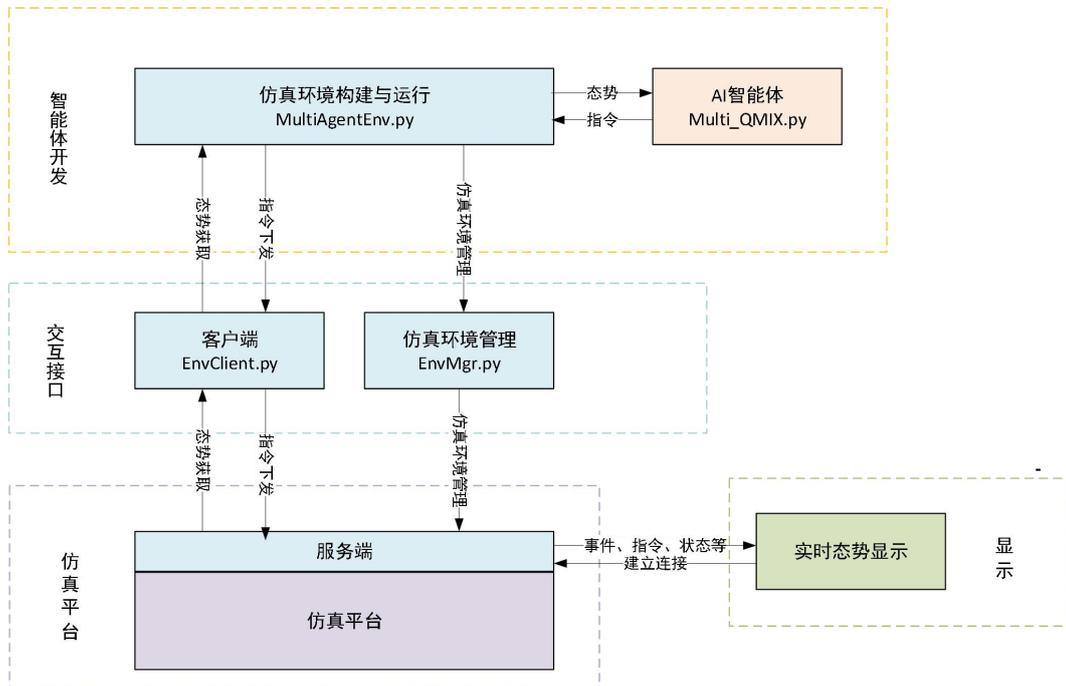


“城池攻略” 智能博弈系统和训练 开发指南

一 仿真环境和训练框架介绍

仿真环境和训练框架包括智能体开发、交互接口和仿真平台三个模块。智能体开发模块包括用户训练智能对战模型的相关接口。交互接口连接智能体开发模块和仿真平台模块，实现了态势获取和智能体生成的动作指令下发功能。仿真平台提供作战环境并实时显示态势信息。



二 智能博弈模型相关接口

智能博弈模型包括两种方案：基于强化学习的多智能体模型和基于规则的智能模型。

1 通用接口模型

1.1 rpyc 连接客户端

本节介绍 `EnvClient` 类，该类与仿真平台建立 `rpyc` 连接、获取态势和下发指令。

表1 类EnvClient及其方法

函数	功能	参数说明	
<code>__init__()</code>	用于建立 <code>rpyc</code> 连接的初始化方法	<code>server</code>	宿主机IP地址
		<code>port</code>	容器5455端口映射到外部宿主机的端口号
<code>get_observation()</code>	获取态势	\	
<code>take_action()</code>	下发指令	<code>cmd_list</code>	智能体生成的指令列表

1.2 本地仿真平台管理类

本节介绍本地仿真平台管理类，该类管理本地仿真环境的各种操作，包括初始化、启动、停止、暂停、继续、关闭、重置等功能管理，主要方法和功能说明如下，详细参数含义见代码中注释。

表2 类EnvMgr及其方法

函数	功能说明
<code>__init__()</code>	仿真环境参数初始化
<code>sock_send()</code>	建立与仿真系统的连接
<code>reopen()</code>	仿真环境重新打开
<code>socket_close()</code>	关闭推演任务
<code>socket_stop()</code>	停止推演
<code>socket_open()</code>	打开想定
<code>socket_start()</code>	开始推演
<code>socket_pause()</code>	暂停推演
<code>socket_resume()</code>	继续推演

1.3 作战单元执行动作指令

本节介绍作战单元执行的动作指令，包括无人机移动指令、无人车移动指令及士兵移动指令等。

表3 作战单元执行动作指令

函数	功能	参数说明	
MoveDirecWrj ()	指定无人机移动方向	self_id	无人机编号
		course	无人机航向
		point_x	无人机在x轴上速度
		point_y	无人机在y轴上速度
		point_z	无人机在z轴上速度
MoveToPointWrc()	无人车移动到指定地点	self_id	无人车编号
		point_x	无人车指定点位置x坐标
		point_y	无人车指定点位置y坐标
		speed	无人车移动速度
MoveToPointSold()	士兵移动到指定地点	self_id	士兵编号
		point_x	士兵指定点位置x坐标
		point_y	士兵指定点位置y坐标
		speed	士兵移动速度
UvaArrack()	无人机攻击指定目标	self_id	无人机编号
		target_id	指定目标编号
FireTargetSold()	火力攻击指定目标	self_id	攻击方编号
		target_id	指定目标编号
		shell_type	默认设置为1

2 基于强化学习的多智能体智能对战模型

2.1 智能体环境类

本节介绍多智能体训练环境类，该类从仿真环境中获取态势信息，并使用智能体模型生成的动作序列控制仿真环境中的作战单元执行动作。

表4 类MultiAgentEnv及其方法

函数	功能	参数说明	
__init__()	初始化智能体训练环境实例	scen_path	想定文件路径
		host_name	仿真环境所在主机ip
		num_agents	智能体数量
		actions	智能体执行动作数量
		max_step	一次训练智能体和环境交互的最大步数
reset()	智能体训练开始前重置环境	\	
step()	<ol style="list-style-type: none">1. 将智能体模型传来的动作序列转换为作战单元实际执行的动作指令2. 通过客户端类EnvClient执行的动作指令发送给仿真系统并执行相应的动作3. 执行完动作后，更新环境状态4. 返回下一个状态、执行完动作后的奖励，本轮任务是否完成等信息	actions	多智能体模型生成的动作序列，一个智能体对应一个动作
get_obs()	获取态势信息	\	
get_state()	从态势信息中提取模型训练需要的状态信息	\	
close()	关闭想定	\	
__open()	打开想定	\	
__start()	启动推演	\	
__pause()	暂停推演	\	
__resume()	暂停后继续推演	\	
__stop()	停止推演	\	

2.2 多智能体模型示例类

本节给出一个多智能体模型示例，并说明该示例类的参数含义。在实际智能博弈算法挑战赛，参赛队伍设计自己的智能体模型，并使用 MultiAgentEnv 环境类训练自己的模型。

表5 类QMIX及其方法

函数	功能	参数说明	
__init__()	初始化Q网络、经验回放缓冲区、优化器	num_agents	智能体数量
		state_dim	状态空间向量维度
		action_space	动作空间维度
select_action()	依据当前状态states生成多智能体执行的动作序列	states	当前状态
		epsilon	探索率
update()	更新QMIX模型参数	batch_size	批大小
		gamma	折扣因子
store_transition()	将当前状态、动作序列、下一个状态、奖励、完成情况存储到经验回放区	transition	当前状态、动作序列、下一个状态、奖励、完成情况

参赛队伍使用下列方法训练并保存自己的模型。训练过程中会保存最优模型，如果已保存模型，可直接加载性能最好的模型，无需重新训练模型。

表6 智能体训练相关方法

函数	功能	参数说明	
train_qmix()	训练QMIX模型	env	智能体训练环境
		state_dim	状态维度
		num_episodes	训练次数
		epsilon	探索率
		num_agents	智能体数量
		model_path	模型保存路径
load_best_model()	加载最优Q网络模型	agent	智能体模型
		save_path	最优Q网络存储路径

3 基于规则的智能模型

3.1 对战地图中道路类

本节介绍对战地图中对道路进行数字化的道路类。该类生成一条道路曲线，可以计算道路曲线的切线并判断给定位置是否到达目标点附近。

表7 类Road及其方法

函数	功能	参数说明	
__init__()	根据关键点序列使用插值方法形成一条光滑的道路曲线	points	道路中心关键点序列
		width	道路宽度
get_closest_index()	得到道路曲线上与给定位置距离最近的点	position	给定位置
get_tangent_direction()	根据给定的点索引和移动方向计算在道路曲线上的移动方向（前后为切线方向，左右为垂直道路方向）	index	道路曲线上点的索引
		direction	给定的作战单位移动方向（前后左右）
is_within_end_radius()	判断当前位置是否在以中点为圆心、半径为radius的圆内	position	当前位置
		radius	半径

作战单元移动方法根据当前位置、移动方向和移动距离计算移动后的位置。运行规则智能体的方法使用智能体生成动作序列直到完成对战任务。

表8 作战单元移动及运行规则智能体的方法

函数	功能	参数说明	
move_unit()	从当前位置按照指令方向移动一段距离	road	道路类
		current_position	作战单元当前位置
		direction	移动方向
		distance	移动距离
run()	运行规则智能体，根据态势信息生成动作直到敌方目标均被消灭并到达目标位置	agent	智能体

3.2 作战编队类

本节介绍作战单元编队类，该类具体功能包括形成指定的编队、判断目标是否进入编队攻击范围以及编队移动的功能。

表9 类Fleet及其方法

函数	功能	参数说明	
__init__()	初始化编队信息	leader_position	领队位置
		follower_positions	跟随位置
		fleet_type	编队类型
		attack_distance	攻击距离
		triangle_side_length	三角形边长
		rectangle_width	矩形宽度
		rectangle_height	矩形高度
set_memeber_ids()	设置队员ID	leader_id	领队ID
		follower_ids	跟随者ID
move_fleet()	移动编队	road	道路实例
		direction	移动方向
		distance	移动距离
check_targets_in_attack_range()	检查目标是否在攻击范围内	target_list	目标列表
adjust_fleet_formation()	调整编队形状（包括直线、三角形及矩形）	road	道路实例
update_fleet_memebers()	更新编队成员	member_statuses	队员状态

3.3 基于规则的智能对战示例

本节介绍基于规则的智能对战示例，该示例按照用户设定的规则完成对战过程。该智能体根据态势信息为每个对战编队生成动作指令，直到完成对战任务并到达目标点附近。用户可自行设计基于规则的智能对战类，并设置对战规则。

表10 基于规则的智能对战类Agent及其方法

函数	功能	参数说明	
__init__()	初始化规则智能体	road	道路实例
get_targets()	获取目标列表	\	
update_targets()	更新列表	new_targets	目标列表
get_states()	获取仿真态势	obs	态势信息
step()	根据仿真态势生成动作指令	obs	态势信息