

2022 IEEE CoG

RoboMaster Sim2Real Challenge

比赛规则手册

V2.0

2022.6.18

主办单位：中国科学院自动化研究所



目录

1. 比赛介绍.....	4
1.1 比赛任务.....	4
1.2 赛道信息.....	4
2. 比赛场地.....	5
2.1 概述.....	5
2.2 初始位置.....	7
3. 实验环境.....	7
3.1 仿真平台.....	7
3.2 实体平台.....	8
3.3 数据接口.....	8
3.4 仿真到实体迁移可能需要考虑的问题.....	10
4. 比赛规则.....	10
4.1 目标块激活条件.....	10
4.2 机器人对抗规则.....	11
4.3 计分规则.....	11
5. 比赛流程.....	11
5.1 仿真阶段.....	12
5.2 Sim2Real 阶段.....	12
5.3 发榜阶段.....	12
6. 比赛日程.....	12
7. 参赛.....	13
8. 奖项设置.....	13

更新日志

日期	版本	修改记录
2022.05.07	V1.3	首次发布
2022.05.17	V1.4	<ul style="list-style-type: none">● 完善了每个赛道所能使用的信息说明● 修改了计分规则中“与敌方机器人对抗的得分”计算机制● 增加了晋级第二阶段的最低标准说明
2022.06.18	V2.0	<ul style="list-style-type: none">● 增加了对实体平台的描述● 增加了对第二阶段（Sim2Real）阶段使用的仿真环境变动说明● 增加了仿真到实体迁移可能遇到的问题描述● 增加了第二阶段（Sim2Real）关于比赛总时长的解释● 增加了实际系统测试的相关说明● 更新了奖项设置说明

1. 比赛介绍

在仿真游戏任务中，越来越多的 AI 算法超越了人类玩家。然而在实际环境中，AI 算法的应用受到很多限制。2022 COG RoboMaster Sim2Real 挑战赛将全面测试 AI 算法在实体机器人的应用可能。

2022 IEEE CoG RoboMaster Sim2Real 挑战赛旨在通过仿真提升机器人的探索和对抗能力，考察动态环境中机器人定位、导航、对抗能力以及从仿真到实体的算法迁移能力。比赛通过全自动运行的 RoboMaster EP 机器人导航寻找并识别对应目标区域，与防守机器人对抗以获得相应分数。

1.1 比赛任务

在 3 分钟比赛时间内，参赛机器人需要顺序完成两个任务：5 个目标块激活，以及防守机器人进行射击对抗。目标激活任务考察参赛机器人运动避障能力，同时考察算法在有里程计/激光雷达/图像等传感器噪声影响情况下的仿真到实体的迁移能力。在射击对抗任务中，参赛机器人需要在保持自身血量的情况下，尽可能的击打防守机器人，考察参赛机器人在对抗过程中的决策能力和算法迁移能力。

比赛划分为两个阶段，参赛队伍首先要在仿真器中完成算法开发，而后根据组委会反馈的算法实际运行结果对算法进行完善，完成在实体机器人上的寻找目标以及对抗任务。此项比赛采用组织方提供的统一机器人平台，参赛人员集中精力开发从仿真到实体的人工智能算法，由组织者在实体机器人上进行算法测试，最终根据任务完成效率和伤害综合表现进行排名。



图 1-1 RoboMaster EP 机器人主体

1.2 赛道信息

根据参赛机器人的输入不同，将比赛分为两个赛道，参赛队伍可根据自身情况进行选择。

赛道 1（信息完备的导航和对抗）：提供参赛机器人车载相机拍摄图像，激光雷达扫描数据，参赛机器人在地图中的位置，血量和剩余子弹数量，目标块的位置和激活状态，敌方机器人的激活状态，位置，血量和剩余子弹数量，以及碰撞相关信息，开发 AI 算法输出参赛机器人的速度控制指令和射击控制指令；

赛道 2（基于图像的导航和对抗）：提供参赛机器人车载相机拍摄的图像和目标块的位置，参赛机器人血量和剩余子弹数量，敌方机器人激活状态，血量和剩余子弹数量，以及碰撞相关信息，开发 AI 算法输出参赛机器人的速度控制指令和射击控制指令。

2. 比赛场地

2.1 概述

本次比赛采用大小为 $8.08 \times 4.48(\text{m})$ 的矩形区域作为比赛场地（场地布局采用 RMUA 比赛场地布置，具体参数可参考 [RMUA 规则手册](#)），场地中包括固定障碍物和动态随机障碍物，如图 2-1 所示。固定障碍物是摆放在场地中大小高低不同、位置固定的障碍物。动态随机障碍物（ $30\text{cm} \times 30\text{cm} \times 40\text{cm}$ ）为 5 个位置随机的目标块和防守机器人。参赛机器人在完成任务同时，需躲避固定障碍物和动态障碍物。

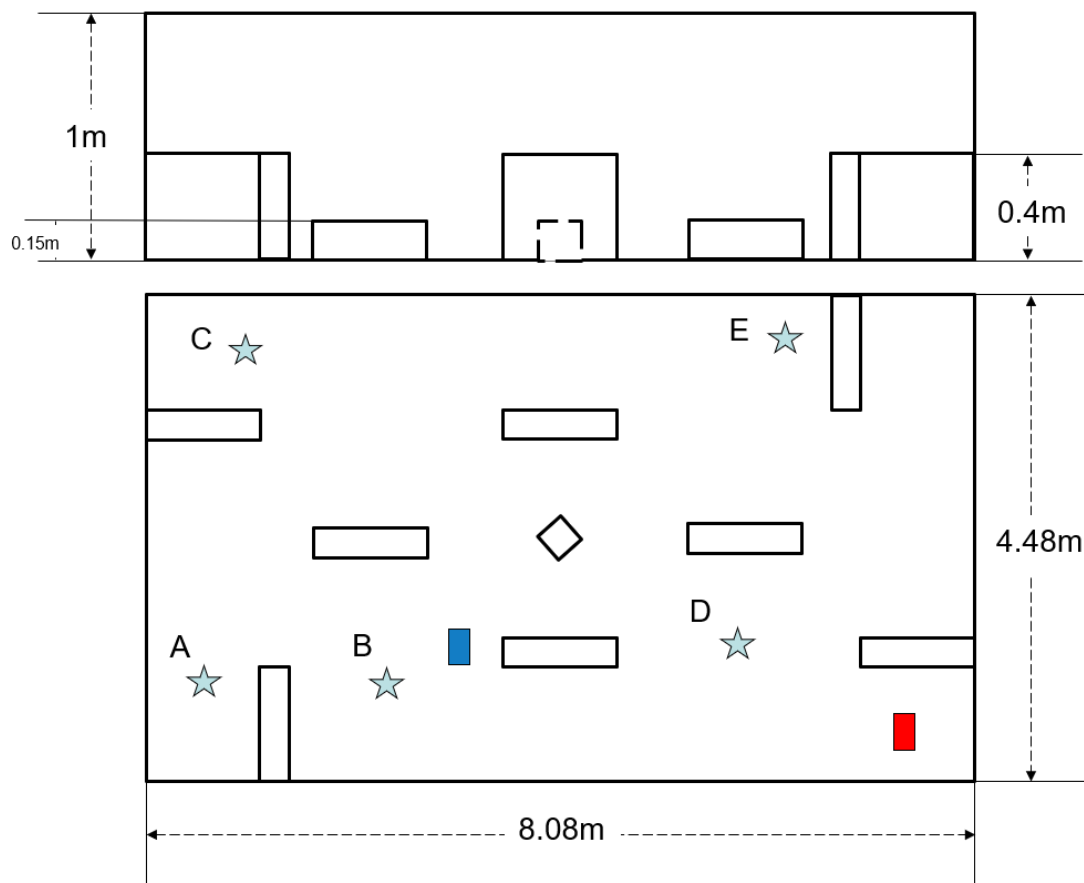


图 2-1 场地模块示意图

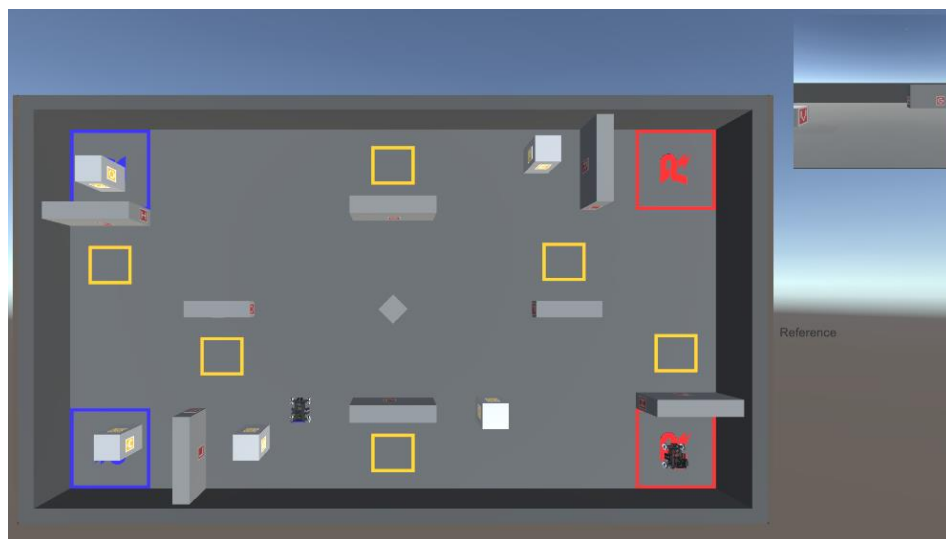


图 2-2 场地全局示意图

2.2 初始位置

参赛机器人初始位置：比赛正式开始时，参赛机器人随机初始在场地某一位置。参赛机器人的位置对赛道 1 是已知信息，对赛道 2 是未知信息。

目标块初始位置：比赛正式开始时，目标块随机出现在场地中 5 个位置，目标块位置作为已知信息发送给参赛机器人。场地内一共有 5 个待激活的目标，每个目标的每个面贴有对应的激活标签，供参赛机器人进行目标激活，如图 2-3。目标的标签分别是 A、B、C、D、E，参赛机器人需要按顺序激活目标。目标块的位置对赛道 1 和赛道 2 都是已知信息。

防守机器人：比赛正式开始时，防守机器人位置随机出现在场地中，在寻找 5 个目标块阶段，防守机器人处于静默状态，不会移动和射击。当 5 个目标块全部被顺序激活后，防守机器人被激活，开始移动射击参赛机器人，对抗过程中，防守机器人的位置是已知信息。



图 2-3 目标样式示意图

3. 实验环境

3.1 仿真平台

比赛的仿真平台采用 Unity 2020.3.12。Unity 是目前 3D 仿真建模领域最出色的平台之一，具有高灵活性，接口丰富等优点。比赛中组委会将提供端到端的可执行任务库 Unity Lab 供参赛队伍使用。

3.2 实体平台

比赛的实体机器人平台是基于 RoboMaster EP 步兵机器人。在此基础上，搭载了激光雷达思岚 RPLiDAR S2 和计算平台 Jetson Xavier NX。激光雷达和计算平台的安装位置具体如图 3-1 所示。

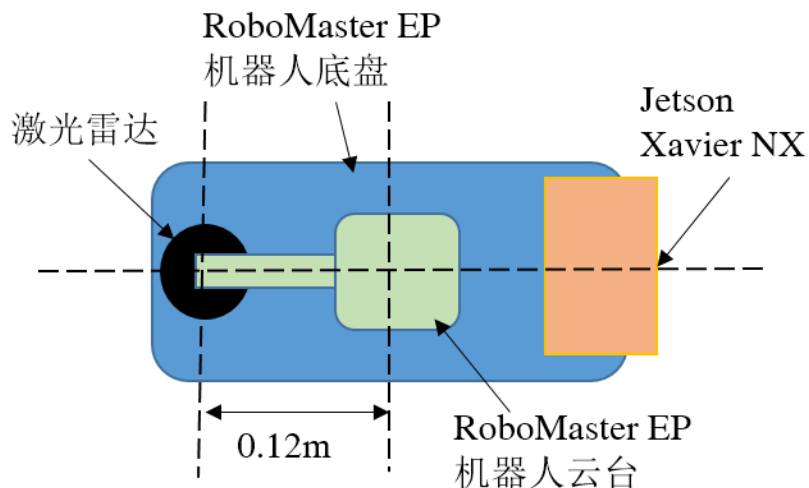


图 3-1 传感器安装示意图

3.3 数据接口

组委会将在 GitHub 上提供算法可交互的环境以及可与环境交互的基线算法，同时将环境接口封装在 CogEnvDecoder 库中，参赛队伍需要在自己的环境配置中安装，算法所需要的依赖将发布在 GitHub 中的 requirement.yaml 文件中，环境的接口信息，也将以示例文件 api_test.py 的形式给出。详情请访问官方的 GitHub 网站：<https://github.com/DRL-CASIA/COG-sim2real-challenge>。

在仿真环境接口中，给出了双方机器人的精确位置信息，没有噪声的激光雷达数据和图像数据，以及比赛中收集到的所有信息，方便参赛选手定义适用于自己算法的奖励函数。但是在测试第一阶段（仿真阶段）提交代码时，所使用的参赛机器人位置会叠加 0.5m 的均匀分布偏差（同一个 episode 的试验中，偏差是固定的，不同的 episode 之间偏差是服从均匀分布）和 0.1m 的均匀分布噪声（每个 step 之间噪声都是不同的，服从均匀分布），激光雷达会叠加 0.05m 的测量噪声。参赛选手需要在训练算法时需要考虑智能体在该条件下的泛化性能。

为了使仿真器与实体环境更加接近，相比于第一阶段，第二阶段的仿真器做了如下变动：

1. 增加了机器人动力学参数接口（摩擦力参数、机器人重量、每个轮子的转动惯量、电机参数（电流力矩常数）和 PID 参数），可以在仿真环境 reset 时指定参数值；
2. 在观测中增加了速度状态反馈，具体见 `api_test.py`；
3. 增加了非时间同步模式，同时算法推理的时间将包含在比赛总时间中，参与分数计算。仿真器的 `step` 函数负责发送动作指令，仿真器会不停地执行该动作指令，直到接收到下一时刻新指令。如果两次动作的时间间隔大于 0.5 秒，仿真器将强制停止机器人运动和射击。实体机器人上采用上述模式设置；
4. 去除强时间同步带来的另一个影响是：使用 `env.step(action)` 收集到的状态不是当前 `action` 造成的影响，而是上一时刻发生的 `action` 造成的影响。如果 `action` 计算时间过长，上一个 `action` 会一直被仿真器重复执行。因此，参赛选手需要保证算法的实时性。具体的执行过程如下图所示。

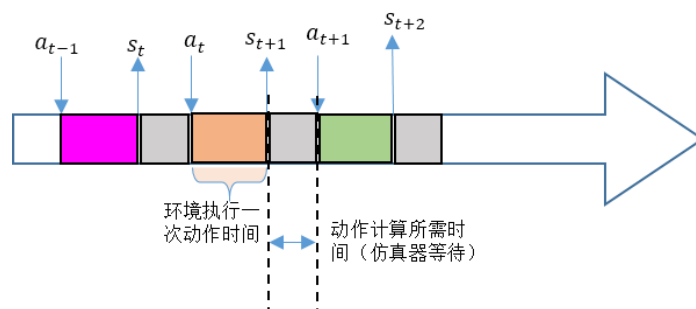


图 3-2 强时间同步模式下的仿真器执行过程。仿真器在执行完当前动作后，会等待程序计算下一时刻执行动作

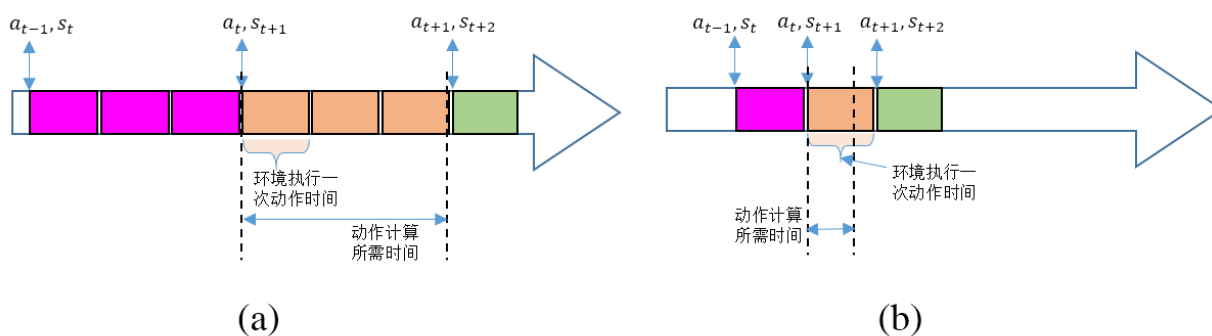


图 3-3 去除强时间同步后的仿真器执行过程。调用 `step` 函数后，会立刻返回当前时刻状态，此时，仿真器会执行当前动作后，在下一个动作到来之前，会重复执行上一时刻的动作（如图 a）。如果下一时刻动作到达时间比环境执行动作结束时间要早，则需要等待当前动作完成(如图 b)。

5. 为使用视觉的方法提供了图像采集仿真平台。该平台采用视觉三维重建对实体测试环境进行了建模，但由于渲染效率较低导致仿真速度较慢，目前不建议作为训练平台，可作为图像采集平台。

3.4 仿真到实体迁移可能需要考虑的问题

关于动力学部分：仿真器中目前无法对轮子打滑对速度变化造成的影响进行模拟，机器人相邻两个时刻控制速度差异较大或与障碍物碰撞的情况下可能会发生打滑现象；地面铺设的材质有两种，一种是摩擦力系数较大的胶质，所占面积比例较大，另一中是摩擦力系数较小的玻璃材质，所占面积比例比较小。

关于激光雷达：激光雷达数据中可能会存在 Inf 或 Nan 数值，这是由于在激光雷达扫描到物体可能没有数据反馈导致的，参赛选手需要对数据进行预处理。另外，由于激光雷达的角分辨率在 0.12 度左右，相比于仿真环境中激光雷达的采样角度，实际激光雷达的采样角度可能会存在 0.12 度左右的误差。此外，由于传感器硬件限制，激光雷达的采集帧率在 10Hz 左右（相邻几个 step 可能返回相同的激光雷达数据），而仿真环境中激光雷达帧率最高可到 50Hz。

关于是否发生碰撞的判定：由于无法直接测量到碰撞事件，测试中会采用与本车之外其他物体（边界）之间的距离进行判定。对于移动物体使用的是位置信息，由于定位系统的波动误差，相比于仿真环境，其测量结果也会有微小波动。

关于图像：如果使用了图像数据，实际图像与仿真图像之间存在差异，参赛选手需要考虑图像之间的差异性。由于相机硬件限制，图像采集帧率最高在 30Hz 左右（相邻 2 个 step 可能返回相同的图像数据），而仿真环境中最高帧率在 50Hz 左右。

4. 比赛规则

比赛总时长为 3 分钟，比赛开始，参赛机器人和 5 个目标块随机出现在比赛场地中，参赛机器人需寻找 5 个目标块，并按照固定顺序（ABCDE）依次完成激活，完成激活后原本静默的防守机器人会被激活，会对参赛机器人进行攻击，参赛机器人需要尽可能地对防守机器人射击并保持自身血量。

4.1 目标块激活条件

比赛过程中参赛机器人需要按照顺序依次激活目标块，需要满足下述所有条件：

1. 参赛机器人与目标块的距离小于 1m

2. 参赛机器人与目标块之间没有障碍物
3. 参赛机器人在世界坐标系下的朝向与机器人与目标块的连线所构成的夹角小于 30 度
4. 目标块是按照 ABCDE 的顺序激活(例如抵达目标块 C 之前, 依次抵达过 AB, 否则目标块无法激活)

4.2 机器人对抗规则

比赛开始时, 防守机器人作为固定障碍物出现在场地中, 只有当五个目标块都激活完成之后, 比赛进入对抗阶段, 防守机器人才被激活, 开始移动与参赛机器人射击对抗。

对抗开始时, 防守机器人和参赛机器人的初始血量为 800, 双方机器人的射击频率为 1Hz, 被击中一次掉血为 100, 每个机器人可射击次数为 24 次, 当下发射击指令多于 24 次时, 射击指令将不会被执行。

4.3 计分规则

参赛机器人的得分 $Score = 60 \times N + A \times 0.5 \times (D + H) - T - 10K$, 考虑以下四个部分:

1. 寻找目标的得分: $60 \times N$, 按顺序激活一个目标获得奖励 60, N 为成功激活目标的个数, 最高得分为 300;
2. 与敌方机器人对抗的得分: $A \times 0.5 \times (D + H)$, A 为敌方机器人是否被激活, 未被激活时 $A=0$, 激活后该值为 1。敌我双方初始血量为 800, 比赛结束时防守机器人的伤害为 $D=800$ -防守机器人血量剩余, 参赛机器人的血量剩余为 H , 希望参赛机器人在保持不被击中的情况下, 尽可能的对防守机器人造成伤害, 最高得分为 800;
3. 比赛总用时: $T(s)$, 希望参赛机器人快速完成任务, 用时越长, 得分越低, 最多扣减为 180;
4. 碰撞惩罚: $K = 2 \times T_k$, T_k 为连续碰撞时间 (单位为秒), 碰撞时间越长, 得分越低, 最多扣减 3600。

例如: 参赛机器人成功到达 5 个目标块后, 与防守机器人对抗并取胜, 剩余血量 100, 总用时 150s, 发生碰撞时间 5 秒, 那么其总得分为 $60 \times 5 + 0.5 \times (800 + 100) - 150 - 10 \times (2 \times 5) = 500$ 。

需要注意的是: 在第一阶段(仿真阶段)中比赛总用时(T)不包含动作计算时间, 而在第二阶段(Sim2Real 阶段)中比赛总用时会将动作计算时间包含在内。

5. 比赛流程

比赛分为仿真阶段、Sim2Real 阶段和发榜三个阶段。

5.1 仿真阶段

5 月 11 日--6 月 18 日为仿真阶段，参赛队伍需在仿真器 Unity 中开发算法，控制机器人完成激活目标和设计对抗的任务，具体流程如下：

1. 参赛队伍通过官网给出 GitHub 网址下载 Unity 环境，参考官方文档配置开发环境；
2. 参赛队伍在仿真环境中开发算法完成比赛任务，并提交代码到官网；
3. 组委会根据仿真测试的结果，决定进入 Sim2Real 阶段的参赛队伍。

5.2 Sim2Real 阶段

6 月 20 日--7 月 30 日 Sim2Real 阶段，成功晋级（最低要求：测试过程中平均每次实验激活目标点数量不少于 3 个，在此基础上根据得分择优选择）的参赛队伍需要根据组委会反馈的实体测试结果对控制算法进行优化，提交过程如下：

1. 参赛队伍将代码提交至比赛官网；
2. 组委会下载（6 月 26 日之后的每周一）各参赛队伍更新的代码并部署到 EP 机器人上执行任务。在测试过程中，考虑到机器人的安全和硬件损耗问题，工作人员首先会用简单测试等级对算法进行评估，当算法平均碰撞时间小于 20 秒时，才进行实际系统测试。如果实际系统测试过程中累积碰撞时间超过 20 秒时，则会人为干预强制终止程序。当机器人在测试过程中出现冲撞障碍物行为，工作人员会强制终止测试并向参赛队伍反馈测试问题。
3. 测试后，将 EP 机器人的数据和比赛录像反馈给参赛队伍；
4. 参赛队伍根据反馈数据和比赛录像进行算法优化。

5.3 发榜阶段

组委会将于 UTC+8（即北京时间）8 月公布比赛结果。

组委会将各参赛队伍最后提交的代码部署到 EP 机器人上运行；将比赛过程中的数据和得分统一发布在比赛官网。

6. 比赛日程

表 6-1 比赛日程

日程	项目	备注
----	----	----

2022 年 4 月 1 日- 2022 年 5 月 10 日	正式报名	公布赛事规则及相关材料
2022 年 5 月 11 日 - 2022 年 6 月 10 日	仿真阶段	参赛队员提交代码到比赛官网
2022 年 6 月 18 日	代码提交	参赛队员提交代码到比赛官网
2022 年 6 月 20 日	第一阶段成绩公布	公布第一阶段成绩和入选第二阶段队伍
2022 年 6 月 20 日 - 2022 年 7 月 30 日	Sim2Real 阶段	组委会反馈代码在真实场地的测试结果
2022 年 7 月 31 日	代码提交	参赛队员提交代码到比赛官网
2022 年 8 月	发榜	

7. 参赛

参赛队员登录比赛官网注册报名比赛，然后组建队伍：

1. 每名参赛队员只准许加入一支队伍；
2. 每支队伍人数：1-3 人；
3. 获奖队伍需是在校的本科生或研究生；
4. 每个队伍可有 1-2 名指导老师；
5. 比赛官网 <https://eval.ai/web/challenges/challenge-page/1513/overview>。
6. 参加比赛队伍在官网报名后，请发送比赛报名信息到邮箱 CogSim2real2022@gmail.com，报名信息包括：队长姓名，邮箱，单位（学校/公司）和队员姓名、邮箱和单位（学校/公司）

8. 奖项设置

表 8-1 奖项设置

奖项	排名	数量	奖励
----	----	----	----

特等奖	第一名	1	500 美元
一等奖	第二名	1	mavic mini
二等奖	第三名	1	RoboMaster TT
三等奖	第四名	1	Osmo Mobile5

注：由于赛道 2 提交队伍过少，将不为赛道 2 设置单独奖项，仅针对赛道 1 设置奖项。