

2019 FIRA 仿真 5 自动化平台使用手册

FIRA 国际机器人足球仿真平台中型场地组是最早出现的 FIRA 仿真平台，由澳大利亚 GRIFFITH 大学主持开发，该平台基于 Havok 引擎开发，具有真实感强，能很好的仿真实物机器人的真实物理碰撞等特性，并且随着操作系统的升级，能一直兼容各个版本的 Windows 操作系统，平台具有良好的稳定性。

2018 年 FIRA 国际考虑到原有仿真 5 比赛过程中双方队员操作机器人摆放，平台不够自动化，对原有平台进行了自动化改进。自动比赛裁判系统可以让比赛队伍通过程序自动摆放机器人，免除人工摆放，省去了人工费时费力的工作，使得平台工作更加自动化和策略可以更好地智能化。

以往比赛规则没有对双方在球场四周推球进行限定，2018 年 FIRA 国际组织修订了规则，禁止比赛双方在比赛四周推球的消耗比赛时间，极大提高了比赛的观赏性和更有效地利用比赛时间。

对于比赛日志文件的生成和发布，过去比赛队伍通过视频录像的方式，费时、费力、又费磁盘空间，新的自动化平台自动将比赛双方的比赛空间信息记录形成日志文件，赛后发布在互联网，方便大家的交流、学习和进步。

1、平台下载地址

<https://github.com/zerowind168/SIM5-exe/raw/master/SIM5Installation.exe>

2、平台运行和开发环境

(1) 硬件需求：

1) 酷睿五代 i5 处理器 i5-6400 或更高性能的 CPU; 2G 或以上系统内存; 10G 以上可以硬盘空间。

2) 具有 1G 显存或以上级别的显示卡。

3) 至少可以支持 1920x1080 分辨率的 17 英寸及以上显示器。

(2) 软件需求：

1) Windows 7 旗舰版的操作系统。

2) DirectX 8.0或以上版本。

3) C++语言开发动态链接库程序，可以使用VC6及以上环境开发。

3、平台文件组成

将 FIRA 仿真 5 平台安装包启动运行，需要将 SimuroSot5 仿真机器人足球比

赛平台安装到 C 盘根目录(C:\), 安装程序在 C 盘根目录下建立一个文件夹, 名字为 Strategy, 如图 1 所示, 文件结构组成如下。

```
[c:\Strategy]
.....[blue]
          .....Team1.dll
.....[yellow]
          .....Team2.dll
.....SimuroSot5.exe
.....WorldModel.exe
.....Referee.dll
.....Readme.txt
.....Strategy4Blue.dll
.....Strategy4Yellow.dll
.....[src]
          .....[Strategy4Blue]
          .....[Strategy4Yellow]
```

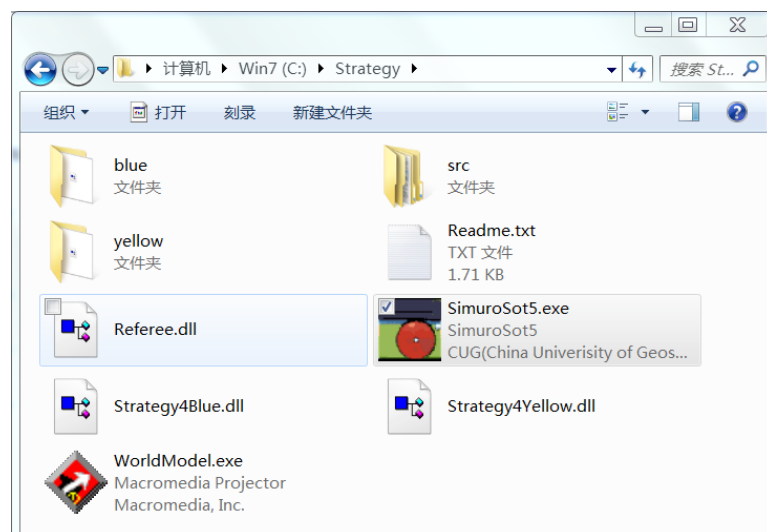


图 1 仿真 5 平台文件组成图

其中, 黑色粗体部分列出的文件和文件夹对于策略开发人员是不能改动的程序和文件, 蓝色斜体的文件和文件夹提供的是供策略开发人员开发和更新的部分。

blue 文件夹里面的 team1.dll 和 yellow 文件夹里面的 team2.dll 是裁判代理模块, 用来执行自动裁判判罚和发送比赛环境数据的。Referee.dll 是裁判模块, 供 team1 和 team2 代理模块调用执行规则的判罚。

SimuroSot5.exe 是平台的主控程序, 负责平台的启动、暂停和关闭。同时, 负责比赛中规则的判罚以及机器人和球的摆位控制等。

WorldModel.exe 是比赛仿真环境的主要服务程序，负责对比赛策略的自动调用、模拟机器人、球和球场的世界物理模型，并将赛场环境数据提供给策略模块，接收策略的数据。

Strategy4Blue.dll 是蓝队比赛策略程序，默认是平台自带的演示程序，每个参赛队伍需要自己设计和开发自己蓝队队伍的策略程序，替换该文件。注意需要保留这个文件名，并实现指定的接口函数。具体的接口函数见后续介绍。

Strategy4Yellow.dll 是黄队比赛策略程序，默认是平台自带的演示程序，每个参赛队伍需要自己设计和开发自己黄队队伍的策略程序，替换该文件。注意需要保留这个文件名，并实现指定的接口函数。具体的接口函数见后续介绍。

src 文件夹包含黄队和蓝队策略的模板程序源代码，该模板中有需要实现的指定函数接口、数据结构和相关说明。参赛队伍直接在模板的基础上实现自己的队伍策略即可。策略模板代码是基于 Visual Studio 2010 环境开发的。

4、平台操作

打开 SimuroSot5.exe 程序  启动平台（请不要单独启动 WorldModel.exe），平台运行界面图 2 所示。



图 2 仿真 5 比赛平台

点击“Start”按钮开始比赛，平台会加载双方策略文件 Strategy4Blue.dll 和 Strategy4Yellow.dll。点击“Pause”可以暂停平台，点击“Close”将关闭

仿真平台。比赛进行期间，仿真平台自动将比赛双方的球员位姿信息和球的位置信息记录生成日志文件，并存放在 C:\Strategy 文件夹里，如：20180617090302-5-DEMO Yellow-DEMO Blue.rlg。

特别注意：

本比赛平台在比赛过程中是自动运行控制的，不需要人工干预。由于比赛平台在控制过程中需要执行鼠标控制，所以，在比赛进行中，不可以对鼠标和键盘进行操控，以免对比赛平台产生干扰。

5、策略开发接口

策略模块需要提供 5 个接口函数供代理裁判模块调用，这 5 个接口函数如下介绍。

5.1 SetFormerRobots(PlayMode gameState, Robot robots[])

该函数在代理裁判需要拿到先摆方的球员位置数据时被调用，该函数会通过参数gameState提供判罚系统的判罚结果，策略程序需要根据判罚结果（即开球、争球、点球、门球）设置本方的五名球员的位置和朝向数据到robots数组。最终，仿真平台根据robots数据自动摆放本方机器人。

5.2 SetLaterRobots(PlayMode gameState, Robot formerRobots[], Vector3D ball, Robot laterRobots[])

该函数在代理裁判需要拿到后摆方的球员位置数据时被调用，该函数会通过参数gameState提供判罚系统的判罚结果，formerRobots提供先摆方的球员位置数据，ball提供球的位置数据，然后根据这些信息设置本方想要摆放的五名球员的位置和朝向数据到laterRobots数组。最终，仿真平台根据laterRobots数据自动摆放本方机器人。

5.3 SetBall(PlayMode gameState, Vector3D * pBall)

该函数在代理裁判需要拿到开门球方队伍想要摆放的球的位置数据时被调用，该函数设置想要摆放的球的位置数据到pBall。最终，仿真平台根据pBall数据自动摆放开门球方的门球位置。

5.4 RunStrategy(Environment *pEnv)

该函数在代理裁判模块确认当前周期场上没有任何判罚（开球、争球、门球、点球等）产生时被调用，该函数提供当前仿真环境赛场数据pEnv，策略程序根据赛场数据做出决策，设定相应的策略控制命令至pEnv（机器人控制命令）。

5.5 SetBlueTeamName(char* teamName)/ SetYellowTeamName(char* teamName)

该函数在代理裁判模块一开始比赛是调用，通过该函数来设定比赛的蓝队或者黄队队伍名称。仿真平台根据此设定将队伍名称显示到平台控制部件的队伍名称处，如果在仿真平台的控制部件处没有显示自己队伍的名字，说明自己的策略没有正确连接到仿真平台。

6、平台环境坐标系

仿真平台的场地标示和尺寸坐标见图 3。仿真平台和策略系统交互的坐标系统的原点在场地的左下角处 (0,0)，x 轴数值大小水平向右增长，y 轴数值大小垂直向上增长。长度单位使用的是厘米（注意，不是英寸单位）。

第 5 节中提到的策略程序中的接口函数涉及到的参数数据都是基于此坐标系。

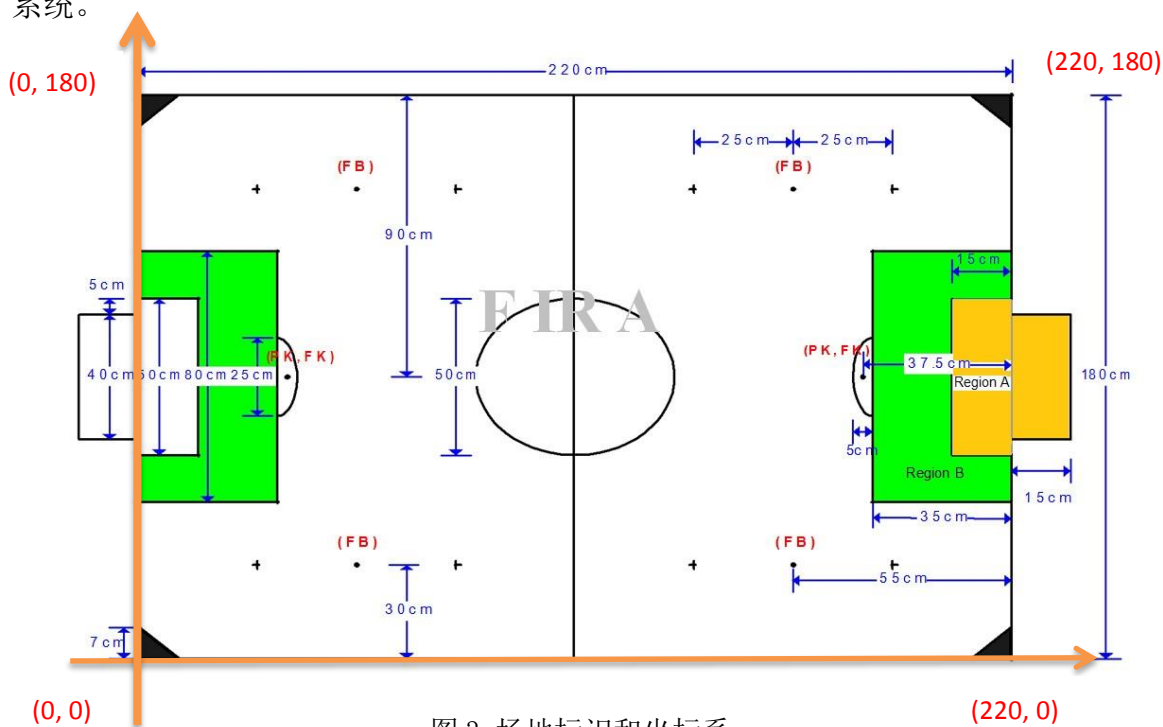


图3 场地标识和坐标系