

# 设计实验室2 季 控制机器人

6. 01-2011年秋

## 目

- 实验状态机控制真实的机器
- 研究6. 01机器人上的真实世界距离传感器：声纳
- 搭建并演示状态机，使机器人完成任务：  
跟随一个边界

## 1 材料

这个实验室应该和一个搭档一起做。 各合伙企业应具备：

- 实验室的笔记本电脑。
- 机器人，（长，灰色）串行电缆，和（短，蓝色）串行到USB适配器。
  - 串行电缆为米黄色或灰色长电缆。 大多数机器人已经有了一个附件。
  - 警告：如果你的机器人开始走得太快或离开你，拿起它！ ！
- 白色泡沫芯板，一面有气泡包装。

一定要把你所有的代码和数据邮寄给你的搭档。 你俩都需要带着它来参加你的第一次面试。

## 2 简单的大脑

大脑是一个Python程序，它指定机器人的行为。 在“机器人基础设施指南”中详细描述了构建和运行大脑的过程。

### 目的：

- 建立一个状态机“大脑”来控制机器人，首先是在翱翔模拟器-Lator上，然后是在一个真正的机器人上。
- 在翱翔中运行一个简单的大脑，并记录机器人的路径
  - 修改简单大脑，使机器人转动到位
  - 在先锋机器人平台上运行大脑



**资源：**

- ~/Desktop/6.01/设计Lab02/smBrain.py：一个简单的机器人大脑，使用6.01状态机类sm
- 教程：翱翔的虚拟世界

详细指导：使用6.01实验室笔记本电脑进行以下操作：

**1.在模拟器里运行大脑。**

- a.在终端窗口中，键入solar&。
- b.单击SOAR的模拟器按钮并双击tutorial.py。这将一个特定的虚拟世界加载到我们的机器人模拟器中。
- c.单击翱翔的大脑按钮，导航到桌面/6.01/DesignLab02/smBrain.py，然后单击“打开”。这将特定的状态机定义加载到机器人模拟器中。该状态机描述了机器人将采取的行动，以响应感知到的信息-关于它周围的虚拟世界。
- d.点击翱翔的开始按钮，让机器人运行一段时间。
- e.点击翱翔的停止按钮。
- f.请注意生成的图形；它显示了机器人在大脑运行时-下降的路径的“黏液轨迹”。你可以关上窗户。（如果您不希望大脑产生黏液轨迹，可以将绘图Slime Trail参数设置为smBrain.py文件中的Robot Graphics构造函数为false）。

**2.修改大脑并运行它。**

- a.在终端窗口中，键入空闲&以打开空闲环境。
- b.单击“空闲文件”菜单，选择“打开。。。”，导航到Desktop/6.01/DesignLab02/smBrain.py，然后单击Open。
- c.控制机器人动作的状态机由MySMClass定义定义。将此状态机视为以感官数据作为输入，并将其作为-结构中的输出返回给机器人如何行为。 **I0. 行动**通过MySm类的get Next值方法返回的对象告诉机器人如何改变它的行为，并且有两个对我们很重要的属性：
  - ◆ fvel：指定机器人的前进速度（单位：米/秒）
  - ◆ Rvel：指定机器人的旋转速度（以每秒弧度为单位），其中正旋转-逆时针旋转
- d.找到在大脑中设置速度的地方，然后修改它，使模拟机器人旋转到位。
- e.保存文件。
- f.回到“翱翔”窗口，单击“重新加载大脑”按钮
- g.通过单击“开始”和“停止”按钮来运行大脑。

**3.在机器人上运行**

- a.将机器人连接到您的笔记本电脑上，确保电缆绑在机器人背面的手柄周围。
- b.机器人上的电源，侧板上有开关。

- c. 点击翱翔的先锋按钮，选择机器人。你应该能听到声纳传感器发出滴答声。
- d. 一个合作伙伴应该负责保护机器人的安全。防止电缆缠绕在机器人的轮子上。如果机器人开始远离你，拿起它，然后，使用机器人上的开关关闭它。
- e. 点击翱翔的开始按钮。

### 3 声纳

**目的：** 调查声纳传感器的行为，并修改机器人的大脑，使机器人与障碍物保持一定的距离。不要花超过10或15分钟的时间来试验声纳。当你完成后，请工作人员做一个检查。

`soar.io.SensorInput` 我的SMClass的get Next Values方法的inp参数是类的一个实例，我们将其导入为io。传感器输入。它有两个在-的悼念，计量和声纳。对于这个实验室，我们将只使用声纳属性，其中包含一个8个数字的列表，表示来自机器人的8个声纳传感器的读数，该传感器以米为单位提供距离读数。列表中的第一个读数（索引0）来自最左边（从机器人的角度）传感器；来自最右边传感器的读数是最后一个（索引7）。

**详细指导：**

- 修改大脑，使它将两个速度设置为0，并取消注释线

打印Inp. sonars[3]

重新加载大脑并运行它。它将打印inp. sonars[3]的值，这是来自前向声纳传感器之一的读数。

- 从多远可以得到可靠的距离读数？当最接近的东西比那更远时会发生什么？
- 离传感器很近的东西会发生什么？
- 声纳传感器与表面之间的角度变化是否会影响读数？这种行为是否取决于表面的材料？尝试气泡包装与光滑的泡沫芯。
- 现在，将声纳监视器参数设置为Robot Graphics构造函数为True。

重新加载大脑并运行它。这将带来一个窗口，显示所有声纳读数的图形。光束的长度对应于读数；红色光束对应于“没有有效的测量”。测试你所有的声纳都是通过依次阻塞每个声纳来工作的。如果您注意到任何传感器有问题，请与工作人员交谈。

**检查1。** Wk. 2. 21：向工作人员解释你对声纳的实验结果。证明你知道你的伴侣的名字和电子邮件地址。

使机器人向前移动到它前面大约0.5米的障碍物，并保持它在那个距离，即使障碍物来回移动。通过编辑我的SMClass的get Next Values方法来实现这一点；没有必要改变大脑的任何其他部分。 不要将前向速度设置为高于0.3（或低于-0.3）。在模拟中调试它，单击SORAY的模拟器按钮并选择tutorial.py。一旦它看起来很好，就在一个真正的机器人上运行它，选择翱翔的先锋按钮。

检查2。

Wk. 2. 22: 向工作人员展示你在一个真正的机器人上保持距离的大脑。

## 4遵循边界

目的:

我们现在的目标是建立一个状态机，控制机器人完成一个更复杂的任务:

- 1.当附近没有东西时，它应该向前直走。
- 2.当它到达前面的障碍物时，它应该跟随障碍物的边界，保持机器人的右侧在0.3到0.5之间  
离障碍物几米远。

绘制一个状态转换图，描述壁面跟踪过程中的每个不同情况（状态），以及所需的输出(动作)和下一个状态应该响应该状态下可能的输入(声纳读数。 首先考虑机器人通过空空间直接向前移动的情况，然后考虑您遇到的输入条件和结果的新状态。 仔细想想在内外角该做什么。 记住机器人围绕中心点旋转。 试着把状态数保持在最小。

支票3。 Wk. 2. 23: 向工作人员显示您的状态转换图。 明确说明状态转换的条件是什么, 以及哪些操作与每个状态- 相关。

将当前的sm Brain.py文件复制到边界Brain.py(您可以使用空闲中的SaveAs来完成此操作), 并修改它以实现由图表定义的状态机。 确保您定义了start State属性和get Next Values方法。

努力保持你的解决方案简单和一般。 使用良好的软件实践: 不要重复代码, 使用助记符名称的助手过程, 尝试使用很少的任意常量, 并给出使用描述性名称的常量。

若要调试, 请添加显示相关输入、当前状态、下一状态和输出操作的打印语句。

记录模拟机器人的黏液痕迹, 沿着一系列的墙壁; 确保它能处理外部和内部的角落。 绕着非常锋利的角落或发夹旋转, 比如tutorial.py中的L, 是不需要的, 但是额外的酷。

检查4。 Wk. 2. 24: 向工作人员展示你的边界跟随者。 解释为什么它的行为方式。 把你的密码发给两个搭档。

麻省理工学院开放课程  
<http://ocw.mit.edu>

6.01SC 电气工程和计算机科学导论  
2011年春季

有关引用这些材料或我们的使用条款的信息，请访问：<http://ocw.mit.edu/terms>.