

中国机器人大赛-机器人旅游寻宝项目

——国家级三等奖

林德旸

18 级自动化创新班

3118001018

LaTeX

更新：2020 年 1 月 1 日

摘 要

中国机器人大赛于 2019 年 8 月 27 日至 8 月 31 日，在山东省即墨市举行，在机器人旅游寻宝项目中，我担任主力编程队员，设计整体运行框架，创新性使用链表构建地图信息，方便比赛调试于对于比赛地图新节点的添加；同时此次比赛的所使用的循迹板电路与调节算法也由我独立设计，使用基于 BP 神经网络的动态阈值的调节方法，使得比赛用车在循迹时能够自动适应不同的表面类型，同时也极具有参数初始化方便的特点。由于篇幅有限，在此仅着重介绍个人主要设计的链表式的地图录入方式算法，与动态阈值循迹板的思想与设计，其详细的车辆设计方案与具体的程序工程文件已经上传至 Github，如果您想了解更多关于此项项目，请访问：<https://github.com/ldylab>，在 Repositories 下获取所有比赛项目个人的开源程序，再次感谢！

关键词：中国机器人大赛，游中国项目，链表构建，BP 神经网络动态阈值

1 项目背景

机器人寻宝项目要求机器人在规定的旅游寻宝任务时间（180 秒）内，穿越险境、识别搜寻宝物。时间富裕时，游历尽量多的景点，获得尽量多的得分，并在规定时间内回到出发地。

机器人竞赛的目的是引导参赛队研究、设计并制作具有优秀硬件与软件系统的移动机器人，逐步提高机器人多方面的能力与智能，如：

1) 系统规划与优化能力在设定的假期时间内，通常是不可能游历全部景点（险境）的，应该有选择地完成计划中的旅游活动，并回到出发地点，需要有一定的系统规划与优化能力。想得高分，就要游历尽量多的景点和难度高的景点，但存在来不及在规定的假期回家的危险。每轮比赛中走两次，在第一次经验基础上，第二次可更好体现优化和智能。

2) 应变能力旅游寻宝路线可能在比赛开始前很短的时间内正式公布；有些可移动景点与路障的摆放数量与位置，在机器人放进出发区后，随机确定。这在一定程度上可控制竞赛的难度，并使

旅游路线有一定的不确定性。

3) 视力及定位考验机器人辨别数字、二维码、文字、形状和色彩的能力, 引导机器人视力及定位能力的提高。

4) 爬坡能力坡度大、坡道狭窄, 包括变形坡道及坡道转弯能力, 难度较大。

5) 翻越障碍的能力有门、减速板、台阶等障碍物。

6) 变形能力穿越有些险境(景点)时, 可能需要机器人变形; 有时机器人变形可以获得更好的运动性能。

7) 快速性与稳定性机器人在整个旅游过程中, 始终要在快速性与稳定性中间求得平衡, 否则, 难以取得好的成绩。

2 项目实施情况

其整体设计如下:

表 1: 规则规范

机器人	规则规范	实际设计
肩高	肩膀转动轴心到地面的高度大于 2 倍机器人承重轮最大前后轴距	机器人承重轮最大前后轴距为 135mm, 肩膀转动轴心到地面的高度为 276mm, 满足规范要求。
手臂	有可以分别独立运动的左右手臂。手臂长度大于 1/3 肩高	1/3 肩高为 92mm, 手臂长度为 105mm, 满足规范要求。
脚长	双腿连接点到地面的高度大于 1 倍机器人承重轮最大前后轴距	轴距为 135mm, 双腿连接点到地面的高度为 147mm, 满足规范要求。

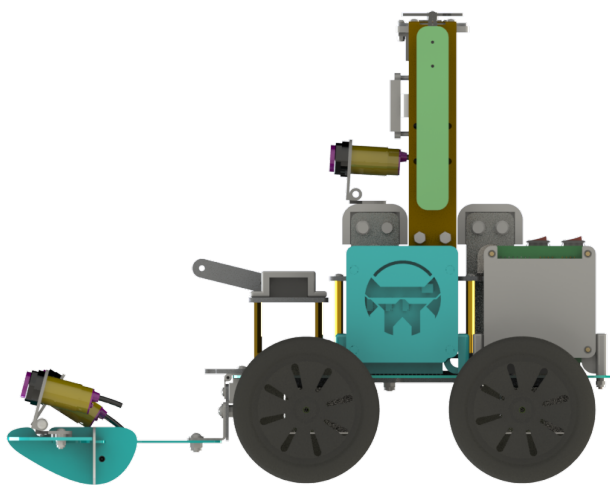


图 1: 侧视图

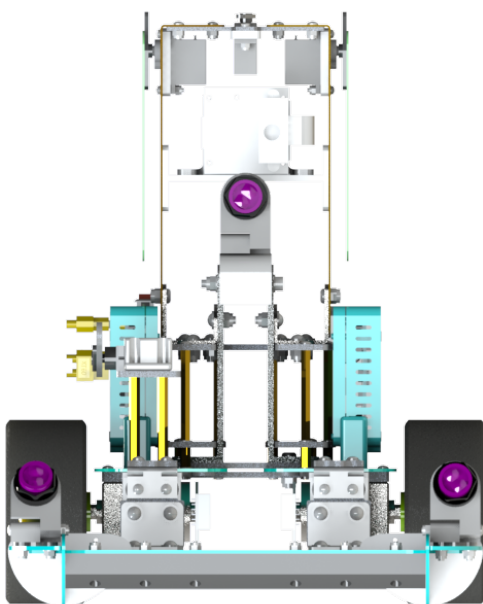


图 2: 正视图

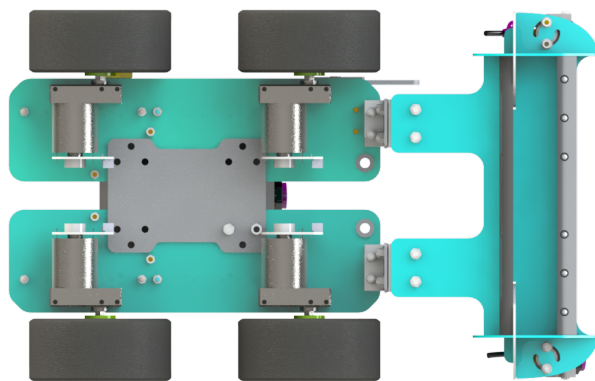


图 3: 仰视图

机器人的底盘由电路板制作而成，也就是直接在电路板上安装电机、连接零件等结构件。在实际设计中，机器人的底盘由两块 3mm 玻纤板连接加固，提高了车体的侧向刚性。机器人的移动底盘使用 4 个带编码器、减速器的直流无刷电机，同时搭配了大轮径轮胎，大大提高了机器人的越障能力。

在机器人中，关键传感器共有 2 种，分别是红外开关传感器和二维码识别模块。二维码识别模块，通过 3D 打印制作的固定架固定于机器人的身体上，用于识别比赛时的二维码。红外开关传感器则使用了 3 个，其中两个安装于循迹板上，用于检测十字路口，另一个安装于双脚胯部连接处，用于检测水桶、挡板等障碍物。

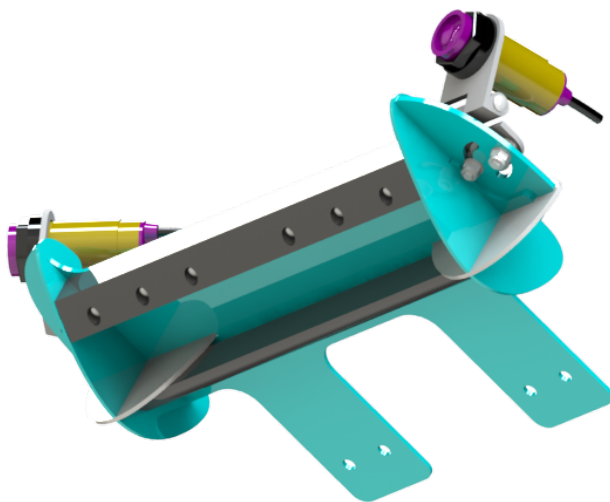


图 4: 循迹板

循迹板的边角使用覆铜板焊接，用于固定红外开关。并且，在循迹板对管的前后都分别安装有挡光板，用于减少外界光源的干扰。同时，大宽度、多接收管的设计可以提高循迹板的分辨率，增强巡线的稳定性。

3 项目个人主要承担工作

3.1 整体程序框架设计

车辆整体采用 STM32F103 单片机，整体程序运行大致流程图如下，其详细代码工程见：

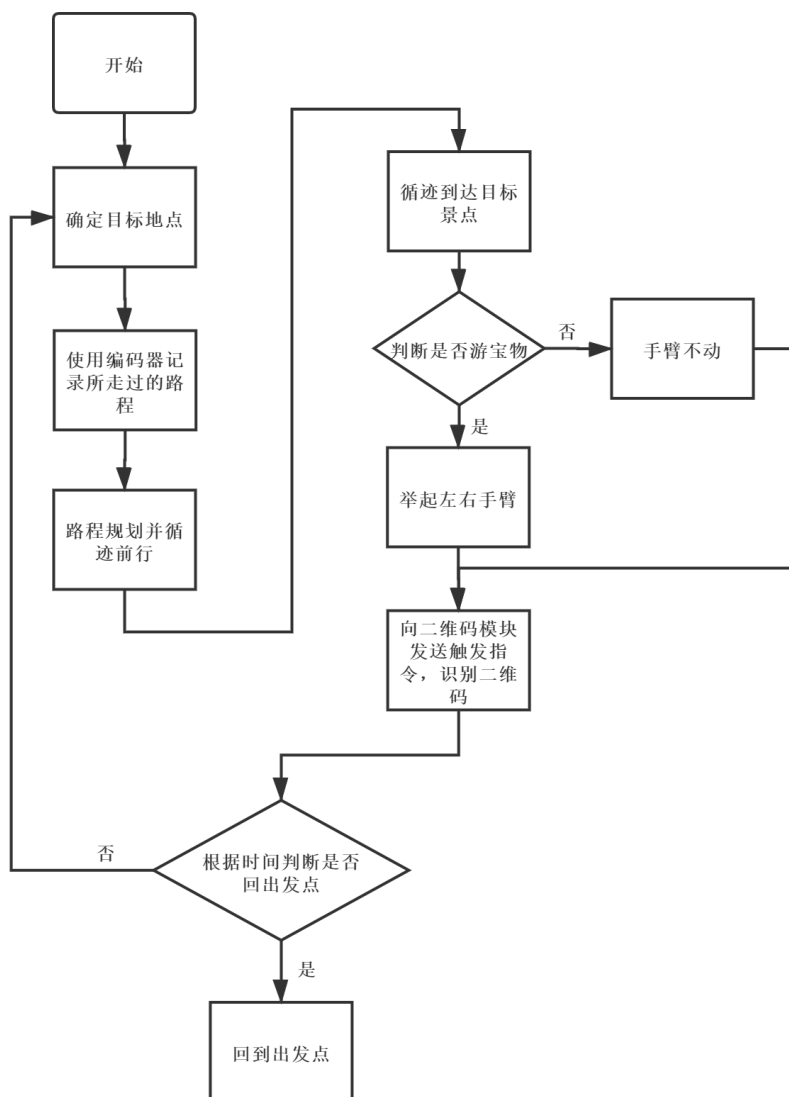


图 5: 整体程序设计流程图

3.2 循迹板整体程序设计

循迹板整体采用 STC 单片机设计，具有 12 路灰度接收管 ADC 输入作为判断场地颜色与目标颜色的输入值，对输入值进行 BP 整合处理后反馈出判断为目标线或是为场地，其主要 BP 算法工作如下图所示：

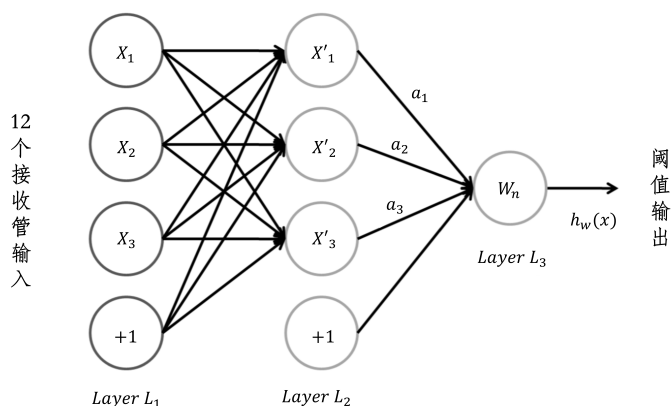


图 6: BP 参数整定

其整定收敛效果如下图所示，可见其有着较强的收敛性，可分辨出两种不同的颜色的表面，同时其稳定性也因自整定的算法而有着极大的提升。

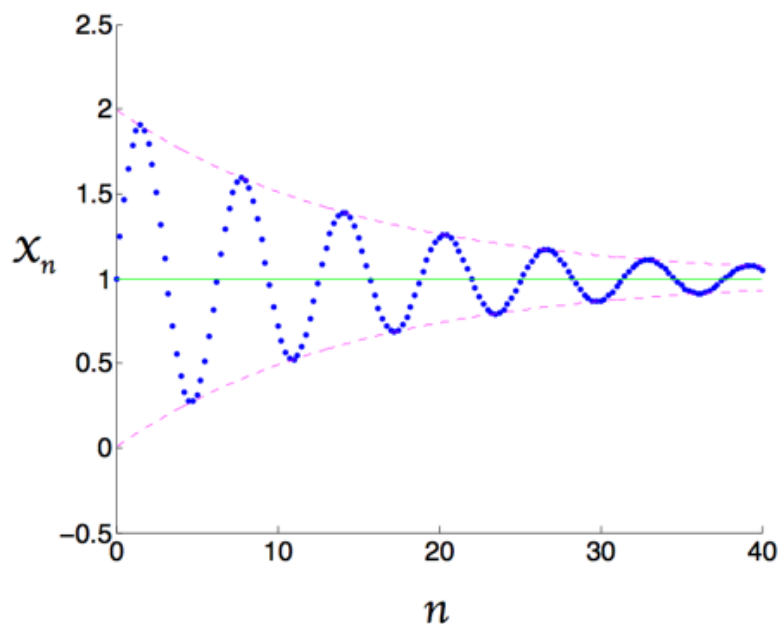


图 7: BP 参数整定收敛图

其整体运行流程图如下：

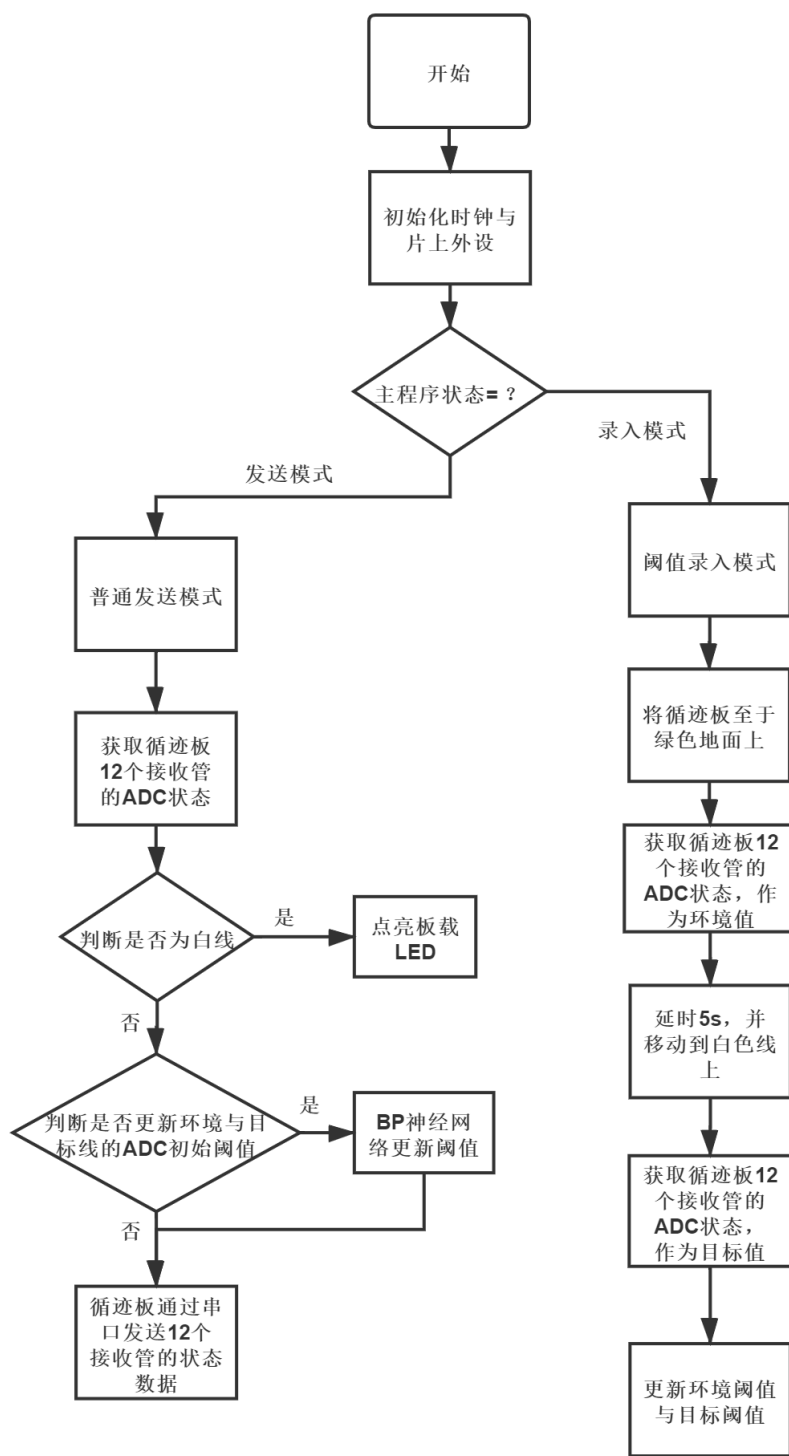


图 8: 循迹板整体流程图

3.3 地图信息录入链表程序设计

其主要设计如下所示:

链表地图节点信息如下

```
1 typedef struct _node
2 {
3     u8  nodenum;    // 地图坐标
4     u16 flag;       // 是否已经走过标识符
5     float angle;    // 地图角度
6     s16 step;       // 地图线长
7     float speed;    // 寻线速度
8     u8 function;    // 到达后做指定动作
9 }NODE;
```

链表地图节点信息具体录入如下 (节选)

```
1 // 地图信息{ 结点名字, 标志位, 角度, 长度(厘米), 速度, 功能函数}
2 // 如 /*S1 1*/ {N3, FLEFT|M2L, 160, 280, 100, 1}, 代表从“寻右线, 线数
   // 目由多变小认为到达路口”, “S1点指向N3点的角度为160度”,
   // “长度为280cm, 推荐速度为100”, “普通路口”
3 NODE Node[10]=
4 {
5     /*S1 1*/ {N3, FLEFT|M2L, 160, 280, 100, 1},
6     /*P1 2*/ {N1, FLEFT|L2M, 180, 50, 40, 1},
7     /*N1 3*/ {P1, FRIGHT, 0, 20, 60, UpStage}, {B2, FRIGHT, 135, 50, 40,
   // BLB}, {B1},
8     /*B1 K*/ {N1, FRIGHT|M2L, 0, 50, 70, 1}, {N2},
9     /*B2 K*/ {N1}, {N4, FLEFT|M2L, 150, 40, 40, 1},
10    /*B3 K*/ {N2, FRIGHT|M2L, -150, 50, 35, 1}, {N4, FRIGHT|M2L
   // , 30, 50, 50, 1},
11    /*N2 4*/ {B3, FLEFT, 30, 50, 30, BLB}, {P2, FLEFT, 180, 30, 30,
   // UpStage}, {B1, RESTMPUZ|FRIGHT, 0, 80, 65, BBridge}
12    /*P2 5*/ {N2, RESTMPUZ|FRIGHT|L2M, 0, 50, 60, 1},
13    /*S2 6*/ {N6, M2L|FRIGHT, 30, 180, 70, 1},
14    /*P3 7*/ {N3, DANGERSTAR|FRIGHT|DRIGHT, 180, 290, 100, 1}
15 }
```


具体路程地图节点信息录入如下(节选)

```
1  u8 route[100]= //先南极再珠峰 跑QQB
2  {
3      N2,B1,N1,P1,           //第一段路程
4      N1,B2,N4,N5,N6,P4,     //打点后第二段路程
5      N6,C1,C2,N13,P6,       //打点后第三段路程
6      N13,N12,N11,N16,       //第二段路程
7      N21,N22,C9,P8,         //第三段路程
8      C9,N22,N21,N20,P7,     //打点后第四段路程
9      N20,N21,N16,N11,N10,N9, //打点后第五段路程
10     N10,N15,S4,N15,C5,N16,  //打点后第六段路程
11     N17,S5,N17,N18,N12,N13, //打点后第七段路程
12     C2,C1,N6,S2,N6,N5,N4,N3,P3, //打点后第八段路程
13     N3,N4,B3,N2,P2,255
14 };
```

通过把地图作为链表的结点极大的节省了地图录入的时间,同时对于每一个路段的速度,角度,与加速度信息录入极为方便,极大的方便了在比赛时的调试。

4 个人总结

通过这一次比赛,自己学习到如何从整个系统去思考问题,怎样去做一个系统性的规划去解决问题。同时也通过这一次的比赛系统学习了位置式 PID,增量式 PID 控制算法,同时也具有创造性的使用 BP 神经网络算法作为参数调节参数的工具并获得稳定不错的效果,使用链表式地图录入方法也极大的提高了整体的调试效率。在暑假这两个月的努力后获得了自己较为满意的比赛结果。

参考文献

- 祝相泉, 黄海龙, 田昊. 无刷直流电机模糊 PID 控制 [J/OL]. 辽宁工业大学学报 (自然科学版),2020(01):1-4[2020-01-01].<http://kns.cnki.net/kcms/detail/21.1567.T.20191227.1041.020.html>.
- Elham Yazdani Bejarbaneh,Ahmad Bagheri,Behnam Yazdani Bejarbaneh,Salinda Buyamin,Saeed Nezamivand Chegini. A new adjusting technique for PID type fuzzy logic controller using PSOSCALF optimization algorithm[J]. Applied Soft Computing Journal,2019.
- 陈通, 周晓辉. 基于 BP 神经网络的深层感知器预测模型 [J]. 计算机与数字工程,2019,47(12):2978-2981+3009.
- S.P. Rajasekar,M. Pitchaimani,Quanxin Zhu. Dynamic threshold probe of stochastic SIR model with saturated incidence rate and saturated treatment function[J]. Physica A: Statistical Mechanics and its Applications,2019,535.