

# 基于二阶段双向搜索的解魔方机器人研究

郑雨辰 王 婷

苏州大学应用技术学院 江苏苏州 215325

**摘 要:**设计了能自动复原三阶魔方的解魔方机器人。提出了“二阶段双向搜索法”,对机器人的研究主要包括以下几个方面:计算机程序求解魔方、单片机程序控制步进电机、摄像头扫描魔方并识别颜色、设计制造金属实物框架和机械手。本文重点讨论了计算机程序求解魔方的思路,即利用大幅度缩短求解时间。该机器人与现有的魔方机器人相比,有机械结构简单、效率高、造价低等优点。

**关键词:**魔方;机器人;二阶段双向搜索

自 1972 年鲁比克教授发明魔方以来,人们探索魔方的解法脚步从未停止。目前国内外魔方爱好者已经研究出一系列的魔方求解算法。本设计在前人的基础上,衍生创新出一种新的求解算法,旨在为求解魔方提供新的突破点,其结构主要包括以下四个模块:①求解魔方的计算机程序;②摄像头识别魔方颜色的计算机程序;③机器人的框架结构及机械手的传动结构;④控制步进电机的单片机程序。

## 1 解魔方求解算法

### 1.1 求解搜索方法

本程序算法的本质是穷举法。

第 1 轮,设定公式步数为 1,有  $6! = 6$  种公式,对给定的打乱状态分别应用这 6 个公式,可得 6 种新状态,若这 6 种状态中出现复原态,则输出相应公式并结束程序。否则进入第 2 轮,公式步数为 2,有  $6^2 = 36$  种公式,搜索是否存在复原态。以此类推,直至穷举出复原态。

这种解法理论上可以解出任意打乱的魔方。但以常见的计算机性能来看,不论是计算时间,还是所需的存储空间,都十分庞大。所以本文提出了“二阶段搜索”这个概念。

### 1.2 二阶段搜索

定义三组状态集合  $G_0$ 、 $G_1$ 、 $G$ :

集合  $G_0$  中仅有一个元素,即魔方的复原状态。

$A = \{U, D, L, R, F, B\}$ ,如果魔方从复原态开始转动,每一步操作仅来自集合  $A$ ,当转动足够多的步数后,所有得到的魔方状态构成了集合  $G$ 。显然,  $G$  是全集。

$A_1 = \{U, D, LL, RR, FF, BB\}$ ,如果魔方从复原态开始转动,每一步操作仅来自集合  $A_1$ ,即对魔方的转动进行限制,左、右、前、后四个面每次只能转动  $180^\circ$ ,当转动足够多的步数后,所有得到的状态都属于集合  $G_1$ 。

三个状态集合的从属关系为:  $G_0 \subseteq G_1 \subseteq G$ 。

打乱状态的魔方属于集合  $G$ ,复原态的魔方属于  $G_0$ 。在上文介绍的穷举法中,由于没有对魔方的转动操作进行限制,不存在  $G_1$ ,直接从  $G$  向着  $G_0$  搜索。记为“ $G-G_0$ ”。

在“二阶段搜索”算法中,“ $G-G_0$ ”的过程被分为了两个阶段:“ $G-G_1$ ”和“ $G_1-G_0$ ”,记为“ $G-G_1-G_0$ ”。 $G_1$  中的每个状态称为“中间状态”。

第一阶段  $G-G_1$ :从打乱状态开始搜索,类似上文提到的穷举法。但是,这里不再是判断新状态是否是  $G_0$ ,而是判断新状态是否属于  $G_1$ ,若发现新状态属于  $G_1$ ,则第一阶段完成。此时可得到两条信息:一个属于  $G_1$  的中间状态  $\{a_i\}$ ,以及一个从打乱状态  $\{a\}$  到中间状态  $\{a_i\}$  的复原公式。

第二阶段  $G_1-G_0$ :类似第一阶段。将  $\{a_i\}$  作为打乱状态,在搜索的过程中判断新状态是否属于  $G_0$ 。该搜索完成后,可得到一个从状态  $\{a_i\}$  到复原态的公式。

两阶段都完成后,将两阶段中各自得到的公式合并,得到从打乱状态  $\{a\}$  到复原态的公式。

由于集合  $G_1$  中的元素不止一个,所以在第一阶段中,只要搜索到任意一个中间状态即可。又由于产生集合  $G_1$  的过程对转动操作进行了

限制,所以  $G_1$  中元素的个数远小于  $G$  中元素的个数。二阶段搜索法对减小计算量有很明显的效果。但这在效率上仍达不到要求。为此,本文提出了双向搜索法。

### 1.3 双向搜索

假设  $G-G_1$  阶段最多需要搜索  $2n(n=1,2,3,\dots)$  步即可完成,我们可以先从打乱状态  $\{a\}$  开始搜索  $n$  轮,即,从步数为 1 的公式开始搜索,直到步数为  $n$  的公式全部搜索完毕,若此时还未搜索到第一阶段的复原公式,则暂停搜索,并将这  $n$  轮搜索过程中产生的所有魔方状态都记为集合  $G_n$ 。并保存每种状态所对应的复原公式。同理,本文将  $G_1$  中的状态  $\{a_i\}$  开始搜索  $n$  步,将这  $n$  步搜索过程中产生的所有魔方状态都记为集合  $G_{n_i}$ 。

对集合  $G_n$  和集合  $G_{n_i}$  取交集,再从交集中任取出一个元素,记为  $\{a_j\}$ 。

通过查表得到由状态  $\{a\}$  到  $\{a_j\}$  的公式,和状态  $\{a_i\}$  到  $\{a_j\}$  的公式。将  $\{a_i\}$  到  $\{a_j\}$  的公式逆推,可得  $\{a_i\}$  到  $\{a\}$  的公式。

将  $\{a\}$  到  $\{a_i\}$  的公式和  $\{a_i\}$  到  $\{a\}$  的公式拼接,得到第一阶段的复原公式。

以上是第一阶段的双向搜索法,第二阶段类似,不再赘述。

实践证明,这种算法大幅度减小了数据量,使计算机程序求解魔方更快捷。

## 2 解魔方机器人控制系统设计

步进电机是一种将脉冲信号转换为步距角的电动机。例如:默认状态下,经过一个脉冲周期,步进电机的主轴旋转  $1.8^\circ$ 。这种电动机可以较为精确地控制旋转角度,适合本项目。

本项目采用 Arduino 单片机作为信号源控制步进电机,其数字 I/O 端口可输出 0V/5V 两种电压,搭配延时函数,可产生脉冲信号。Arduino 程序在接收到复原公式后,逐个解析公式中的字母,向对应的电机发送信号,即可按照预期的动作顺序控制六台电机。

### 3 机器人框架设计与实验调试

整机结构并不复杂。框架由若干竖直、水平的铝合金杆构成,直角处用螺栓连接,方便拆卸。魔方使用空心结构,简化了传动过程。避免了机械卡爪带来的控制部件繁多、结构复杂等缺点。

框架采用欧标 4040 号铝合金;固定板为铝合金板;传动杆采用亚克力板。最终构建出解魔方机器人平台。

### 参考文献:

- [1] (美)Michael Margolis 著,杨昆云译.Arduino 权威指南.2 版.北京:人民邮电出版社,2015.
  - [2] 毛星云,冷雪飞著. OpenCV3 编程入门.北京:电子工业出版社,2015.
  - [3] 濮良贵,纪名刚著.机械设计.9 版.北京:高等教育出版社,2013.
- 作者简介:**  
郑雨辰(1996-),男,汉族,江苏常州人,苏州大学应用技术学院 2014 级机械电子工程,研究方向:机电一体化。