# 将自动机理论变为一门实践课程

Susan H. Rodger

Computer Science

Duke University

Durham, NC 27708

Bart Bressler

Computer Science

Duke University

Durham, NC 27708

Thomas Finley

Computer Science

Cornell University

Ithaca, NY 14853

Stephen Reading

Computer Science

Duke University

Durham, NC 27708

rodger@cs.duke.edu

**摘要**

我们提出了一种在形式语言与自动机课程中通过实践解决课程问题的方法。通过使用JFLAP，学生们可以轻松解决各种非常繁琐的问题，而这些问题通常需要通过纸和笔来进行繁复的运算才能被解决。通过与更为传统的理论问题相结合，学生们可以在自动机理论这个主题上学习更为广泛的知识，解决更为广泛的问题。因此，学生们可以使用JFLAP在计算过程和视觉反馈上探索形式语言与自动机的概念，并且在理论学习上不会依赖JFLAP。此外，我们为JFLAP提出了一个全新的特性：带子程序的图灵机模型。带子程序的图灵机模型通过使用其他图灵机作为自己的一个模块从而帮助人们更方便地构建更复杂的图灵机模型。

**分类**

F.4.3[计算理论] 数理逻辑与形式语言

D.1.7[软件] 编程技术、视觉编程

**通用术语**

理论

**关键词**

JFLAP，自动机，下推自动机，图灵机，文法，SLR文法分析，LL文法分析，L-系统

**1.介绍**

传统上，形式语言与自动机(FLA)课程的书面作业有两种类型：证明题和构建模型的练习题。而第二种类型的问题仅限于一些比较小的例子。即使是构建一个只有八个状态的中等大小的自动机模型，学生也不愿意在这种模型上费劲去手动测试，因为这种测试方式非常的乏味。对这种问题进行评分也非常耗时，而且很容易判断失误。

所以我们提出了一种形式语言与自动机理论课程的实践方式，学生通过这种方式，可以使用JFLAP工具在计算过程和视觉反馈上更方便地探索形式语言与自动机理论的诸多概念。当然我们并不是提倡在课程中取消证明题类型的练习，而是通过这种实践探索的方式来补充课题内容。例如证明这样一个问题：如果L文法是正则的，那么文法(包含所有L文法的字符串的倒序字符串的一种语言)也是正则的。这是课程中常见的一种证明题类型。对于一些学生来说，在证明这个题目之前，先将这个例子可视化，会对证明这个例子有所帮助。学生们可以从这些正则文法L开始，为它们构建一个确定有限自动机(DFA)，然后将这个DFA转换为的DFA。他们必须为这两个DFA创建测试数据以确认DFA是正确的。这种方法让形式语言与自动机的课程与其他的一些计算机专业课的教学方法更加一致，很多计算机专业课是要求学生动手做实验，通过构建，调试以及测试的方法来学习。、

其他人也对形式语言与自动机课程采取了较为类似的实践方式，但只关注其中的较少的一部分课题。Turing’s World[1]中使用的方式让学生可以自行创建图灵机以及自动机并进行实验。这篇文章关注的重点是图灵机以及子状态机。Taylor[8]使用软件Deus Er Machina让用户可以进行图灵机、有限自动机、下推自动机以及其他几种类型的自动机的实验。Forlan[7]是一个与Standard ML结合使用的一个工具集，用于创建和实验有限自动机、正则表达式和正则文法。Language Emulator[9]是一个包含了很多常规的形式语言的工具集，其中包括Moore机和Mealy机，以及各种形式语言之间的转译方式。Grinder[4]开发了FSA模拟器用于进行有限状态自动机的实验。[3]是Webworks的一部分，这是一本涵盖了自动机理论的很多内容的超文本网页书籍，它包含了文本、音频、图片、插画、幻灯片、视频剪辑以及一些自主学习模型。

在本文中，我们提出了JFLAP的概述，然后给出几个例子用于说明如何在计算过程和视觉反馈上使用JFLAP来深入学习形式语言与自动机的概念。接着，我们展示了JFLAP的新特性，包括使用子程序图灵机构建更加复杂的图灵机模型。一个构建好的图灵机可以在其他的图灵机模型中作为子程序来重新命名和复用。最后，我们评估了JFLAP在全世界各地的使用情况，并对未来的工作进行展望。

**2.JFLAP概述**

JFLAP[5,6,2]是一个教学工具，用于创建并实验几种不同类型的非确定性自动机、文法、正则表达式、L-系统，以及不同结构之间的转换。通过JFLAP用户可以构建有限自动机(FA)、下推自动机(PDA)或者多带图灵机(TM)，并且可以观测这些模型在不同输入下的模拟情况。用户也可以输入正则文法、上下文无关文法(CFG)或者无限制文法并观察该文法对字符串的解析过程以及作为结果显示的语法树。

JFLAP允许不同的形式语言和自动机之间互相转换。它可以将NFA转换为DFA再转换为最简DFA，也可以在NFA和正则文法之间进行互相转换，也可以在NFA和正则表达式之间进行互相转换。它可以将不确定性PDA(NPDA)转换到CFG，或者将CFG转换为NPDA，也可以将CFG转换到乔姆斯基范式，同时删除lambda生成式，单一生成式以及其他无用生成式。它可以将CFG转换为LL(1)或者SLR(1)的解析表并且用这个表解析文法中的字符串。最后，它还可以创建L-系统，这是一种特别的文法形式，专门用于建模植物的生长和分形的生成。

1. **使用JFLAP解决问题**

前一节描述了JFLAP中自动机与文法的构造与测试功能，以及它的形式转换功能。这些功能本身允许用户可以用比手工更方便的方式来构建和测试自动机。

现在我们描述其他几种可以用JFLAP解决的问题，而在纸上手动解决这些问题非常费劲。