CPSC 490 Proposal: Operator Automata Theory

Anton Xue
Advisor: Dana Angluin

Abstract

In this project I seek to study finite state machines through the perspective of linear operators. This project will involve developing the formal theory of embedding of finite state machines into function spaces, and using notions of measure and norm to study regular languages.

1 Background

The semantics of programming languages has been of interest to computer scientists since its inception. Roughly, to study the semantics of a computer program is to ask what the program “means”. In particular, three major closely intertwined categories exist:

- **Operational Semantics** is the understanding of how a programming language is meant to be implemented, and ultimately executed on a machine.
- **Axiomatic Semantics** seeks to understand the logical constraints that a program satisfies and generates during execution, and is applied in software verification of program correctness.
- **Denotational Semantics** studies how a program may be interpreted as abstract mathematical models, and allows us to develop techniques for reasoning about execution and correctness.

In this project I am interested in denotational semantics. The study of how a program may be interpreted additionally provides us with tools for embedding different types of programs into first-order logic [5], probability spaces [1] [3], and algebraic structures [2].

In general the problem of program embedding is hard, and straddles a careful line between meaningfulness, feasibility, and practicality. Meaningless and arbitrary interpretations of programs as mathematical objects without proper theory is useless; elegant theories of understanding probabilistic programs through the lens of measure theory exist, but actual implementations often resort to potentially inaccurate numerical methods or incomplete symbolic computation; first order theories are often undecidable, and when they are, tend to have exponential complexity at best. Nevertheless, any such successful endeavors, be they theory or implementation, are important advances in the field of programming language research.

One particular mathematical abstraction is linear spaces, which is widely used throughout both pure and applied fields of mathematics, science, and engineering. Such spaces are the fundamental objects of study in linear algebra, for which theory and practice strikes a balance of generality and applicability. An appeal of linear spaces is that they are simple, for which intuition is often geometric and interpretations straightforward.
However, linear spaces have generally eluded the grasp of programming language researchers. Outside of certain restricted forms of programs, general computer programs often appear to be highly non-linear, and lack obvious interpretations in terms of vectors and linear transforms. As such, programming language semantics research is often focused on rigorous formalization of execution semantics, mechanized proofs of program properties, and applications of formal logic, among others. With few exceptions, the field generally does not utilize linear spaces as an inspiration when attacking domain-specific problems. Despite this, linear spaces are still enticing, and one should not forget them. In this project, I aim to study the embedding of finite state machines [4] into linear spaces, and apply a combination of linear algebra, functional analysis, and combinatorics to study such machines and the regular languages that they generate. I plan to extend my preliminary results on the development of non-counting based measures on regular languages. Such notions allow us to compare more complicated relations between regular languages beyond simple ones such as inclusion, and also allow us to discuss measure-theoretic notions of approximation for languages.

2 Project Outline

In this project I plan to tackle the following problems:

- Explore embeddings of finite state machines into well-established algebraic structures.
- Adapt techniques from linear algebra and mathematical analysis to reason about finite state machines under algebraic representations.
- Devise algorithms that manipulate finite state machines under algebraic representations, and prove bounds on them.

The deliverable for the project will be report of my findings and progress.

References


