Visualization of Non-deterministic Pushdown and Finite Automata Derived From Context-Free Grammars

| Nicolas Hempe Ardila  Department of Computer Science & Engineering  University of South Florida  Tampa, FL, USA  nicolashempe@mail.usf.edu | Spencer Turkel  Department of Computer Science & Engineering  University of South Florida  Tampa, FL, USA  sturkel@mail.usf.edu |  |
| --- | --- | --- |

# ABSTRACT

In this project we are using a web application to visualize a context-free grammar into non-deterministic finite automata (NFA) and non-deterministic pushdown automata (NPDA). The application will take a context-free grammar in a form on the webpage and dynamically update the graphs as you enter the grammar. Each graph will be displayed in the webpage to show the solution.

# INTRODUCTION

The web application, written using TypeScript and Angular, displays a form that will take in a context-free grammar using a text box on the left side for non-terminals and the right side for productions. As the grammar is updated, new text boxes are created to allow the user to expand the grammar. As the user types each character of the grammar, a visualization of an NPDA and NFA (as applicable) will be dynamically drawn, so that the user can view how changes to the CFG affect the automata.

# Methodology

We are using a context-free grammar to construct NFA and NPDA state diagrams to visualize the states and transitions produced by the grammar.

## Non-deterministic Pushdown Automata (NPDA)

A non-deterministic pushdown automaton has a similar application as non-deterministic finite automaton, but the use of stacks allows the automaton to solve context-free languages that require counting. The stack represents the state of the current symbol, the input symbol, and the symbol at the top of the stack

Non-deterministic pushdown automata consist of a 7-tuple (Q, ∑, Γ, q0, Z0, F, δ):

Set of states denoted by Q

Input alphabet denoted by ∑

A stack alphabet denoted by Γ

Start state denoted by q0, q0 ∈ Q

Start symbol denoted by Z0, Z0 ∈ Γ

Transition function denoted by δ

Set of accepting states F, F ⊆ Q

# Results

The following example (Figure 1) shows the Nondeterministic pushdown automaton and Nondeterministic finite automaton with the context-free grammar:

S⟶aA

A⟶abS | λ

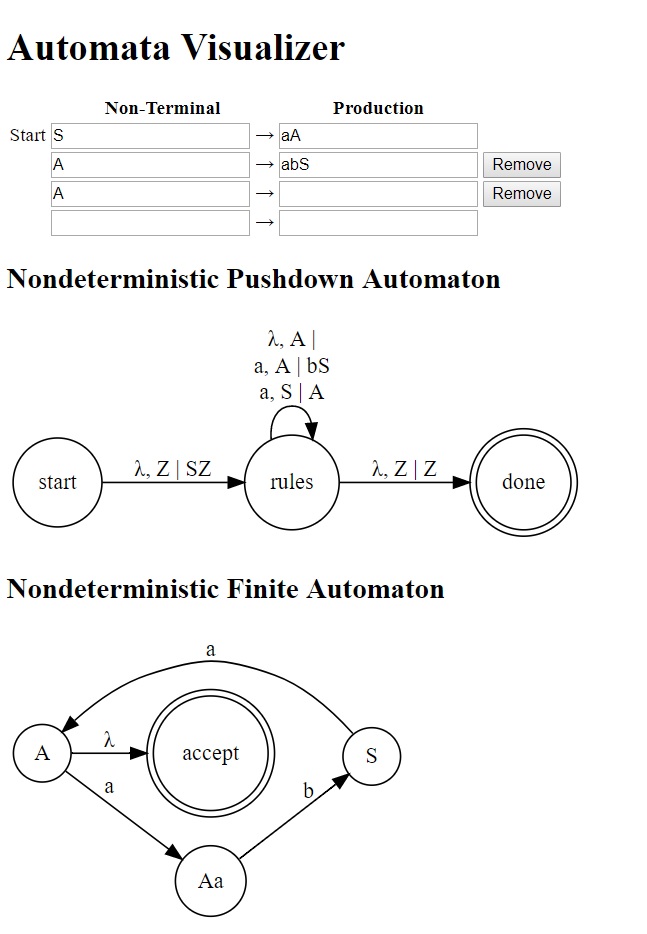


Figure 1: Automata Visualizer

The next example (Figure 2) shows the Nondeterministic pushdown automaton and Nondeterministic finite automaton with the following grammar:

S⟶aA

A⟶bS

B⟶aA | bBS

This example shows that the NFA does not accept this grammar since there is no state that leads to the final state. This is caused by the fact that the given grammar is not linear.

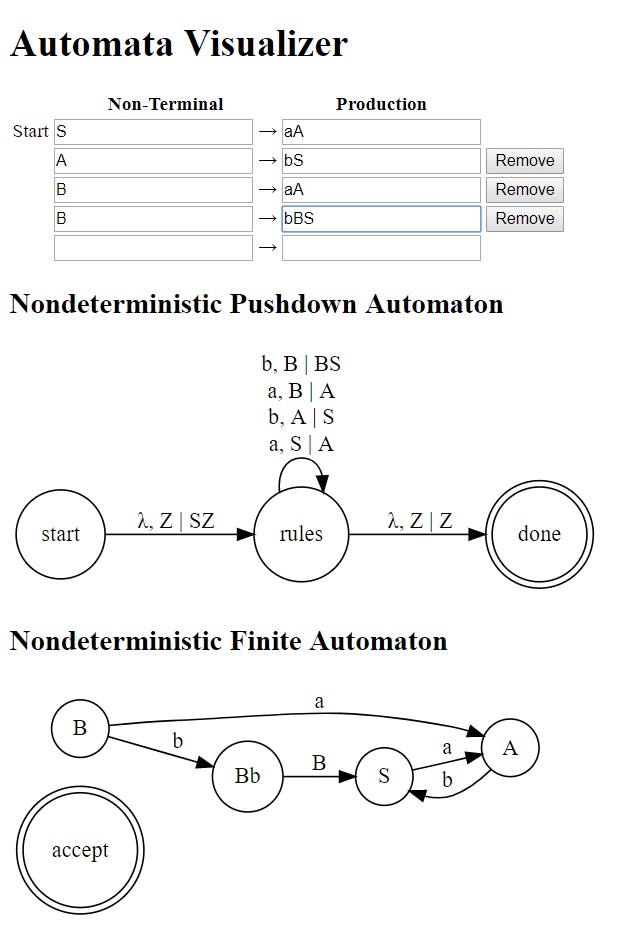


Figure 2: Automata Visualizer

# Conclusion

The authors found that both the construction and use of the tool vastly improved their intuition about the relative power of NPDAs over NFAs.

The construction of the application also revealed the shortcomings of the descriptions of related theorems available on the internet and in the course textbook on conversion between grammars and automata. These descriptions were suitable for a human to read and understand, but were highly abstracted to the point that, during construction of the application, they served more as guidelines rather than algorithmic references.

Further improvements to the application could range from giving more verbose information on why a grammar is not linear, to including step-by-step simulations of running each automaton against a string.