

Applied Formal Methods

Final Project: SatRubik

Nathan Scheirer, Patrick Sivets

Date: 10/26/17

1. Project Plan

1. Define your group and the parameters of your project.
 - (a) The SatRubiks group will consist of two members with an end goal of creating a program in the SAT solver, nuXmv, that will solve any initial condition of a Rubik's cube. This project will adhere the precise rules of a Rubik's cube and will begin with the solving of a 2x2x2 cube (for simplicity) and then followed by a 3x3x3 cube.
2. Who are the members of your group?
 - (a) Nate Scheirer and Patrick Sivets
3. What is your group name?
 - (a) SatRubiks
4. What formal method are you using?
 - (a) We will use Satisfiability planning to solve the Rubik's cube.
5. What specifications will you verify?
 - (a) The rules that we formulate for the Rubik's cube will need to be verified before the program can be considered accurate.
6. What system will you analyze?
 - (a) A Rubik's cube operates in a very specific way so these actions will be the system that we will analyze. (i.e. there are six different possible rotations for a 3x3x3 cube)
7. What does a success look like for your project?
 - (a) A success is defined as our nuXmv program being able to return a valid counterexample representing a series of rotations required to solve the Rubik's cube in any initial state. We will begin the project solving a 2x2x2 cube with the hopes that we will have quick success allowing us to move forward and solve the 3x3x3 cube problem.
 - (b) A side goal will be to determine the least amount of moves required to solve each initial state.
8. How will you demonstrate your analysis?
 - (a) Benchmarks:
 - i. Number of moves, from the calculated counterexample.
 - ii. Number of unique moves, from the counterexample. (i.e. How many moves are repeated or unnecessary?)
 - iii. Run time, from nuXmv.
 - (b) Demo:

- i. We will demonstrate the nuXmv program that we create in class and will report the results in our final project.
- (c) Results:
 - i. We will utilize the output of nuXmv and its various commands to analyze our model to verify its correctness and accuracy. This output will also help us to provide details on performance and efficiency.
- 9. Logistics:
 - (a) Repo structure:
 - Master (Only merged via pull requests from Develop and will always be in working condition)
 - ↑
 - Develop (Working condition but contains changes made by each group member)
 - ↑
 - Feature or bugfix name* (Branches for new features, tests, or bug fixes. Can only be merged to Develop via pull request)
 - (b) Members should check that all the changes made and merged to Develop continue to allow the program to operate before the PR is merged. Documentation and specifications should be up to date as the PR is created as well.
 - (c) The group will meet twice a week for at least an hour per meeting. Each member is expected to put at least a total of nine hours of work into the project a week.
- 10. Timeline:
 - (a) 11/3 - Perform research into work previously done on solving Rubik's cubes with SAT solvers, define the Rubik's cube specifications, and begin writing a program to solve the 2x2x2 problem in nuXmv.
 - (b) 11/10 - Finish defining specifications and produce preliminary results for the 2x2x2 problem (possibly 3x3x3). Write progress report and preliminary results.
 - (c) 11/17 - Refine the 2x2x2 problem and finish, work on porting to the 3x3x3 should be fully underway, and specifications should be concrete at this point.
 - (d) 11/24 - Continue work on solving the 3x3x3 problem.
 - (e) 12/1 - Preliminary results should be close to verified and work on the final report should commence.
 - (f) 12/5 or 12/7 - Finish the 3x3x3 problem, the report, and presentation.