

CS344 Project Specification

Laveen Chandnani - 2004842

1 Title

An analysis of strategies in the re-pairing game.

2 Problem

A Dyck word is a string consisting of opening and closing brackets, in such a way that at any point in the string there are at least as many opening brackets as there are closing brackets on the left side of this point. Dyck words have some interesting combinatorial properties but also have their applications in theoretical computer science, more specifically in the context of formal languages.

The paper by Chistikov and Vyalyi [1] explores a one-player game called the “re-pairing game”; given a Dyck word, a move consists of taking an opening bracket and pairing it with any closing bracket to its right, then erasing them. This process is repeated until the string no longer has any characters, and the width of a play (or strategy) is the maximum number of nonempty segments of symbols seen during a play. The paper gives a simple re-pairing strategy for Dyck words, proves upper and lower bounds for the width of this strategy, and beats this strategy for Dyck words representing complete binary trees. This game is then linked back to a question in automata theory.

One of the open problems at the end of this paper is computing the width of a general Dyck word. This project will conduct a survey on the re-pairing game, explore established results from the paper and attempt to make progress on the given open problem.

3 Objectives

The first half of this project will focus on surveying the results from the paper. This involves covering the relevant background knowledge to understand the paper, and dissecting and expanding on the proofs for any theorems/lemmas/conjectures in the paper in an attempt to get a deeper understanding.

The second half of this project will focus on the open problem. This involves establishing the research landscape on the re-pairing game and any similar literature, and attempting to make progress on the problem itself.

4 Methodology and technical considerations

In order to make progress on the problem, we will start with an experimental approach. This will involve writing code in order to exhaustively apply a variety of strategies to Dyck words of a fixed length, and analyse the strategies which give a minimal width (as there could be multiple strategies which work just as effectively as one another).

We will be using Python as the language of choice due to its simplicity; the focus is on the results of the experimental approach, not so much on the implementation itself. I'm also most familiar with Python and so this would reduce the time needed to write the code for the experiment.

All code will be written using VSCode and will be stored on github for backup and source control. This acts as a contingency in the event of data loss or any other problems which could occur.

5 Timetable

Week	Task
Term 1 Week 2	Submission of project specification
Term 1 Week 2-5	Understanding the relevant literature and dissecting proofs
Term 1 Week 5-8	Writing first half of report
Term 1 Week 8-9	Writing up progress report
Term 1 Week 9	Submission of progress report
Term 1 Week 9-10	Begin thinking about exhaustive search implementation in Python
Christmas Holidays	Write up Python code, collect results and begin to analyse
Term 2 Week 1	Continue analysis of results from Python code and meet with supervisor
Term 2 Week 2-7	Further work on the open problem and writing second half of report
Term 2 Week 8-10	Project presentation and finishing up final report
Easter Holidays	Polishing up final report
Term 3 Week 1	Submission of final report

6 Legal, Social, Ethical and Professional issues

This project does not involve any such issues.

References

- [1] Chistikov, D. and Vyalyi, M. (2020) “Re-pairing brackets” Available at [<http://wrap.warwick.ac.uk/137788/1/WRAP-re-pairing-brackets-Chistikov-2020.pdf>] (Accessed 11 October 2022)