PFP Project Proposal Fastest Scrabble Solver (Short Name: Scrabble)

Helen Chu (hc2932), Alexander Lindenbaum(al4008)

November 29, 2022

1. Project Objective

We aim to implement a parallelized approach to "The World's Fastest Scrabble Program" algorithm proposed by Andrew W. Appel and Guy J. Jacobson [1]. We break our deliverables into two main parts:

1. Implementation of the data structure Directed Acyclic Word Graph (DAWG) to store Scrabble dictionary in a compact manner

2. Parallel implementation of the paper's word generation algorithm to return all possible moves given a board state and rack of tiles [2]

If time does not allow for DAWG implementation, we will store the Scrabble dictionary in a Trie.

2. Approach

The algorithm stores a "cross-check set" of valid characters for each square on the board. Since words must be placed next to an existing word, we look to "anchor" the new word next to an existing tile on the board. Thus, for every possible anchor square, we generate all possible prefixes using the anchor square, then find all matching suffixes of the prefixes by traversing the DAWG.

There are several parts of this algorithm that can be parallelized; our current plan is to iterate through the possible "anchor squares" in parallel.

3. Algorithm Walk through

A rough outline of the algorithm is as follows:

- Convert Scrabble Dictionary to DAWG/Trie format.

- On input of a Scrabble Board as rows of square entries, each square either consists of 1. A tile already placed on the board or 2. A cross-check list for valid letters for that square.

- Given K cores, split up all possible anchor squares (those next to an existing tile) into K sets.

- For each set of anchor squares:

  - Find all possible prefixes through comparing cross-check lists, and their matching suffixes through traversal of the DAWG. Upon reaching a terminal state, store the current string as a valid word.

- Return all set of valid words.

4. References

[1] Andrew W. Appel and Guy J. Jacobson. The world's fastest scrabble program. *Commun. ACM*, 31(5):572–578, may 1988.

[2] Aydin Schwartz. Coding the world's fastest scrabble program in python, Apr 2022.