

Smarter and simple Scrabble strategy

Frej Connolly
Götgatan 78 13TR LÄG1302
118 30 Stockholm
SWEDEN
+46(0)73-963 41 90
connolly@kth.se

Diana Gren
Sköldgatan 8 2TR
118 63 Stockholm
SWEDEN
+46(0)70-467 47 20
dianagr@kth.se

Course: DD143X
Supervisor: Johan Boye
Kungliga Tekniska Högskolan
CSC
March 7, 2012

Abstract

The abstract goes here.

Sammanfattning

Sammanfattning på svenska hamnar här.

Contents

1	Introduction	3
1.1	Problem statement	3
1.1.1	Question	4
1.2	Terminology	4
2	Background	5
2.1	Scrabble game rules	5
2.2	Research	7
2.2.1	Dictionary representation	8
2.2.2	Word generation algorithm	8
2.2.3	Game strategies	10
3	Implementation	11
3.1	Game limitations	11
3.2	Agents	12
3.2.1	High Score Word player	12
3.2.2	Bonus Square player	12
3.2.3	Balance On Rack player	13
4	Results	14
4.1	Test conditions	14
4.1.1	Games	14
4.1.2	Dictionaries	14
4.2	High score words vs Balance	15
4.3	Balance vs Bonus	16
4.4	Bonus vs High score words	17
5	Conclusions	19
5.1	Discussions	19
5.2	Conclusions	20

A Game conditions	21
B Diagrams	22

Chapter 1

Introduction

Scrabble is an old classic game that is fairly easy to learn. However, it is extremely difficult to be a great player. Compared to other classic board games, such as chess, there are more factors to take into consideration when making a move in Scrabble.

Apart from anagramming and generating words, there are other crucial decisions to make. A player would probably find several legal moves in one round, and then would have to decide which one to place. Choosing the move that generates the highest score is not always the ideal way to go. It could be that making such a move would create a situation in the next round where nothing could be done, or let the opponent score high.

Making the choice of which move to use include many factors, and there are several different techniques and strategies one can follow to make the decision easier. Picking up the bonus squares can generate an extremely high score, hence one strategy is to try to always pick up the bonus squares and prevent the opponent from doing the same. Another way of being successful is to prioritize using letters with high score, since they would not only generate a high total score to the player, but could also create difficult situations in the future if not used as soon as possible.

There are several more possible strategies one can follow, and this study aims to investigate a three of them trying to establish which one is the more rewarding.

1.1 Problem statement

Scrabble requires the players to have a good vocabulary and to become a successful player, some knowledge about how many tiles of each letter is available is preferable.

When playing, the ability to keep a good balance between consonants and vowels on the rack can profit future moves. Destroying bonus possibilities for the opponent, and at the same time score them oneself is a crucial skill, and experts are probably aware of which tiles have been placed, and those that remain, allowing them to anticipate the future sequence of events.

1.1.1 Question

Which of the strategies has the most impact on a game? How do the strategies perform against each other, and which one is the most successful one?

1.2 Terminology

Anchor square: Vacant board square adjacent to a placed tile.

Bag Tiles that have not been drawn by players.

BOR: Balance on rack player.

BS: Bonus square player.

Cross check set: Set of possible letters on an anchor square with regard to adjacent word going vertical from named square.

DAWG: Directed acyclic word graph.

HSW: High score word player.

Rack Tiles on a player's hand.

Chapter 2

Background

2.1 Scrabble game rules

Scrabble is word game were two to four players are given a set of letter tiles to form words to be placed out on a four-cornered board of squares [10] [11] [9].

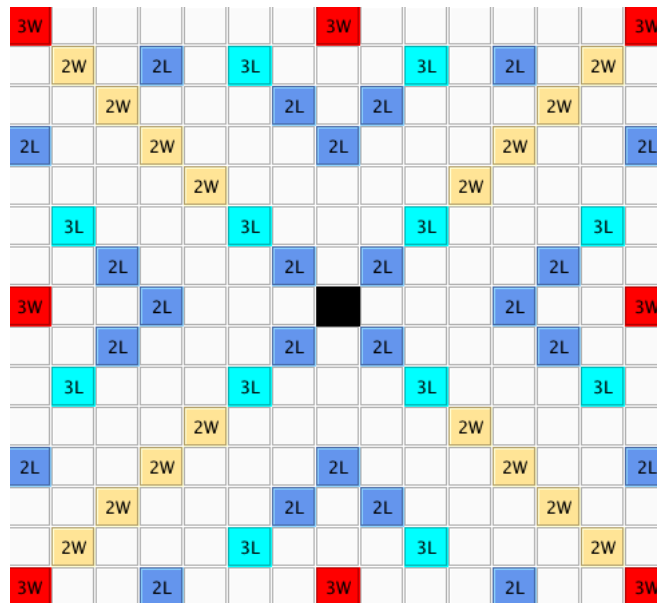


Figure 2.1: Game board. 15x15 square board with bonus squares.

Game components

Scrabble consists of

- a 15x15 board with bonus squares and the center marked out. fig. 2.1.
- a set of letter tiles

The players are given 5-8 tiles each to keep on their *rack*. Each letter reward points. Common letters gives lower points. The following list shows letter points and the number of occurrences of each tile in the bag.

- 1 point **A**x8, **D**x5, **E**x7, **I**x5, **L**x5, **N**x6, **R**x8, **S**x8, **T**x8
- 2 points **G**x3, **H**x2, **K**x3, **M**x3, **O**x5
- 3 points **F**x2, **V**x2, **Å**x2
- 4 points **B**x2, **P**x2, **U**x3, **Ä**x2, **Ö**x2
- 7 points **J**x1, **Y**x1
- 8 points **C**x1, **X**x1
- 10 points **Z**x1

Objectives

Each round the player has to construct a word consisting of at least two letters by using the tiles on the *rack* and the board. Words can be placed horizontally or vertically. Each word has to be placed adjacent to at least one tile already on the board. The player starting has to position the first word so one letter is on top of the center square.

Turn

Each turn a player can choose from one of the following.

- Lay out a word onto the board. Then pick up new tiles from the bag, the same number as used from the rack so the player have a total of 8 tiles again.
- Pass, receiving 0 points.
- Exchange a number of tiles from the rack with the bag.

Bonus squares

There are four different types of *bonus squares* dispersed throughout the board. When a tile is placed on top of one of them the player is rewarded a bonus score for either that particular tile or the whole word. Letter bonus is calculated by multiplying the point for the corresponding letter on top of the bonus square by either two or three. Word bonus is multiplied by two or three to the total score of the word. If a tile on a word bonus square is part of both a horizontal and vertical word, the bonus applied to both each of them respectively. If a word is positioned over both letter and bonus tiles, the score of the letter bonuses is calculated first before multiplying the word. Bonuses can only be used once when a tile is placed on the square.

2L: Double letter

3L: Triple letter

2W: Double word

3W: Triple word

If a player uses all the tiles on the rack an additional 50 points is rewarded the player, after an eventual bonus score.

End of game

The game ends when a player has emptied the rack, and there are no tiles left in the bag. Also, when there has been a total number of four zero points moves in a row, i.e. the players passed or exchanged tiles.

After the game, all the players still having tiles on the rack get a penalty. The sum of the letter scores on the rack is removed from the total score and the winner of the game is the player with the highest score in the end.

2.2 Research

As mentioned above, this is a well studied problem, and many papers and articles are to be found where different problems in the game of Scrabble have been studied. This paper will mostly refer to Appel and Jacobsen [2] who discovered a way to represent the dictionary so that it takes up minimum space in memory.

2.2.1 Dictionary representation

Words from a dictionary can be stored by representing the letters in trie. Each path can represent several words by letting the nodes mark the end of a word Figure 2.2.

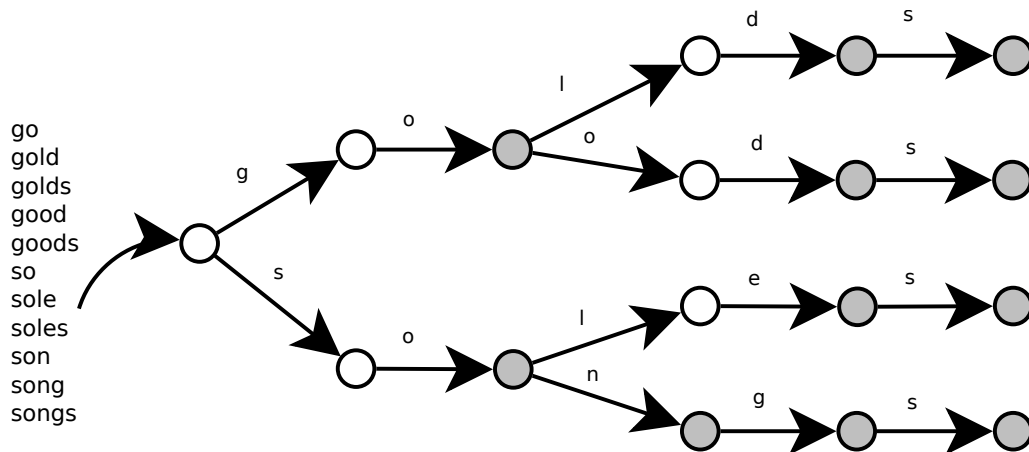


Figure 2.2: Example of a small dictionary consisting of 11 words represented with a trie data structure. End of words marked with grey nodes.

Appel and Jacobsen showed that the dictionary representation size can be reduced with a *Directed Acyclic Word Graph*, referred to as a *DAWG* [2]. The DAWG can be constructed by first creating a trie and minimizing it by finding cases where two or more words can share a letter (node), creating a new edge from the previous node in one of the words to the other words node. Then remove the unnecessary edge and node which is no longer in use 2.3. A trie has a lot of redundancy, because of edges and nodes being identical, no such thing occurs in a DAWG. All the identical are removed, and reduced to only one occurrence. In the dawg example fig. 2.2 there is 10 nodes and 12 edges. That is significant less than the 17 nodes and 16 edges it consumes in a trie fig. 2.2. Another example is illustrated in the mentioned paper. A dictionary consisting of more than 100 000 words taking up a memory space of 0.5 MB with a trie, could be reduced to occupy only 175 kB with a DAWG.

2.2.2 Word generation algorithm

Finding and forming words to lay out on the board is difficult. Appel and Jacobsen presented a solution to the problem by reducing it to one dimension. Instead of searching both up and right it is possible to generate words only

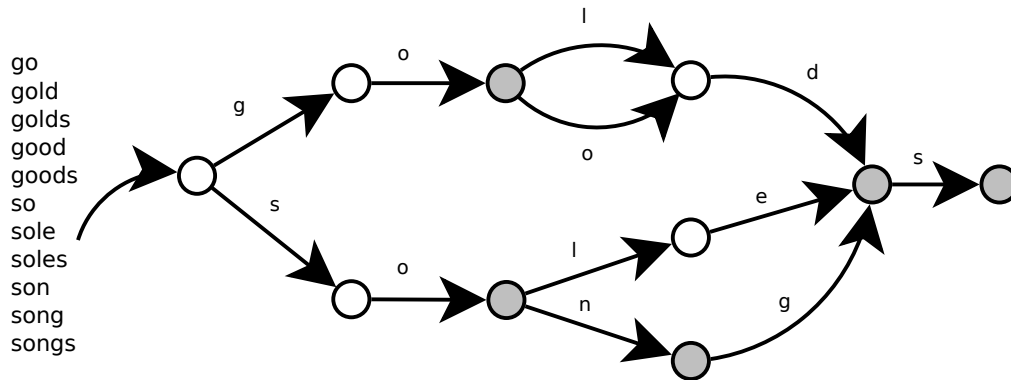


Figure 2.3: Example of a small dictionary consisting of 11 words represented with a DAWG. End of words marked with grey nodes.

horizontally. The argument is that generating a word vertically is basically the same thing as generating a word horizontally. The only difference is that the board is transposed. Therefore, the algorithm to generate words is limited to only generate horizontally, and for each move we do two searches, where one is over the transposed board. [2]

Anchor squares

A key in the algorithm implemented are the *anchor squares*, which are the empty squares next to a non-empty square as can be seen in figure 2.4. These are important since words can only be built from already existing tiles. In the very first move there is only one anchor; the center square, since the player making the first move always has to start in the center square.

Cross-check sets

When placing a word horizontally, the vertically placed letters also have to create a legal word. It is quite easy to establish that if we place a word horizontally, the vertical word can increase with only one letter at a time. This makes it possible to calculate, for each anchor square, which set of letters that are possible to place at that square. The calculations can be made before each move, and allows the player to place a word by row, without considering the rest of the board. The set of available letter for one square is in Appel and Jacobsen's paper [2] referred to as a *cross-check set*



Figure 2.4: Anchor squares. The adjacent tiles are the anchor tiles from which the word generation begins.

2.2.3 Game strategies

If one is a beginner at Scrabble, there are many small tricks to learn to advance quickly. Some of the tricks can be listed as:

- Difficult letters. Letters with high score are usually more difficult to place and should be used as soon as possible.
- Balance on the rack. It is easy to get stuck with only consonants or only vowels on the rack, keeping a good balance can therefore be a good idea to avoid deadlocks in the future.
- Bonus squares. Hitting the bonus squares can generate a high score. Although it is preferable to avoid opening opportunities for the opponent to do the same.
- Word extensions. According to Sheppard [1], the ability to find extensions (i.e. prefixes and suffixes) to already placed words, is a key to being a successful player since such moves can generate very high scores.
- Vocabulary. There are many short words in the language, and it is instantly rewarding if a player can learn many short word letters.

Chapter 3

Implementation

The study is based on an implementation that follows the example of Appel and Jacobsen [2], with some small modifications. Since implementing a DAWG seemed like a time consuming project, a choice was made to stay with the trie. The advantage with a DAWG is that it saves a lot of memory, but there were no problems with memory space, and therefore an unnecessary thing to focus on.

In the implementation there were some game limitations made to make the study fit into the given time span. The game limitations are described in section 3.1.

Three different agents were implemented with different strategies to follow and they are all described in section 3.2.

The analysis and the final results are shown in section 4.

3.1 Game limitations

Almost all tournament play involve only two players. The tests is therefore run with two agents. There exists a large variation of the game rules. The rules used is a slightly simplified version of the most common rules used by clubs and tournament play worldwide. TODO lägg in källa till regler.

To limit the work load to fit into the time span, some limitations were made to the game. This study focuses on games played by only two players, and the implementation is therefore customized to handle only two player games. The players have two options during a turn; either lay a word, or pass the turn and receive zero points. In other words; the possibility to change tiles if no word can be found is removed.

The bag is also minimized, and do not contain any blank letters or other special tiles that are not letters. The result is that a Q or a W never can be

used. In addition, the game is played in Swedish by the agents.

3.2 Agents

To see which of the very basic strategies listed in section 2.2.3 is the most successful one against the others, three agents with different strategies were implemented. They would play against each other in order to generate results for later analysis. This section describes the three different agents implemented, and their strategy in the game. In order to evaluate which strategies are the more rewarding, the agents are implemented with only *one* strategy. An agent who calculates the total score of a move and chooses the highest one will easily win over the tested agents in this study. The agents will be referred to as the HSW player, the BS player and the BOR player (see section 1.2).

3.2.1 High Score Word player

One strategy to follow when playing Scrabble is to exploit the fact that some letters are worth more than others. Letters with higher scores are less common in the language and therefore more difficult to use. Naturally, it is preferable to use them as soon as possible, to not risk difficult situations later in the game. The HSW player prefers placing words including high score letters. By all legal moves generated, the HSW player will choose the one where the word itself has the highest score. The HSW algorithm is as follows:

```
function ChooseMove(inputMove)
  if score of inputMove.word greater than highestScore
    nextMove <- inputMove;
  end if
end function
```

The HSW player saves one move. If a better move is found than the one saved, the new move will overwrite the old one.

3.2.2 Bonus Square player

Sometimes, it can be more rewarding to place a relatively short word than a longer one. This is because of the bonus squares spread out on the board. They can multiply the value of either one letter, or the entire word placed, and can therefore generate quite high scores. The BS player strives to place words over bonus squares, to hopefully generate a high score. If there are

several opportunities to reach a bonus tile, the most rewarding one is chosen by the agent, i.e. the word reaching the bonus square with the highest bonus.

```
function ChooseMove(inputMove)
    bonus <- 0;
    for each letter in inputMove.word
        squareBonus <- bonus for letter.square;
        if squareBonus greater than bonus
            bonus <- squareBonus;
        end if
    end for

    if bonus is higher than highestBonus
        nextMove <- inputMove;
        highestBonus <- bonus;
    end if
end function
```

3.2.3 Balance On Rack player

An important thing to think about when playing Scrabble is to plan for the next move. If a player ends up with only consonants on the rack, the possibility of laying out a word is reduced. The BOR player tries to always keep a good balance between vowels and consonants on the rack, to avoid deadlocks later in the game. The agent knows the *ideal ratio* and tries to lay out words so that the words left on the rack returns a ratio as close to the ideal ratio as possible. Tests with different ideal ratios are made.

```
function ChooseMove(inputMove)
    ratio <- number of vowels on rack /
                number of tiles on rack;

    difference <- abs(ratio - idealRatio);

    if difference is less than ratioDifference
        nextMove <- inputMove;
        ratioDifference <- difference;
    end if
end function
```


Chapter 4

Results

Each of the agents has been tested against the others, to let us evaluate the impact of each strategy. Explanations of the agents can be read about in section 3.2. Section 4.1 describes under which conditions the tests were made.

Sections 4.2, 4.3 and 4.4 show the results from each run.

4.1 Test conditions

4.1.1 Games

In one run the agents were set to play 1000 games against each other and to minimize the impact of opening the game in the results, both agents started the game 500 times. Each run were made 10 times, so the total result is based on 10 000 games. It was established that there were no significant changes in the results between running 1000 games, and 10 000 games, and would therefore probably be redundant to make more tests.

4.1.2 Dictionaries

Two dictionaries were used in the tests; one consisting of only the normal form of each word and the other consisting of all possible forms of each words, where the latter is a set of 480391 words. The former consists of 83247 words. The rules of Swedish Scrabble do not allow words of all forms, but the english does. Therefore, it seemed interesting to compare how the agents would perform with different conditions. There could be situations where for instance the bonus player has an advantage if conjunctions are allowed, since it strives always after covering the bonus squares. The large

dictionary could allow the player to place short extensions to already placed words.

The longest word in the used dictionary is ammoniumdiamintetrakistiocyanatokromat. The shortest possible word to place on board is all two letter words Section ?? . The number of two letter words is 121. Table 4.1.

	Base words	All words
Mean	9.5	11.2
Longest	ammoniumdiamintetrakistiocyanatokromat	
Shortest	121 two letter words	

Table 4.1: Mean, largest and smallest lengths of legal words in the dictionary.

4.2 High score words vs Balance

The results from runs over the large dictionary and the small dictionary do not differ that much, as can be seen in figure B.1 and figure B.2. The BOR player uses a ratio between vowels and consonants on the rack, and strives to keep the ratio after each move. The wanted number of vowels on hand will be referred to as *vowel ratio*. Several tests were made with the BOR player to see if there is any difference when altering the ratio parameter. The ratio is calculated as the number of vowels left on hand divided into the number of consonants left on hand. Table 4.4 shows the results of the runs made with different ideal number of vowels on hand.

	BOR	HSW
Wins	1	9999
Draws	0	0
Wins when started playing	0	5009
Mean score	124	420
Median score	123	415
Highest score	299	745
Lowest score	-16	-12

Table 4.2: Result from 10 000 games with vowel ratio 1/8 and words in all inflection forms from the Swedish dictionary (480 391 words).

The results show that the HSW player is generally more successful playing the game. The BOR player does not take the word score in consideration when playing, but only the ratio, and it is obviously not the most rewarding

	BOR	HSW
Wins	1206	8762
Draws	32	32
Wins when started playing	643	4431
Mean score	214	309
Median score	213	307
Highest score	407	624
Lowest score	-9	-19

Table 4.3: Result from 10 000 games with vowel ratio 8/8 and words in all inflection forms from the Swedish dictionary (480 391 words).

strategy to follow. An interesting observation is that if the BOR agent tries to keep a ratio of vowels close to 1, i.e all tiles being vowels, there is a significant peak in the number of winning games for the agent. On the other hand, trying to keep one vowel on hand after each round seems to not be as smart, since the agent lost almost 100% of all games.

Vowel ratio	HSW wins	BOR wins
0/8	9922	73
1/8	9999	1
2/8	9484	507
3/8	9777	212
4/8	9735	262
5/8	9627	368
6/8	9146	838
7/8	9863	134
8/8	8762	1206

Table 4.4: Winnings over 10000 games depending on vowel ratio. The ratio is defined by the number of vowels divided by the number of tiles on rack.

4.3 Balance vs Bonus

The BOR player continues to perform doubtly against the BS player. Also here the BS player consideres what bonus it will be given in a certain move, and chooses the highest one. Naturally the BOR player that only cares about its ratio on the rack will perform worse. The results from runs with different vowel ratio in the BOR player can be seen in table 4.5.

Note that the BOR player is stable with its game results. In figure B.2 one can see that the BS player in some games made an extremely poor performance, receiving zero points or less, while the BOR player keeps the graph quite thin, i.e. the results differ less from game to game.

Vowel ratio	BS wins	BOR wins
0/8	9925	72
1/8	9987	12
2/8	9670	325
3/8	9829	165
4/8	9818	172
5/8	9731	261
6/8	9370	620
7/8	9859	141
8/8	9074	915

Table 4.5: Winnings over 10000 games depending on vowel ratio. The ratio is defined by the number of vowels divided by the number of tiles on rack.

4.4 Bonus vs High score words

BS wins just over 6 of 10 times against *HSW* when using all inflection forms from the Swedish dictionary table 4.6 and well over 5 of 10 times when only the base forms of words is used table 4.7. When *BS* started playing it won 52% of its wins when using all inflection forms. For *HSW* 53% of its wins. When using only the base form both of them were winning 51% its wins when started laying out words.

The resulting mean, median, highest and lowest scores is significant higher when using all inflection forms. Lowest score is negative for both players when using only base forms. *BS* mean score is 24% higher in table 4.6 compared to table 4.7. The same applies for the end score which is higher when using all words Appendix Figure B.3.

	Bonus Squares	High score words
Wins	6104	3857
Draws	39	39
Wins when started playing	3148	2040
Mean score	308	278
Median score	301	273
Highest score	724	606
Lowest score	85	61

Table 4.6: Result from 10 000 games with words in all inflection forms from the Swedish dictionary (480 391 words).

	Bonus Squares	High score words
Wins	5427	4521
Draws	52	52
Wins when started playing	2758	2314
Mean score	248	240
Median score	246	234
Highest score	488	506
Lowest score	-15	-11

Table 4.7: Result from 10 000 games with only words in base form from the Swedish dictionary (83 247 words).

Chapter 5

Conclusions

5.1 Discussions

BOR player

The BOR player is significantly better with a vowel ratio of 8/8. It can probably be explained from the fact that when choosing a move, and the vowel ratio is 8/8 (1) it is easier for the agent to place longer words. Imagine the agent placing a word using all tiles but one, and the remaining tile on hand is a vowel. This would be a preferable move if the ratio is 1, but not if the ratio is less. If the ratio would be 4/8 for example, the agent would choose a move that leaves two tiles, where one is a vowel, over the move with a longer word.

HSW player

High score word player conclusions go here.

BS player

Choosing to place words that will occupy a bonus square is rewarding. It will not only give a high score but will also destroy the possibility for the opponent to reach a bonus square.

It has active playing style with the mission to conquer the bonus squares at all cost. With the power of a big dictionary it will have a higher probability to reach a bonus. It is easy to extend words already on board when having the possibility to use all inflection forms of words. The end score will also be higher Figure B.3. When using only base words it will struggle to find bonus squares. This is due to

5.2 Conclusions

Appendix A

Game conditions

Appendix B

Diagrams

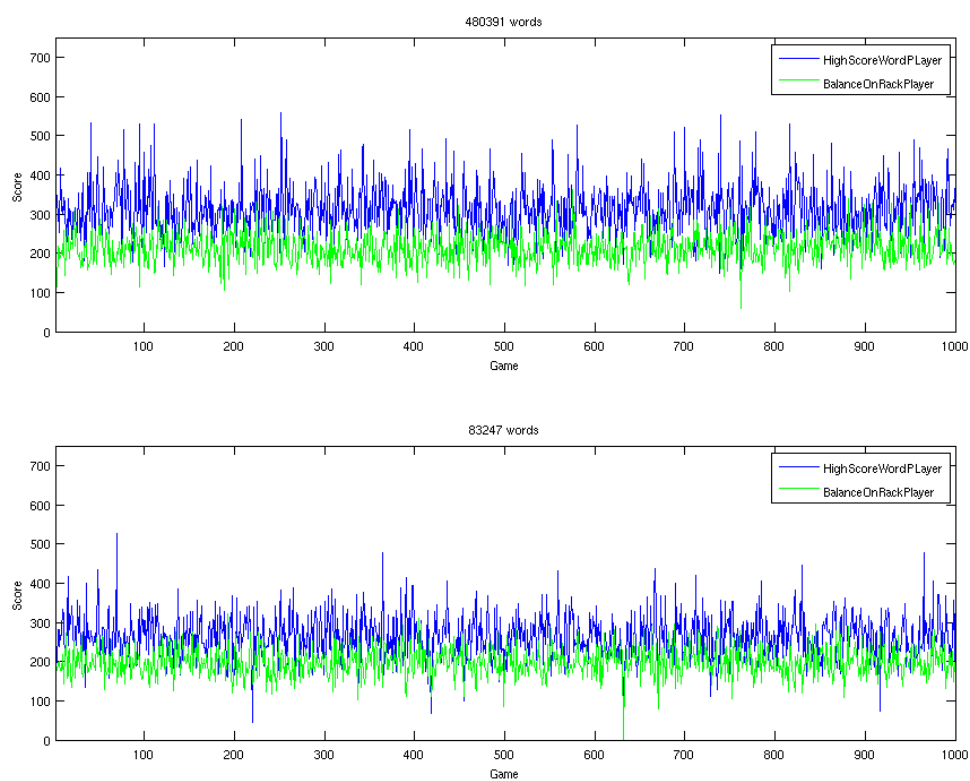


Figure B.1: Scores with vowel ratio 8/8. Scores of the two players in 1000 games with 83 247 words available.

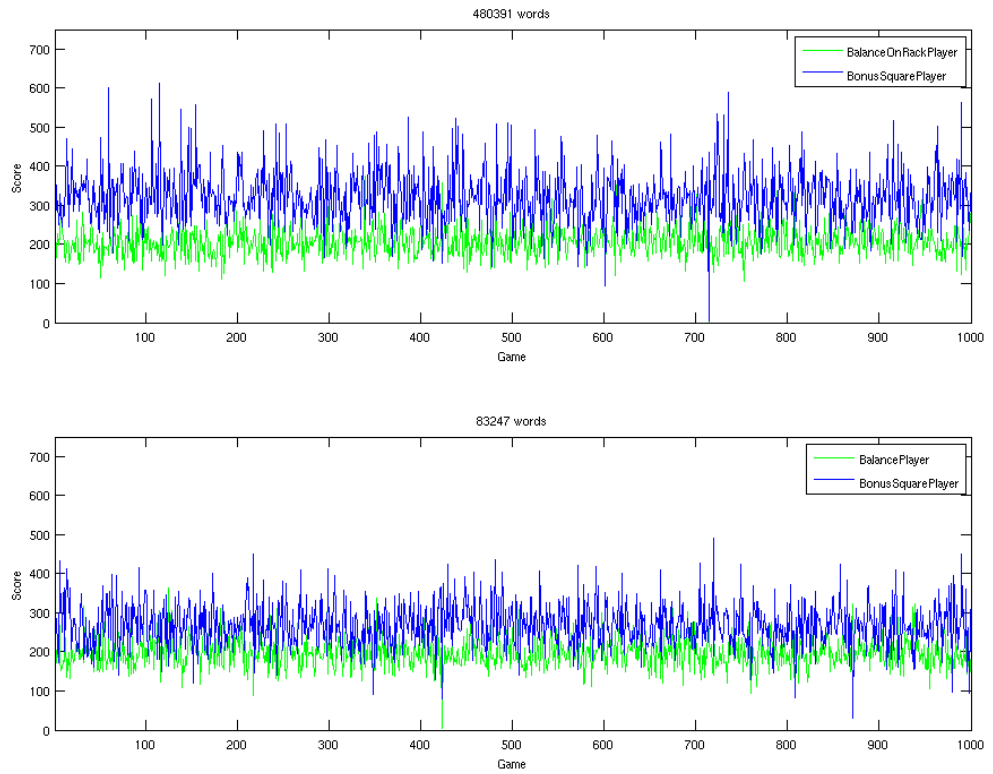


Figure B.2: Scores with vowel ratio 8/8. Scores of the two players in 1000 games with 480 391 words available.

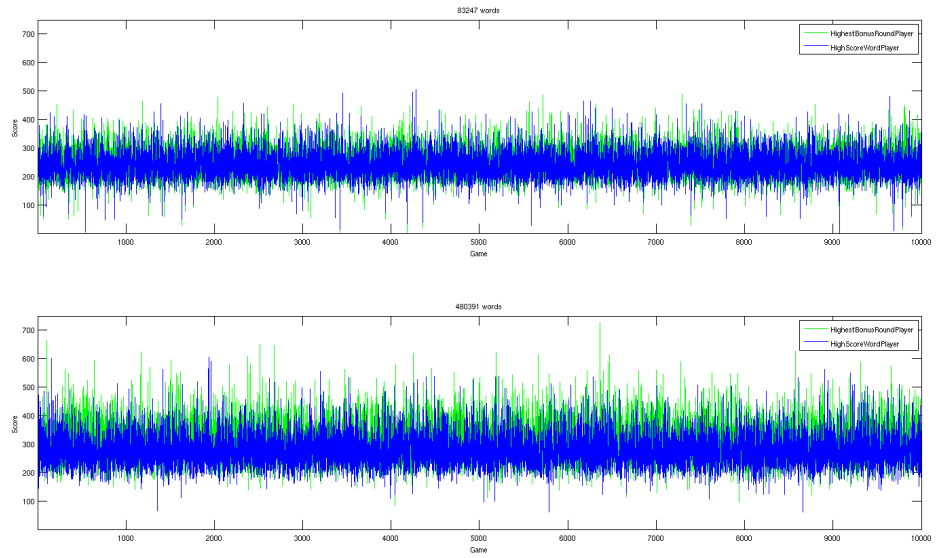


Figure B.3: End scores of the two players in 10000 games. Upper graph with base words only and lower graph with all inflection forms.

Bibliography

- [1] Sheppard, B., 2002, Towards Perfect Play of Scrabble, Maastricht.
- [2] Appel A. W., Jacobson, G. J., 1985, The World's Fastest Scrabble Program, Commun. ACM, 31(5), 572-585, May 1988.
- [3] Gordon, S. A., 1993, A Faster Scrabble Move Generation Algorithm, Software - Practice and experience, Vol. 24(2), 219-232, February 1994.
- [4] Katz-Brown, J., O'Laughlin, J., Fultz, J., Liberty, M., Buddhdev, A., 2006, Quakle - open source crossword game software, <http://people.csail.mit.edu/jasonkb/quackle/>, 2 January 2012, 12 February 2012.
- [5] Wikipedia, Scrabble, <http://en.wikipedia.org/wiki/Scrabble>, 12 February 2012.
- [6] Russell S., Norvig P. Artificial Intelligence: A Modern Approach 3rd ed. Prentice Hall. 2009
- [7] Manning C. D., Raghavan P., Schütze H., Introduction to Information Retrieval, Cambridge University Press, 2008, p. 49-65.
- [8] Swedish dictionary Den stora svenska ordlistan, <http://code.google.com/p/dsso/downloads/detail?name=dsso-1.52.txt&can=1&q=>
- [9] Swedish tournament rules by Svenska Scrabbleförbundets, <http://www.scrabbleforbundet.se/index.php?option=content&task=view&id=16&Itemid=>
- [10] World English-Language Scrabble Players' Association (WESPA) rules, <http://www.wespa.org/rules/RulesV2nov11.pdf>
- [11] North American SCRABBLE Players Association official tournament rules, <http://www.scrabbleplayers.org/wiki/images/a/af/Rules-20110605.pdf>