go

By Mustafa Jaffar

School of legends

Comp sci

Table of Contents

[1.0 Analysis 2](#_Toc75256765)

[1.1 Identification 2](#_Toc75256766)

[1.1.1 Why this is solvable by a computational approach 2](#_Toc75256767)

[1.2 Stakeholders 3](#_Toc75256768)

[1.2.1 End Users 3](#_Toc75256769)

[1.2.2 Requirements 4](#_Toc75256770)

[1.3 research 4](#_Toc75256771)

# Analysis

## Identification

Go is a board game which is said to be one of the oldest board games as it was created nearly 4000 years ago in China. It is popular in china, kora and especially in japan. Japan had 4 very competitive go schools which was set up and funded by the government during the Tokugawa, which made Go a profession in Japan. It later became globally popular in other countries after World War 2. There is also a European Go federation which holds many different competitions and championships in several countries every year.

Go is a 2-player abstract strategy board game where the opponent is trying to capture territories on the board by encapsulating the territory with their pieces, also known as stones. The two plyers get assigned a colour at the start and usually the weaker opponent gets white, at a turn you can either place a stone down or pass a stone to your opponent to ‘surrender’ your go. You win by capturing the most territories on the board, where one territory is an intersection between lines on the playing board.

I will be creating an online virtual version of Go that will allow multipole users from the same network to play the game. They will be able to choose their difficulty level and get a randomly selected person at the same difficulty level to play with them. This is will be suitable to all skill levels as I will accommodate players that have no prior knowledge of go to play as well.

### 1.1.1 Why this is solvable by a computational approach

The reason this game can be approached computationally is because Go is not a very popular game and not very many people are either good at the game or plays it, so by creating it a virtual game we can expose the whole world to it and allow people of all skill levels and backgrounds to play the game and advance their skills.

The game can also span many hours at the highest skill levels, so allowing them to save and store games virtually and easily restore them can be solved using software. Players are also rated using Kyu and Dan. Where Kyu is a set of student level ranks and Dan is a set of master level rank. People would, and most probably, end up forgetting their rank and their rating number if just done using pen and paper, however if it was stored virtually on a user-friendly interface, it would allow the players to get rid of the hassle of needing to remember their rating and allow them to compare their ratings with others across the world.

Another reason that they may not have the board and pieces available to them all the time or have enough people to play with. To overcome this having a virtual version allows you to always be able to play on demand and to know there are always going to be people wanting to play with you.

We can computationally find a cycle within the game board which would represent a capturing of stones. There are many cycle detection algorithms such as Hamiltonian cycle detection and the Floyd Warshall cycle detection algorithm, I will discuss these algorithms in more detail later. This can allow us to quickly identify if a player has successfully

captured a set of stones or not.

## Stakeholders

### 1.2.1 End Users

My end users will educational institutions wanting to teach the game and players wanting to advance their skills within the game. For this

I will interview and talk with teachers and other people within the school to see what their expectations and wants are for a virtual Go game and see if I can reasonably reach their needs while developing the game.

Players with little or no knowledge of the game will be able to select the board size of 9x9 and play a relatively simple game to start off with as they can learn how to play the game and begin to understand the rules, they will be matched with someone at the same skill level as them so they can reasonably advance within the game.

Intermediate players with be able to choose a board size that is either 13x13 or 17x17 to be able to play a more stimulating and challenging game as they will need a more advanced level of play for them to enjoy the game.

For the most skilled players they can play with board sizes 19x19 which should satisfy their needs and allow them to play a challenging enough game for them. They will most probably play for hours so the ability to save is and save the game state is a needed requirement.

Another more general stakeholder is the casual player of Go, this is where they may not have the board and pieces available to them all the time or have enough people to play with. To overcome this having a virtual version allows you to always be able to play on demand and to know there are always going to be people wanting to play with you.

|  |  |  |
| --- | --- | --- |
| The user | Interaction | Availability |
| Newbie | * Smaller sized 9x9 boards for games * Using the online feature to find other players at same level * Maybe 10x10 boards for advancement | * 2 times a week |
| Intermediate player | * Medium sized boards for games * Playing games of about an hour or more * Tries to advance by using larger boards | * weekly |
| Advanced players | * larger sized boards for games such as 17x17 or 19x19 * they will play for long period, such as 2-3 hours or more * will need the ability to pause and save a game state * maybe play smaller boards for shorter amount of time such as 30 mins | * weekly |

### 1.2.2 Requirements

The system will be written in python 3, so as a guide the user must have the necessary requirements to run python 3, they will also need enough disk space to store the game. The hardware requirements are that they will need a mouse to control the stones and where to place them, a screen so they can see visually what the current state of the system is, a keyboard to be able to press the enter key when required and enough hard disk space to store the Go game itself.

Here are the basic requirements to run python 3 from the python website itself

## Research

### 1.3.1 online-go.com

This is online-go.com version of the game. They have

Features:

* Their boards is supposed to model a wooden go board, imitating the real life original version of Go
* They have a menu section in the left hand side of the screen
* They highlight the icon or word the user is currently hovering their mouse over
* Their board puts in automatic stones put in for you when you are over the different positions
* They also have a fundamentals section and a basics section for beginners when they start the game online to they can learn how to play the game

Limitations:

### 1.3.2 Go by AI Factory Limited

Features:

* It has a AI which you can play against
* Their board is modelling a wooden go board, imitating the real life original version of Go
* It allows you to change the different factors of the game, such as the board size, Chinese/Japanese rules, the difficulty of the AI and what handicap white should have, in a setting menu
* It keeps the number of stones captured by the opposition in brackets by their username
* It also has a two player hot seat option
* It gives you hints when you get to the higher levels, such as Dan 1

Limitations:

* It doesn’t allow you to randomise your starting colour. This isn’t good as it wouldn’t allow the user to have the chance to play as the second player. This would not meet the needs of my end user as they would want to experience the game from both angles as they would be players from a range of experience so the most experienced wanting to challenge themselves while the least experienced wanting to learn how to think while being the second player. So this would not be a suitable feature for my solution.
* You cannot play online against others, only hot seat or against an AI. This would not meet the needs of my end users as being able to play others while not having anyone physically near you is the key idea behind my solution. They would not be able to play whenever they want against another human.
* They do not offer the ability to save the current game, this would not be suitable as a single game of go could last a couple hours. This wouldn’t satisfy the needs of my users might not have the time block of a couple of hours in a day to play a full game of go. So saving the game state would be a part of my solution

### 1.3.3 Online GO by Zen Android

Features:

* They have used the wooden board background for the board to try and imitate the real life version
* They have a rating system which you can use to compare yourself to other players
* Allows for an online game between other players
* It tells you the number of captures stones in the centre at the top of the screen with ‘Prisoners’ in the middle
* It has an AI that also tells you the percentage chance for you to win or lose while playing against it

## 1.4 success criteria for my solution

# Design

## 2.1 decomposition

## 2.2 algorithms being used

## 2.3 usability

## 2.4 structure

## 2.5 testing

# Development

## Prototype 1

## Prototype 2

## 3.3 Prototype 3