Approximate matching for Go board positions

Alonso GRAGERA

The University of Tokyo, JAPAN
alonso@is.s.u-tokyo.ac.jp

Abstract. Knowledge is crucial for being successful in playing Go, and this remains true even for computer programs where knowledge is used in combination with a search method such as Monte Carlo Tree Search. This paper proposes a systematic method to improve the usage of positional information databases and enhance algorithms for Go by using an approximate matching (similitude-based) to retrieve information instead of only the exact matching (Zobrish hash).

Keywords: computer go, approximate matching, influence models

1 Introduction

Game playing has been a part of the core of artificial intelligence research since it became a field of study; and since the game of Go remains as one of its few grand challenges, it has a growing interest last years. Even after having studied computer Go for more than four decades now, the ultimate goal of defeating human professional players remains unachieved.

Although many progresses have been made towards this goal in several areas of the game, programs lack of an efficient ability for extracting and analyzing knowledge from similar situations as human players do, which is being considered a major weakness. The purpose of the paper is to introduce a method to measure the similitude between board positions with applications in different areas of the game from information retrieval in opening books [1] to traffic minimization in distributed algorithms [2], along with broad pattern matching [3] and winning-percentage evaluation [4].

The rest of the paper is organized as follows: Section II presents a brief introduction to the game of Go and its rules. Section III introduces some required preliminary works on influence modeling. Section IV defines formally the concept of similarity between positions and presents an way to estimate it. Section V provides some example of its use. And at last, Section VI gives a summary of results, its applications and discussions of future works.

2 The game of Go

The game of Go (also known as *Igo*, *Weiqi* or *Baduk*) is believed to have originated in Central Asia more than 2500 years ago [5], going up to more than 4000 years ago according to some sources [6], this makes it one of the oldest known board games.

Since the old times it has been regarded as an sign of intelligence; in China, it was one of the four arts that any scholar must master *(Music, Go, Calligraphy, Painting)*, in Japan, the best player of the country was given the position of a minister *(Godokoro)* in the government. And even nowadays still remains as the last board game where humans are significantly better than computers.

2.1 Rules

Although the game itself is very difficult to master, its rules are relatively simple and comprehensible. And since some essential knowledge of the game of Go is mandatory for subsequent discussions, a brief introduction to the rules of Go is stated below.

Rule 1 – Players
The game is played by two players, called *Black* and *White*.

Rule 2 – Board
The board is a grid of horizontal and vertical lines, usually of size 19 x 19.

Rule 3 – Stones
Black uses black stones and *White* uses white stones, which are placed in the board intersections.

Rule 4 – Turns
Players take alternate turns, starting with *Black*, to place their respective stones (or pass).

Rule 5 – Captures
If a player surrounds (adjacent intersections) the opponent's stone or stones completely, he captures those stones and removes them from the board.

Rule 6 – Ko
One may not play a move which repeats a previous board position.

Rule 7 – Score
There are several alternative rule sets to score, but the most used one is the Japanese rules [7]. In this rule set the game ends after two consecutive passes, then each surrounded intersection and captured stones counts as a point, and *White* is given a 6.5 points compensation for not moving first.

3 Influence models

Influence models, also known as *influence functions* or *influence maps* are intended to be a representation of how each element of the model exert some effect to its surroundings. This idea was first introduced by Albert Zobrist[8], and it has become a very extended AI technique in modern games.

In the early computer Go research literature this method was very popular and broadly used in combination with expert knowledge, but over the years it has been replaced as new techniques were introduced, finally remaining used only as an score estimator; but nowadays even that function has been take over by the MCTS algorithms. Neither the less we will show that this methods are still useful in situations where expert knowledge is required.

In the past several authors have proposed different models, that are presented below, each one emphasizing a different aspect of the board position.

3.1 Stones influence model

The first idea that comes to mind is the trivial case where the stones themselves are used as the only relevant things of the model, without exerting any effect to their surroundings. This is obtained by just initializing the black stones to $+1$, the white stones to -1 and the empty intersection to 0.

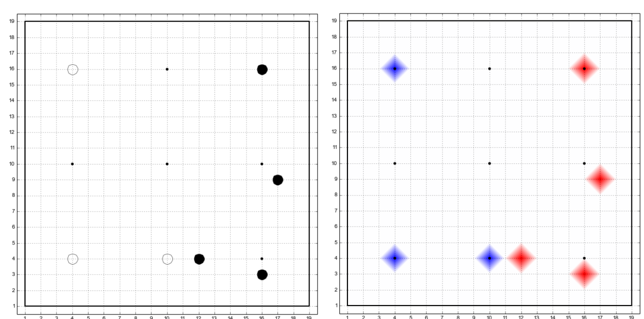


Fig. 1: Example of the stones influence model

This model results in a very simple, yet quite powerful function, that to detect trivial changes and it can be computed in an extremely efficient way.

3.2 Zobrist's influence model

Introduced by Albert L. Zobrist as part of its PhD thesis, and used in the first Go playing program [8], this model intended to capture the nature of influence in the game of Go. It studied the idea of influence as a concept emerging from the stones, with a limited range of 4 spaces and a decreasing strength, and gave them the ability to create synergies (additives when two influences of the same color collide, or canceling when the collision is between opposites, in a similar way to a magnetic field model).

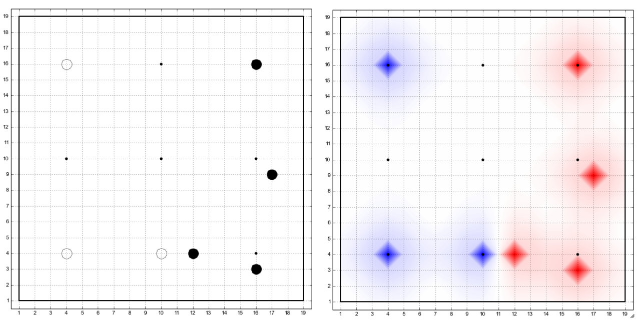


Fig. 2: Example of the Zobrist's influence model

This model is obtained by initialization the black stones to +64, the white stones to -64 and the empty intersections to 0. Then each intersection positive gives +1 to its neighbors (except for the for the ones occupied by stones), and similarly each negative intersection gives -1. The computation is completed after this transmission process is repeated exactly for times.

3.3 Ryder's influence model

This model was done by Jonathan L. Ryder also as part of its PhD thesis [9], inspired on the previous work done by Zobrist. Like Zobtist, Ryder also used an influence function to provide a numeric number to indicate the degree of tactical control each stone exert over its neighbors. His influence function is also similar to Zobrist in that black influence is positive and white influence is negative and the influence value at each intersection is the sum of the influence values propagated by its neighbors.

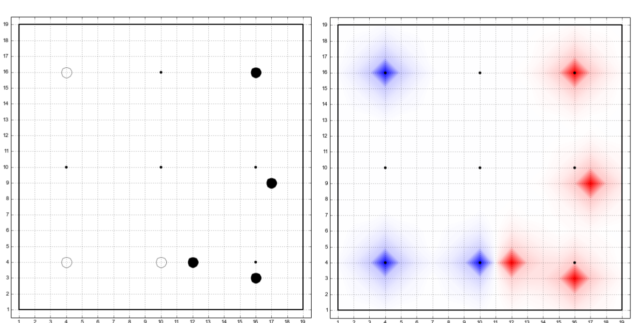


Fig. 3: Example of the Ryder's influence model

But in the case of Ryder's function, it is simpler than the Zobrist's one; each stone contributes a fixed value to its neighbors (without the need of an iterative process) according to the following pattern:

Black

			1			
	1	2	5	2	1	
	2	6	13	6	2	
1	5	13	64	13	5	1
	2	6	13	6	2	
	1	2	5	2	1	
			1			

White

			−1			
	−1	−2	−5	−2	−1	
	−2	−6	−13	−6	−2	
−1	−5	−13	−64	−13	−5	−1
	−2	−6	−13	−6	−2	
	−1	−2	−5	−2	−1	
			−1			

3.4 Spight's influence model

In 2002, the researcher William Spight introduced the idea behind his non-numerical influence model inspired in a waves coming put of the stones (therefore also known as *Spight's wavefronts analysis*) [10]. In practice, this model seems to find the equidistant boundaries between groups with opposite colors.

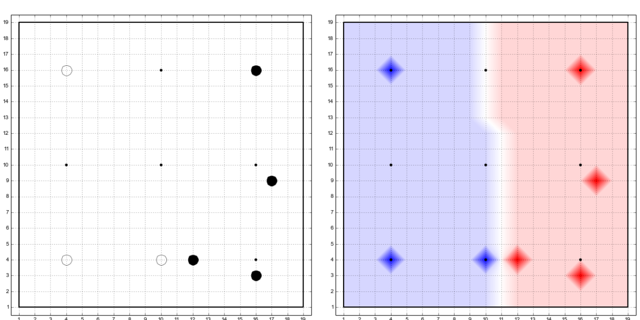


Fig. 4: Example of the Spight's wavefront analysis

3.5 Bouzy's influence model

In 2003, Bruno Bouzy published a refinement of the Zobrist influence model to accurately predict territory in the same way that human players do, therefore this model is better known as *Bouzy's territory model* [11]. It is based in the application of mathematical morphology given by the following operators:

Dilation	**External boundary**	**Closing set**
$D(A) = A + N(A)$	$ExtBound(A) = D(A) - A$	$Closing(A) = E(D(A))$

Erosion	**Internal boundary**	**Opening set**
$E(A) = A - N(UA)$	$IntBound(A) = A - E(A)$	$Opening(A) = D(E(A))$

Table 1: Basic operators

To redefine initial model by using the following Zobrist-like operators:

Zobrist Dilation	E_z = Add the #neightbours of the same color
Zobrist Erosion	D_z = Subtract the #neightbours of the opposite color or empty
Zobrist Territory	$X_z(e, d) = E_z^e \circ D_z^d$

Table 2: Zobrist-like operators

Some of the most common used ones are $X_z(13, 4)$ and $X_z(21, 5)$.

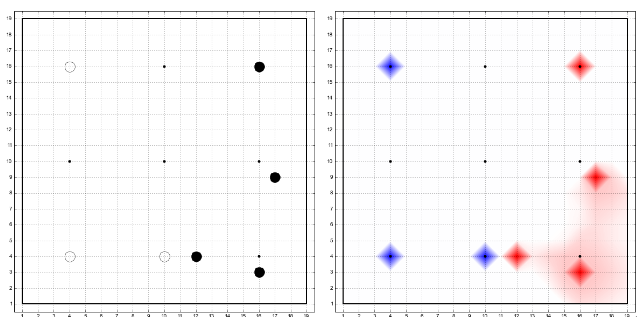


Fig. 5: Example of the Bouzy's territory model with $X_z(21, 5)$

4 Approximate matching of board positions

The necessity of extracting new knowledge from the program existing data, as mentioned in the section I, not only exists but it is also one of the current major weaknesses. In order to provide new tools to help in this task, in this paper, some methods to calculate similitude measures, between two board positions are introduced. This problem itself is extremely hard, since experts are based in their intuition do decide when two positions are similar or not.

4.1 Similitude *a posteriori*

The most reasonable approach in this case would to consider two board positions closer the more professional-level follow-up moves they have in common, after being rotated/mirrored to their respective canonical forms.

Definition 1. *Let $next(x)$ be set of follow up moves of the position $x \in \mathcal{P}$, we can define the similitude as*

$$\hat{s}_{pos}(x, y) = 1 - \frac{2\ |next(x) \cap next(y)|}{|next(x)| + |next(y)|}$$

4.2 Similitude *a priori*

But since the previous definition of similitude is impossible to compute without previous knowledge of the possible good moves for the given position, and most compelling problems on computer Go are related to the decision of the next good move, an alternative measure only dependent of the board position directly is required.

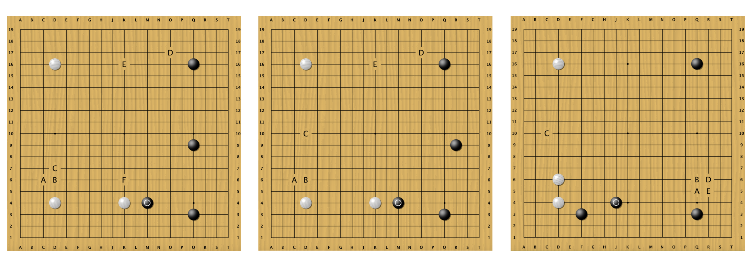
Definition 2. *Let $\hat{S}(\mathcal{P}, \mathcal{P}, \mathcal{F})$ be a family of similitude measures between two board positions $x, y \in \mathcal{P}$, under a given influence model $f \in \mathcal{F}$, defined by*

$$\hat{s}_f(x, y) = \begin{cases} 1 & if\ x = y \\ 1 - \dfrac{2}{1 + e^{\frac{1}{\alpha \sum \sum |f(x) - f(y)|}}} & otherwise \end{cases}$$

where α is a configurable model-dependent parameter.

5 Experimental evidences

On experimenting with opening books evidences that support the believe on this measurement good behavior have been gathered, and a representative example is presented in Fig.6.



(A) (B) (C)

Fig. 6: Example where $\hat{s}_{pos}(A, B) = 0.72$ and $\hat{s}_{zob}(A, B) = 0.75$, while $\hat{s}_{pos}(A, C) = 0$ and $\hat{s}_{zob}(A, C) = 0.37$, for $\alpha = 0.0013$

Though the results were very encouraging, the performance of the proposed measure is far from being completely understood, and a formal evaluation approach must be definitively made before advance any further, making the construction of a similitude-based professional games' board position datasets a top priority.

6 Conclusions, applications and future works

In summary, this paper has presented the concepts of similitude between board positions, and proposed methods to compute it. These methods described in section IV, are of a general nature and could be applied as an alternative search subroutine, where exact search is currently used [12], for example in opening book construction, message traffic reduction in massive parallelizations, pro-games clustering, winning probability evaluation, and combining results from local search of life-and-death subproblem with MCTS-based algorithms.

Nevertheless, as mentioned in the section V, the lack of formal evaluation still exist and should be properly addressed before it can be used more effectively in Go playing. These are the next directions for future research.

References

1. Mullins, J., and Drake, P.: .Using Human Knowledge to Improve Opening Strategy in Computer Go. In Proceedings of the 2010 International Conference on Artificial Intelligence, CSREA Press, 2010.
2. Yoshizoe K., Kishimoto A., Kaneko T., Yoshimoto H., and Ishikawa Y.: Scalable Distributed Monte-Carlo Tree Search. In Proceedings of the 4th Symposium on Combinatorial Search (SoCS'2011), pages 180-187, 2011.
3. Honda T., Viennot S., and Ikeda. K.: Broad Pattern Matching for the Opening Moves in Go. (Japanese) 17th Game Programming Workshop, 2012.
4. Takeuchi, S., Kaneko, T., and Yamaguchi, K.: Evaluation of Game Tree Search Methods by Game Records. IEEE Transactions on Computational Intelligence and AI in Games, 2 (4), 288 - 302, Dec. 2010.
5. Confucius: Analects, Book 17, Chapter 22, 500 BC.
6. The Editors of Encyclopædia Britannica: Go game. In Encyclopædia Britannica (online). Source: http://global.britannica.com/topic/go-game
7. Davis, J.: The Japanese Rules of Go [English translation]. Nihon Kiin and Kansai Kiin (1989). Source: http://www.cs.cmu.edu/ wjh/go/rules/Japanese.html
8. Zobrist A.: A model of visual organization for the game of Go. Ph.D. thesis, University of Wisconsin, Madison, Wisconsin, 1969.
9. Ryder J.: Heuristic analysis of large trees as generated in the game of Go. Ph.D. thesis, Stanford University, 1971.
10. Spight W.: Wave Front Analysis. Sensei's Library, 2003. Source: http://senseis.xmp.net/?WaveFrontAnalysis
11. Bouzy B.: Mathematical morphology applied to Computer Go. International Journal of Pattern Recognition and Artificial Intelligence, 17(2):257268, 2003.
12. Zobrist A.: A New Hashing Method with Application for Game Playing. Tech. Rep. 88, University of Wisconsin, Madison, Wisconsin, 1969.