THE IMPACT OF SEARCH DEPTH ON CHESS PLAYING STRENGTH

Diogo R. Ferreira[1]

Oeiras, Portugal

ABSTRACT

How deep does a chess Grandmaster think? This question has been asked many times, and yet there is hardly a definite answer. Raw depth and pure calculation are certainly not the only factors in the thinking process of a chess player, but it would be interesting to know more about the relationship between search depth and playing strength, so that the strength of a given player (which is usually expressed in terms of an Elo rating) can be said to correspond to a certain equivalent depth (of some given engine). Since the thinking depth of a human player is difficult to determine, we carry out an experiment with a chess engine running at different search depths in order to obtain an average score that can be translated into a rating difference in the Elo scale. However, knowing the rating difference is not sufficient; we need have at least one value of engine depth for which the corresponding Elo rating has been estimated, so that the Elo ratings for other values of search depth can also be determined. In order to obtain the Elo rating that corresponds to HOUDINI 1.5a 64-bit running at a fixed iteration depth of 20 plies, we carry out an analysis of the quality of play at the Candidates Tournament 2013. From these results, we show how the search depth of that particular engine correlates with the Elo ratings of human players. The paper also discusses related work on self-play experiments and the effect of diminishing returns, which becomes apparent in our experiment.

1. INTRODUCTION

In chess there is an enormous gap in playing ability between an amateur, such as a club player, and a player of master or Grandmaster strength (cf. Euwe and Meiden, 1994) . The greater ability comes from the fact that the Grandmaster *sees* and *knows* more than the amateur. He[2] *sees* more in the sense of being able to calculate more accurately and look more deeply into the game, but also *knows* more in the sense of possessing some form of structured knowledge that is built up over many years of practice and study (Ross, 2006). With such structured knowledge, Grandmasters will actually think *less* than other players when facing a given position, since they can immediately recognize the features in the position, and which candidate moves such a position affords or requires (de Groot, 2008). The story goes that when Capablanca was asked about how many moves he looked ahead, he answered "only one, but it's always the right one". In contrast, Kasparov is reported to have had a "flash of genius" when, while waiting for his opponent to play the 24th move, he saw the entire game continuation up to move 37 (Burgess, Nunn, and Emms, 2010).[3]

Currently, it is unclear how many moves a Grandmaster routinely thinks ahead, since it depends on the particular position on the board, as well as on the amount of structured knowledge that the player can resort to when facing a given position. In this work, we refrain from any consideration of what goes on in a player's mind, and instead we assume that the strength of a player is equivalent to the act of thinking up to a certain depth. For example, a grandmaster may decide to play a certain move based on a series of considerations of which the effects extend 10 or 20 moves deep (e.g., "keep both bishops on the board since it appears that lines could be opened in the future"), whereas an amateur player may be unaware of what could happen just 5 or 6 moves ahead (e.g., "exchange bishop for knight and then capture the pawn"). We postulate that the strength of a player is equivalent to that of a given

[1]IST – Technical University of Lisbon, Avenida Prof. Dr. Cavaco Silva, 2744-016 Oeiras, Portugal. Email:diogo.ferreira@ist.utl.pt

[2]For brevity, we use 'he' and 'she' whenever 'he or she' and 'his or her' are meant.

[3]The game was played between Kasparov and Topalov at Wijk aan Zee, 1999. Each move includes one ply by White and another by Black, so from move 24 to move 37 there are 26 plies. In this article we often use "move" as a synonym for "ply".

chess engine working at some fixed iteration depth d. Here, as in Heinz (2000), "fixed iteration depth" refers to imposing a depth limit on the iterative deepening of a chess program.

By equating the strength of a player to the search depth of a chess engine, the underlying assumption is that the stronger a player is, the higher will be the equivalent depth d. This does not mean that the player will follow the same iterative deepening procedure up to depth d, as the chess program will do, but only that the player is on the same level of strength as a chess engine with fixed iteration depth d. This means that, over a large number of games, the average score between player and engine would be close to 0.5 (i.e., closer to 0.5 than what would happen for $d-1$ or $d+1$). The main goal of this work is to establish a relationship between player strength and equivalent depth, so that it is possible to determine how deep the engine should search in order to reach a given playing strength. Even though the search depth of an engine does not necessarily relate to the thinking process of a human player, the relationship between search depth and playing strength will nevertheless provide a feeling for how deep a Grandmaster thinks in comparison to other less skilled players.

Since playing strength is usually measured as a rating in the Elo scale (Elo, 1978), we may formulate our task as follows: determine the relationship between a fixed iteration depth d and a corresponding Elo rating r. For this purpose, we carried out an experiment over the course of several months, where we had a strong engine (HOUDINI 1.5a 64-bit) playing against itself in a series of 24,000 games at different values of fixed iteration depth. This kind of self-play experiment has been performed extensively in the literature (e.g., Thompson, 1982; Junghanns *et al*., 1997; Heinz, 2000) , but here we use it for a different purpose. In our experiment, as in previous experiments, we use the average score between two instances of the same engine running at depths d_1 and d_2 to obtain an estimate of the rating difference $\bar{r}$ between those depths. However, this provides only a *relative* measure of strength $\bar{r} = r_1 - r_2$ for a given depth difference $\bar{d} = d_1 - d_2$. We are specifically interested in finding out the *absolute* rating r_1 (or r_2) that corresponds to a given depth d_1 (or d_2). For that purpose, we need to have at least one fixed iteration depth d for which an estimated Elo rating r is known. This can be obtained by comparing the strength of the engine at a fixed depth d with one or more players whose actual Elo ratings are known.

The paper is structured as follows. Section 2 describes the 24,000-game self-play experiment that we carried out with a chess engine running at various search depths in the range $6 \leq d_1, d_2 \leq 20$ plies. From this experiment, we collected the average score between each pair of depths, and used that to compute an estimated rating difference between those depths, as explained in Section 3. In Section 4, we describe how to estimate the engine strength at some depth d (in this case, $d = 20$) as an Elo rating. Finally, Section 5 arrives at the desired relationship between search depth and playing strength, where playing strength is measured as an absolute rating in the Elo scale. Section 6 discusses related work on self-play experiments, and Section 7 concludes the paper.

2. THE EXPERIMENT

We conducted an experiment where we had a chess engine playing against itself in a series of games at different values of a fixed iteration depth. The engine that we used in this experiment was HOUDINI 1.5a 64-bit, which, at the time when the experiment had begun, was one of the strongest engines available, only superseded by HOUDINI 2.0c. The initial idea was to play a series of $N = 100$ games for each pair of depth values (d_1, d_2) where $0 < d_1, d_2 \leq 20$, with the upper limit being set by the available computing power and the intended duration for the whole experiment. Since there might be slight differences in the results depending on whether the engine plays White or Black, we settled for $N = 200$ games with alternating colors between each pair (d_1, d_2).

Table 1 shows the number of games played. In every case, there were 100 games where the first engine played White against the second, and there were another 100 games where the second engine played White against the first. In the diagonal of Table 1 the numbers are doubled simply because $d_1 = d_2$. When $d_1 = d_2$, we have the same depth on both sides, and therefore it would not have been necessary to have $N = 200$ games. Nevertheless, we carried out the experiment in exactly the same way.

In Table 1 the minimum depth is 6 and the maximum depth is 20. A minimum depth of 6 was used because, below that, the engine would play the same moves in every game, and therefore all games would be the same. Starting with depth 6, the engine often chooses different moves in the same position, so in the end there were 23,906 different games (i.e., 99.6%) out of the 24,000 games in Table 1. The experiment was carried out on a regular desktop PC with an AMD64 dual-core processor at 2.2 GHz, and it lasted for several months, more precisely from July 23, 2012 to December 30, 2012. In order not to prolong the experiment even further, we settled on a maximum depth of 20. The experiment was carried out under the following conditions.

		Black														
		6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
White	6	200	100	100	100	100	100	100	100	100	100	100	100	100	100	100
	7	100	200	100	100	100	100	100	100	100	100	100	100	100	100	100
	8	100	100	200	100	100	100	100	100	100	100	100	100	100	100	100
	9	100	100	100	200	100	100	100	100	100	100	100	100	100	100	100
	10	100	100	100	100	200	100	100	100	100	100	100	100	100	100	100
	11	100	100	100	100	100	200	100	100	100	100	100	100	100	100	100
	12	100	100	100	100	100	100	200	100	100	100	100	100	100	100	100
	13	100	100	100	100	100	100	100	200	100	100	100	100	100	100	100
	14	100	100	100	100	100	100	100	100	200	100	100	100	100	100	100
	15	100	100	100	100	100	100	100	100	100	200	100	100	100	100	100
	16	100	100	100	100	100	100	100	100	100	100	200	100	100	100	100
	17	100	100	100	100	100	100	100	100	100	100	100	200	100	100	100
	18	100	100	100	100	100	100	100	100	100	100	100	100	200	100	100
	19	100	100	100	100	100	100	100	100	100	100	100	100	100	200	100
	20	100	100	100	100	100	100	100	100	100	100	100	100	100	100	200

Table 1: Number of games between each pair of depths

- Each game was played between two separate instances of the same engine (HOUDINI 1.5a 64-bit), each with 512 MB of hash table size.
- No opening book was used. Each engine started to think from move 1. This, however, did not prevent that regular openings were played, even a wide variety of openings appeared on the board, such as the Queen's gambit, the Ruy Lopez, the French defense, or the Scandinavian defense.
- Pondering was off, meaning that each engine was allowed to think only when it was its turn to move. This allowed each engine to take full advantage of the dual-core processor when it was its turn to think.
- At the end of each game, the two instances of the engine were shut down, and two new instances of the engine were started up for the next game. This was done in order to avoid any reuse of previous analysis, which could result in similar play across games.
- Every game ended either with checkmate or by threefold repetition of the same position (in this case the game ended in a draw). In this experiment there was no need to implement other drawing rules, such as stalemate, the 50-move rule, or lack of material for checkmate, as these never occurred.
- As usual, each decided game was counted as 1.0 for the winning side and 0.0 for the losing side, and each draw was counted as 0.5 for both sides.

Table 2 shows the average score between each pair of depths, from the point of view of the engine playing with White. (Due to space restrictions, the average scores are shown with only 2 decimal places, but the full precision is used in subsequent calculations.) In each cell, the score has been computed as an average over the number of games in the corresponding cell of Table 1.

depth	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
6	0.46	0.28	0.15	0.06	0.04	0.04	0.01	0.02	0.00	0.01	0.00	0.01	0.01	0.00	0.00
7	0.69	0.47	0.30	0.18	0.07	0.05	0.01	0.03	0.01	0.00	0.01	0.00	0.01	0.00	0.00
8	0.84	0.69	0.53	0.30	0.13	0.08	0.03	0.04	0.02	0.01	0.01	0.01	0.01	0.01	0.01
9	0.92	0.85	0.81	0.55	0.33	0.20	0.14	0.08	0.07	0.04	0.02	0.03	0.02	0.01	0.00
10	0.99	0.94	0.90	0.77	0.55	0.38	0.34	0.20	0.12	0.09	0.05	0.05	0.01	0.01	0.01
11	0.99	0.96	0.95	0.83	0.73	0.54	0.42	0.23	0.20	0.15	0.10	0.08	0.04	0.05	0.04
12	0.99	0.98	0.96	0.94	0.82	0.74	0.52	0.43	0.32	0.24	0.14	0.13	0.04	0.07	0.06
13	1.00	0.98	0.98	0.97	0.92	0.83	0.65	0.57	0.41	0.34	0.29	0.19	0.11	0.11	0.06
14	1.00	0.99	1.00	0.98	0.96	0.91	0.78	0.64	0.50	0.38	0.30	0.28	0.20	0.23	0.12
15	0.99	0.99	0.99	0.98	0.96	0.93	0.87	0.79	0.62	0.51	0.41	0.38	0.28	0.29	0.21
16	1.00	1.00	1.00	0.99	0.99	0.92	0.91	0.80	0.70	0.61	0.53	0.45	0.38	0.32	0.21
17	0.99	1.00	1.00	1.00	0.98	0.96	0.97	0.92	0.85	0.71	0.66	0.57	0.47	0.39	0.30
18	1.00	0.99	0.99	1.00	0.99	0.97	0.95	0.93	0.88	0.81	0.76	0.59	0.54	0.46	0.39
19	1.00	1.00	1.00	1.00	0.99	0.99	0.98	0.94	0.92	0.88	0.78	0.74	0.56	0.53	0.49
20	1.00	1.00	0.99	1.00	0.99	0.99	0.97	0.97	0.97	0.91	0.91	0.78	0.70	0.64	0.53

Table 2: Average scores from White's point of view

The diagonal in Table 2 shows the results for the cases where the two engines had equal depths, and therefore equal strength. Despite this fact, in most cases White seems to have a slight advantage. If we consider the 3,000 games on the diagonal of Table 1 where both White and Black had the same depth, then the average score over these games is 0.5263, which is more or less in line with the advantage that is commonly attributed to playing White.[4] For our purposes, we will ignore this slight advantage and instead we will focus on the aggregated score over all games played between each pair of depths, irrespective of colors.

Table 3 shows the average score between each pair of depths, regardless of who played Black or White. For example, if depth 6 played 100 games with White against depth 7, and another 100 games with Black against depth 7, the results from these 200 games are averaged together; in this case, the overall score of depth 6 against depth 7 is 0.29, as shown in Table 3. (Again, the average score is shown with 2 decimal places, but the full precision is used in calculations.) Therefore, Table 3 is symmetric across the diagonal, and every cell in the diagonal is 0.5 since, with the depth being equal on both sides, the results are averaged together.

depth	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
6	0.50	0.29	0.15	0.07	0.02	0.03	0.01	0.01	0.00	0.01	0.00	0.01	0.00	0.00	0.00
7	0.71	0.50	0.30	0.16	0.07	0.04	0.01	0.02	0.01	0.00	0.00	0.00	0.01	0.00	0.00
8	0.84	0.70	0.50	0.24	0.12	0.06	0.04	0.03	0.01	0.01	0.01	0.00	0.01	0.01	0.01
9	0.93	0.84	0.76	0.50	0.28	0.18	0.10	0.06	0.04	0.03	0.01	0.01	0.01	0.00	0.00
10	0.98	0.93	0.89	0.72	0.50	0.32	0.26	0.14	0.08	0.06	0.03	0.03	0.01	0.01	0.01
11	0.97	0.96	0.94	0.81	0.68	0.50	0.34	0.20	0.15	0.11	0.09	0.06	0.03	0.03	0.03
12	0.99	0.98	0.96	0.90	0.74	0.66	0.50	0.39	0.27	0.19	0.12	0.08	0.04	0.04	0.04
13	0.99	0.98	0.97	0.94	0.86	0.80	0.61	0.50	0.39	0.27	0.25	0.14	0.09	0.09	0.04
14	1.00	0.99	0.99	0.95	0.92	0.85	0.73	0.61	0.50	0.38	0.30	0.21	0.16	0.15	0.07
15	0.99	1.00	0.99	0.97	0.94	0.89	0.81	0.73	0.62	0.50	0.40	0.33	0.24	0.21	0.15
16	1.00	1.00	0.99	0.98	0.97	0.91	0.89	0.75	0.70	0.60	0.50	0.40	0.31	0.27	0.15
17	0.99	1.00	1.00	0.99	0.97	0.94	0.92	0.86	0.79	0.67	0.60	0.50	0.44	0.32	0.26
18	1.00	0.99	0.99	0.99	0.99	0.97	0.95	0.91	0.84	0.76	0.69	0.56	0.50	0.45	0.34
19	1.00	1.00	0.99	1.00	0.99	0.97	0.96	0.92	0.85	0.79	0.73	0.68	0.55	0.50	0.43
20	1.00	1.00	0.99	1.00	0.99	0.97	0.96	0.96	0.93	0.85	0.85	0.74	0.66	0.57	0.50

Table 3: Average scores over 200 games irrespective of colors

3. ANALYSIS

The results obtained in the previous section suggest that the average score bears a relationship to the difference in depth between the opponents, much like the expected score between two players bears a relationship to the difference in their Elo ratings. Consider for example depth 13, which is in the middle of Table 3, and has an equal number of both weaker and stronger opponents (depths 6–12 and depths 14–20, respectively). In the column that corresponds to depth 13, we can see that the average score goes from 0.01 (for an opponent with depth 6) to 0.96 (for an opponent with depth 20). These average scores are plotted in Figure 1.

It is apparent that these data seem to follow some sort of sigmoid function. While there are several kinds of sigmoid functions that one could consider (such as the logistic function $\frac{1}{1+e^{-x}}$, the hyperbolic tangent $\tanh(x)$, the error function $\mathrm{erf}(x)$, and other functions such as $\frac{x}{\sqrt{1+x^2}}$), here we are interested in a rather particular function: the Elo curve. If we would be able to fit the Elo curve to the data in Figure 1, then this would allow us to establish a direct relationship between rating difference and depth difference as a straightforward re-scaling factor.

3.1 The Elo Curve

The Elo rating system (Elo, 1978) is the mechanism adopted by FIDE[5] to estimate the strength of chess players. In the Elo system – as opposed to the USCF rating system (Glickman and Doan, 2013), for example – a difference of $\bar{r}$ rating points between two players corresponds to an expected score of:

$$p = \Phi\left(\frac{\bar{r}}{200\sqrt{2}}\right) \tag{1}$$

[4] See, for example, the numbers from several studies reported in: http://en.wikipedia.org/wiki/First-move_advantage_in_chess

[5] *Fédération Internationale des Échecs* (World Chess Federation): http://www.fide.com

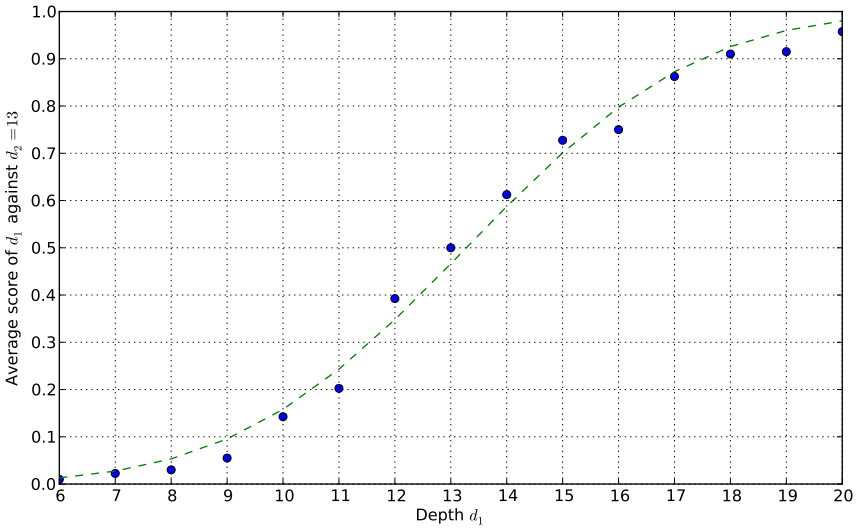

Figure 1: Average score over 200 games for each depth d_1 against depth $d_2 = 13$

where $\Phi(\cdot)$ is the cumulative distribution function (CDF) of the standard normal distribution.

Figure 2 shows a plot of the expected score p in terms of the rating difference $\bar{r}$. For example, assume that two players are rated $r_1 = 2600$ and $r_2 = 2400$. Then $\bar{r} = r_1 - r_2 = 200$ and $p = \Phi(\frac{1}{\sqrt{2}}) \simeq 0.76$ for the higher rated player. For the lower-rated player, we have $\bar{r}' = r_2 - r_1 = -200$ and $p' = \Phi(-\frac{1}{\sqrt{2}}) \simeq 0.24$. These results do not depend on the actual ratings r_1 and r_2, but only on their difference. The expected score would be the same for $r_1 = 1800$ and $r_2 = 1600$, for example.

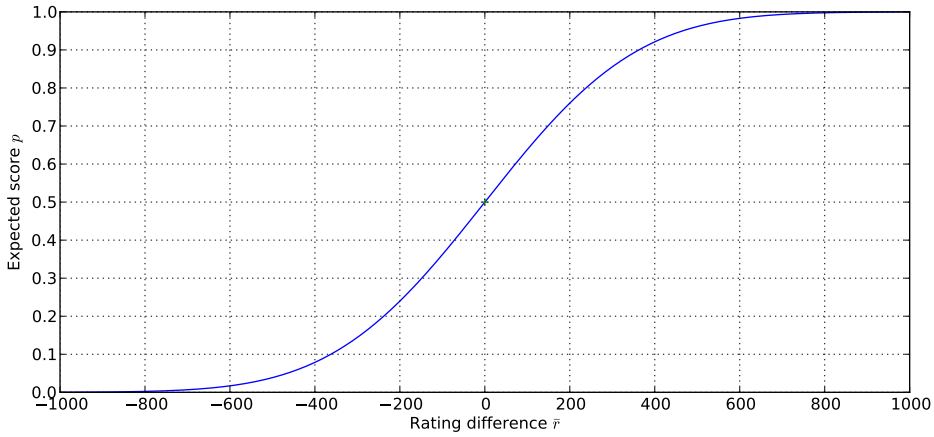

Figure 2: The Elo curve

In practice, the Elo curve can be used in both directions: either the rating difference is given and one computes the expected score, as in Eq. (1); or, alternatively, an average score over several games is given and one computes an estimate for the rating difference between the two players, as follows:

$$\bar{r} = 200\sqrt{2} \cdot \Phi^{-1}(p) \tag{2}$$

3.2 Fitting the Elo curve

The CDF of a normal distribution with some mean μ and standard deviation σ is given by:

$$\Phi\left(\frac{x-\mu}{\sigma}\right) = \frac{1}{2}\left[1 + \text{erf}\left(\frac{x-\mu}{\sigma\sqrt{2}}\right)\right] \tag{3}$$

The Elo curve is just an instance of this function with $\mu_{Elo} = 0$ and $\sigma_{Elo} = 200\sqrt{2}$. We can fit the function in Eq. (3) to the data in Figure 1 by finding the parameters μ and σ which minimize the sum of the squared errors across all data points. This approach is commonly known as *least-squares fitting* (Weisstein, n.d.). For the data in Figure 1, we find that the best fit (in the sense of least squares) is given by the parameters,

$$\mu_{(13)} = 13.2774 \qquad \sigma_{(13)} = 3.2683 \tag{4}$$

which yield the curve shown as a dashed line in Figure 1. (Note that the parameters $\mu_{(13)}$ and $\sigma_{(13)}$ are expressed in units of depth, whereas the parameters μ_{Elo} and σ_{Elo} are expressed in Elo points.) In particular, the parameter $\sigma_{(13)}$ is especially relevant since it means that the dashed curve in Figure 1 is the CDF of a normal distribution with standard deviation $\sigma_{(13)} = 3.2683$. Comparing this value with the same parameter for the Elo curve (i.e., $\sigma_{Elo} = 200\sqrt{2}$), we find that the relationship between the Elo scale and the depth scale is:

$$\eta_{(13)} = \frac{\sigma_{Elo}}{\sigma_{(13)}} \simeq 86.5 \text{ Elo points per unit of depth.} \tag{5}$$

This means that when playing against an opponent of depth 13, the engine will be 86.5 Elo points stronger for each unit increase in search depth.[6] For human players, this suggests that the effort of trying to look ahead 1 or 2 moves more deeply is well rewarded with a noticeable increase in strength, especially if one considers that in the Elo system, players are divided into classes according to their strength, and these classes are defined in intervals of 200 rating points.

It should be noted that the value in Eq. (5) is a point estimate. Given that the 95% confidence interval associated with the parameter $\sigma_{(13)} = 3.2683$ is ± 0.3399, the parameter $\eta_{(13)}$ can be shown to be in the approximate range $[78.4, 96.6]$ with 95% confidence as well.

The question now is whether the results for depth 13 can be generalized to other search depths. Carrying out the same analysis for each column in Table 3 (where we refer to each column as depth d_2) yields the results in Table 4. Here, there is a noticeable increase in the parameter $\sigma_{(d_2)}$ as d_2 increases. Consequently, there is also a noticeable decrease in the number of Elo points per unit of depth (i.e., $\eta_{(d_2)}$) as d_2 increases.

d_2	$\sigma_{(d_2)}$	$\eta_{(d_2)} = 200\sqrt{2}/\sigma_{(d_2)}$	95% conf. int.
6	1.9924	142.0	[134.0, 151.0]
7	1.9780	143.0	[135.6, 151.2]
8	1.7977	157.3	[146.1, 170.4]
9	2.0270	139.5	[125.6, 156.9]
10	2.3239	121.7	[108.5, 138.6]
11	2.5405	111.3	[98.8, 127.5]
12	2.9641	95.4	[87.4, 105.1]
13	3.2683	86.5	[78.4, 96.6]
14	3.4898	81.0	[74.1, 89.4]
15	3.8632	73.2	[68.0, 79.2]
16	3.8719	73.0	[68.1, 78.8]
17	4.0597	69.7	[65.7, 74.2]
18	4.1727	67.8	[62.6, 73.9]
19	4.5585	62.0	[57.9, 66.8]
20	4.2689	66.3	[61.3, 72.1]

Table 4: Data-fitting parameters for each value of search depth d_2

[6] This can also be referred to as "rating points per ply" as in Heinz (1998), or as "ΔElo per ply" as in Heinz (2000).

The observations above mean that when facing a strong opponent (say, with depth $d_2 = 20$), a player gains comparatively less by increasing its own depth d_1 than when playing against a weaker opponent (say, depth $d_2 = 6$). This is a symptom of *diminishing returns*, an effect that has been extensively discussed and demonstrated in several experiments in the literature (cf. Junghanns *et al*., 1997; Heinz, 2001; Steenhuisen, 2005; Guid and Bratko, 2007) . Here, the effect of diminishing returns consists in the fact that the same difference in depth becomes less of an advantage when the overall depth of both sides increases.

Figure 3 shows a plot of the fitting curve for each depth d_2. For comparison, all curves have been centered at the origin by using $d_1 - d_2$ in the horizontal axis (instead of using d_1 as in Figure 1). Each curve can be seen as giving the expected score of depth d_1 against depth d_2 as a function of $d_1 - d_2$, for some fixed depth d_2. The effect of diminishing returns is apparent by the decrease in the slope of the curve as d_2 increases.

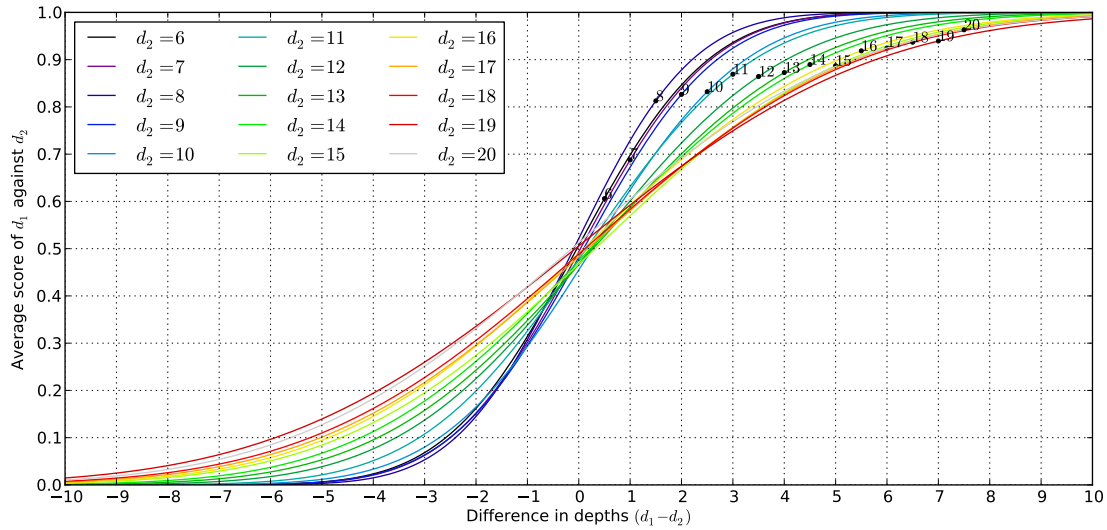

Figure 3: Data-fitting curve for each value of depth d_2

4. DETERMINING THE STRENGTH OF THE ENGINE

In the previous section we have established a direct relationship between a difference in search depth and a rating difference in the Elo scale. Basically, for two instances of HOUDINI 1.5a 64-bit with fixed iteration depths d_1 and d_2, their estimated rating difference is given by:

$$\bar{r} = \eta_{(d_2)} \times (d_1 - d_2) \tag{6}$$

Since $\bar{r} = r_1 - r_2$, one can find r_1 if r_2 is known, or r_2 if r_1 is known, but not both if only $\bar{r}$ is known. In previous work (Ferreira, 2012) we have determined that the strength of HOUDINI 1.5a 64-bit playing at depth 20 is about 2860 Elo. However, this estimate was obtained based on the analysis of a specific tournament (the London Chess Classic 2011) with 9 players with an average rating of about 2748 Elo.

In this section, we recapitulate that estimation procedure and also perform it again for a more recent tournament (the Candidates Tournament 2013). This will allow us to check the consistency of such an estimate and also to obtain a more accurate (i.e., recent) number for subsequent calculations.

4.1 Histogram of gain

In Ferreira (2012) we have described an approach to determine the strength of players based on an analysis of the gain per move. This gain is defined as the difference in position evaluations *before* and *after* the player makes a

move. The approach consists in building a histogram of gain for each player, across all the moves of that player in a single game, or, preferably, across all the moves of that player in a given tournament.

Figure 4 provides an example to illustrate how the gain per move is calculated. In this case, it is a position reached on move 25 in a game between Aronian and Gelfand in round 2 of the Candidates Tournament 2013. The position evaluation by HOUDINI 1.5a 64-bit at depth 20 is $+0.16$, which can be interpreted as a slight advantage for White. However, when Black plays 25...♖c8 the evaluation suddenly jumps to $+0.89$, i.e., a significant advantage for White. The reason for this is that White can strike with 26 ♗h6+, winning a pawn either on f7 (26...♔×h6 27 ♖×c8 ♗×c8 28 ♘×f7+ ♔g7 29 ♘×d8) or a7 (26...♔g8 27 ♖×c8 ♗×c8 28 ♘c6 ♗f6 29 ♘×a7).

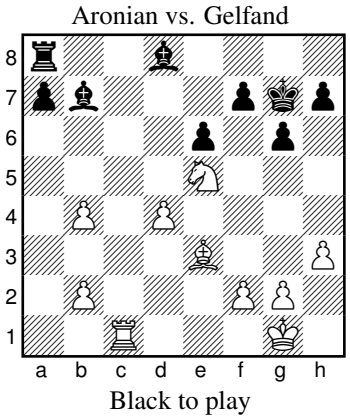

Figure 4: Position from a game in the Candidates Tournament 2013

Therefore, the gain of move 25...♖c8 is $0.16 - 0.89 = -0.73$ for Gelfand. When Aronian plays 26 ♗h6+, the position evaluations before and after this move are both $+0.89$, meaning that the gain of this move is 0.0 for Aronian (in other words, Aronian played as expected according to the engine).

By computing the gain per move of a given player across all games in the tournament, one can obtain the distribution of gain for that player. The Candidates Tournament 2013 had 14 rounds, so there were 14 games for each player. Over these 14 games, Gelfand, for example, played a total of 566 moves with a distribution of gain as shown in Figure 5. This distribution is presented as a normalized histogram, where the sum of the heights of all bars is 1.0. From here it is possible to say, for example, that 29.3% of Gelfand's moves in this tournament had gain 0.0. The scenario is similar for other players, who also have a histogram of gain with a peak at the origin.

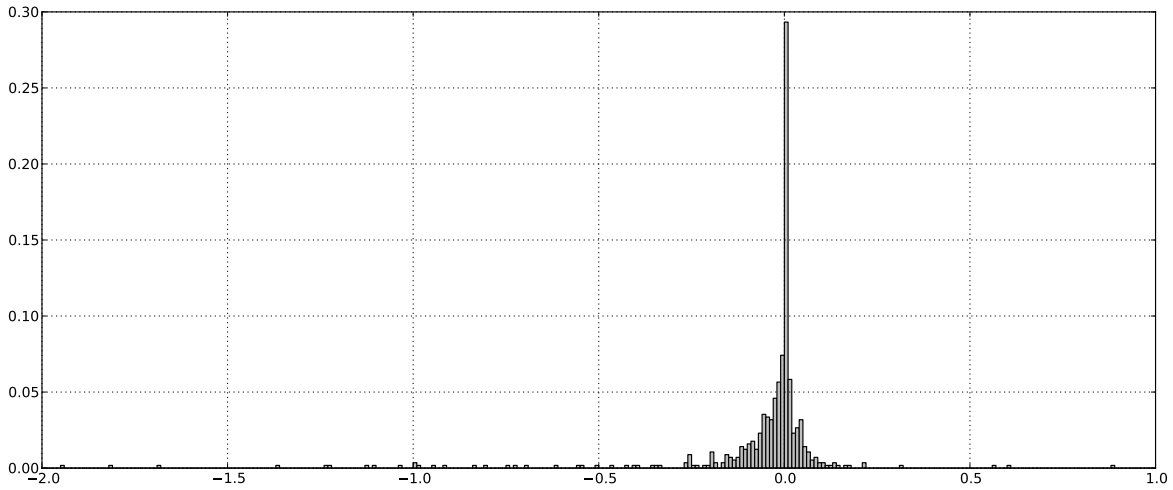

Figure 5: Normalized histogram of gain per move for Gelfand in the Candidates Tournament 2013

4.2 Using the cross-correlation

As explained in Ferreira (2012), the expected score in a game between two players can be computed as the cross-correlation of their histograms of gain. This method provides a reliable way to compare the quality of play between players, because the actual values of the position evaluations are not at all important; what is important is the distribution of gain across the histogram.

For example, if a player has a distribution of gain that has more weight on the positive side of the histogram, then this player will have an expected score of more than 0.5 against a player who has more weight on the negative side of the histogram. Even if two players have a distribution of gain with the same weight on the positive side of the histogram (or on the negative side), the player who has a distribution with more weight further to the right will be the stronger player.

The cross-correlation is calculated as a sum-product of the values in both histograms, and it provides an idea of how much the histogram of a player is to the right of the histogram of another player. This can be translated into a precise estimate for the expected score between the two players, as follows:

- Let X and Y be two discrete random variables with probability distributions $f_X[n]$ and $f_Y[n]$, respectively. Then the distribution of $X-Y$, i.e., the distribution of the difference of the two random variables, is given by the cross-correlation:

$$f_{X-Y}[n] = \sum_{m=-\infty}^{\infty} f_Y[m] \cdot f_X[n+m] \tag{7}$$

- If X and Y represent the gain per move of two players, then $f_X[n]$ and $f_Y[n]$ are their distributions of gain, as in the normalized histogram of Figure 5. The expected score between the two players can then be computed as:

$$\begin{aligned} p_{X,Y} &= 0.5 \cdot P(X=Y) + 1.0 \cdot P(X>Y) \\ &= 0.5 \cdot f_{X-Y}[0] + \sum_{n=1}^{\infty} f_{X-Y}[n] \end{aligned} \tag{8}$$

Once the expected score $p_{X,Y}$ has been determined, a straightforward application of the Elo formula – in particular, Eq. (2) – provides an estimate of the rating difference between the two players, in Elo points.

4.3 Player ranking by strength

In a given tournament, we can compute the histogram for each player by analyzing all the moves of that player in the tournament, as explained above. Then we can compute the expected score between any pair of players, in order to compare the strength of those players in terms of an estimated difference in Elo points.

However, rather than comparing pairs of players, here we are interested in obtaining a ranking of all players in the tournament according to their playing strength in that tournament. We can obtain such a ranking by comparing the strength of each player to the strength of the engine that was used to analyze their moves.

If a player would think exactly as the engine, such a player would play exactly the same moves as the engine, and therefore the gain per move of such a player would always be very close to zero. As an approximation, we consider that the distribution of gain for the engine can be represented as a histogram with a single peak at the origin.

By computing the cross-correlation between the histogram of each player and that of the engine, it is possible to obtain an estimate for the expected score between any given player and the engine. Also, this expected score $p_{i,engine}$ can be translated into an estimated rating difference $\bar{r}_{i,engine}$ in Elo points between player and engine.

The results for the Candidates Tournament 2013 are presented in Table 5, which is sorted by descending $p_{i,engine}$ (or $\bar{r}_{i,engine}$). The results suggest that Carlsen was indeed the strongest player, closely followed by Kramnik. There is an almost perfect agreement between this table and the final standings of the tournament (except for Gelfand and Grischuk, which appear in reverse order due to the tie-breaking rules).

Player	Elo	Points	$p_{i,engine}$	$\bar{r}_{i,engine}$
Carlsen	2872	8.5	0.370287	-93.6
Kramnik	2810	8.5	0.369384	-94.3
Svidler	2747	8.0	0.365234	-97.4
Aronian	2809	8.0	0.355482	-104.8
Gelfand	2740	6.5	0.353357	-106.4
Grischuk	2764	6.5	0.352007	-107.5
Ivanchuk	2757	6.0	0.338358	-117.9
Radjabov	2793	4.0	0.313953	-137.1

Table 5: Relative strength of each player in comparison with the engine

The close agreement between the results in Table 5 and the final standings in the tournament suggests that HOUDINI 1.5a 64-bit at depth 20 is sufficiently strong to be used to evaluate of the quality of play in this tournament. Also, the fact that the last column in Table 5 contains negative values suggests that the engine appears to be stronger that any of these players.[7] The next step is to determine just how strong the engine appears to be.

4.4 Estimating the engine strength

From Table 5 it is possible to estimate the strength of the engine based on the second and fifth columns (i.e., "Elo" and $\bar{r}_{i,engine}$, respectively). Since the actual Elo rating for each player is known, we can subtract $\bar{r}_{i,engine}$ from each Elo rating in order to obtain an estimate of the engine strength. Table 6 shows the result of such a calculation, where the numbers (in the last column) vary widely depending on the Elo rating of each player, and on their actual performance in this tournament.

Player	Elo	$\bar{r}_{i,engine}$	Engine
Carlsen	2872	-93.6	2965.6
Kramnik	2810	-94.3	2904.3
Svidler	2747	-97.4	2844.4
Aronian	2809	-104.8	2913.8
Gelfand	2740	-106.4	2846.4
Grischuk	2764	-107.5	2871.5
Ivanchuk	2757	-117.9	2874.9
Radjabov	2793	-137.1	2930.1
Average:	*2786.5*	*-107.4*	*2893.9*

Table 6: Estimating the strength of the engine

By taking an average of these numbers, we arrive at an estimate for the engine strength of 2893.9 Elo (with a 95% confidence interval of ± 35.4 Elo). It is interesting to note that this result is 107.4 Elo points above the average rating of 2786.5 Elo for the players in this tournament. In Ferreira (2012) we had obtained an estimate of 2860 Elo for exactly the same engine (HOUDINI 1.5a 64-bit running at depth 20); however, this estimate was based on the analysis of a different tournament, the London Chess Classic 2011, where players had an average rating of 2748.1 Elo, which is 111.9 Elo points below the estimated strength of the engine.

If we look at the difference between the estimated strength of the engine and the average rating of players (107.5 Elo points in the case of the Candidates Tournament 2013, and 111.9 Elo points in the case of the London Chess Classic 2011), we find that the difference is similar in both cases, and that the estimation is reasonably consistent. The fact that the difference is slightly smaller in the case of the Candidates Tournament 2013 can be explained by the fact that the set of players in that tournament was slightly stronger than in the London Chess Classic 2011.

[7] At this point, it should be mentioned that no such claim is being made here, this is just meant to highlight a feature that can be observed in the results. About the question of whether computers are actually stronger than humans, which is not the main topic of this work, the reader is referred to e.g. (Newborn, 2011).

4.5 Performance ratings

With an estimate of the engine strength in terms of an Elo rating, we can compute a performance rating for each player in the Candidates Tournament 2013. Basically, these performance ratings are calculated by adding $\bar{r}_{i,engine}$ to the estimated strength of the engine. Table 7 shows the results.

Player	Engine	$\bar{r}_{i,engine}$	Perf.
Carlsen	2893.9	-93.6	2800.2
Kramnik	2893.9	-94.3	2799.6
Svidler	2893.9	-97.4	2796.5
Aronian	2893.9	-104.8	2789.1
Gelfand	2893.9	-106.4	2787.5
Grischuk	2893.9	-107.5	2786.4
Ivanchuk	2893.9	-117.9	2776.0
Radjabov	2893.9	-137.1	2756.8

Table 7: Performance ratings for the Candidates Tournament 2013

The performance ratings can be interpreted as providing the perceived strength of players in this tournament, based on an analysis of their quality of play (as opposed to an analysis that takes into account the results of individual games). An interesting feature of these performance ratings is that their average is the same as the average rating of players (i.e., 2786.5 Elo).

5. REVISITING THE RELATIONSHIP BETWEEN DEPTH AND STRENGTH

Now that we have an estimate for the engine strength at depth $d=20$, we can go back to the analysis of Section 3 in order to determine an absolute rating for other values of fixed iteration depth. Basically, in Section 3 we had already obtained the scaling factor $\eta_{(d_2)}$ between the Elo scale and the depth scale. In particular, for $d_2=20$, we have $\eta_{(20)} \simeq 66.3$ Elo points per unit of depth (see Table 4).

In Section 4, we obtained an estimate for the engine strength at depth 20 of 2893.9 ± 35.4 Elo. Working backwards from $d_2=20$, and using $\eta_{(20)}$ and its confidence interval from Table 4, we can determine the estimated strength associated with any given depth d_1 in terms of an absolute Elo rating. Table 8 shows the results.

Depth d_1	d_1-20	$(d_1-20)\times\eta_{(20)}$	Strength (Elo)	95% conf. int.
20	0	0	2894	[2859, 2929]
19	-1	-66	2828	[2786, 2868]
18	-2	-133	2761	[2714, 2807]
17	-3	-199	2695	[2642, 2745]
16	-4	-265	2629	[2570, 2684]
15	-5	-331	2563	[2498, 2623]
14	-6	-398	2496	[2426, 2562]
13	-7	-464	2430	[2354, 2500]
12	-8	-530	2364	[2282, 2439]
11	-9	-596	2298	[2209, 2378]
10	-10	-663	2231	[2137, 2317]
9	-11	-729	2165	[2065, 2255]
8	-12	-795	2099	[1993, 2194]
7	-13	-861	2033	[1921, 2133]
6	-14	-928	1966	[1849, 2071]

Table 8: Estimated strength of the engine at different search depths

Table 8 is the main contribution of this work, as it shows how different values of fixed iteration depth for HOUDINI 1.5a 64-bit relate to estimated ratings in the Elo scale. These results suggest that, for example, the strength of a club player can be matched by having the engine playing at depth 6 or 7, whereas for a Grandmaster, which must have achieved an Elo rating of at least 2500 at some point, the engine depth should be above 14.

Regardless of whether a Grandmaster plays by theory, by principle, by calculating all variations, or by analogy with similar positions, the fact is that these considerations extend quite deep into the game. The results in Table 8 suggest that a Grandmaster thinks based on considerations with effects extending more than 14 plies deep, at least as seen by the particular engine being used here. If a different engine is used, it may be the case that the same strength can be matched at a higher or lower depth. In any case, Table 8 provides an idea for how deep a stronger player thinks in comparison to other less skilled players.

The results in Table 8 also depend, to some extent, on the estimated strength of the engine, i.e., on the estimated rating for depth 20 that serves as a starting point. If this value is increased or decreased by a certain amount, then the results in Table 8 will be shifted by the same amount. For example, if we use the value of 2860 Elo that was calculated based on an analysis of the London Chess Classic 2011 in Ferreira (2012), all the values in the last column of Table 8 will be shifted 34 Elo points downwards; however, the baseline of 2500 Elo will still be somewhere between depth 14 and depth 15.

Finally, it should also be mentioned that some players – namely Carlsen, Kramnik, and Aronian – were present both in the Candidates Tournament 2013 and in the London Chess Classic 2011, and that their Elo ratings were on average 21 points higher in 2013 when compared to 2011. If Elo ratings in general continue to inflate, then the results in Table 8 should be adjusted according to that inflation.

6. RELATED WORK

The first author to carry out a systematic self-play experiment was Thompson (1982) who found that his own chess engine BELLE (Condon and Thompson, 1983) with depth d scored about 80% against depth $d-1$, with d in the range $4 \leq d \leq 8$. The fact that the score seemed to be roughly constant in that range triggered an intense debate since many authors believed that there should be diminishing returns for deeper search in chess, as had been observed in other games such as checkers (Schaeffer *et al.*, 1993) and Othello (Lee and Mahajan, 1990).

Junghanns *et al.* (1997) used their own engine THE TURK in a self-play experiment between depth d and $d-1$, with d in the same range $4 \leq d \leq 8$, and were able to show a decrease in the winning percentage as d increases. To be able to observe this effect at such low depths, they limited the length of games (to 50 moves and even less) and automatically adjudicated them as a win if one side had an advantage of at least 3 pawns.

Despite the results by Junghanns *et al.* (1997), the issue of diminishing returns was far from being settled. Newborn (1985) had raised the hypothesis that the increase in strength for $d+1$ in comparison to d is related to the probability of finding a new best move at depth $d+1$ in comparison to the probability of finding a new best move at depth d. Following this hypothesis, Hyatt and Newborn (1997) carried out an experiment with CRAFTY, where they observed that the probability of finding a new best move at depth d was fairly constant (at around 0.16) in the range $9 \leq d \leq 14$. Heinz (1998) confirmed these results with DARKTHOUGHT, which seemed to point to an absence of diminishing returns, at least in that depth range.

It was only later that Heinz (2000) found evidence of diminishing returns in a more extensive self-play experiment with FRITZ 6. His experiment was conducted as a series of matches between depths d and $d-1$, for d in the range $6 \leq d \leq 12$. Each match between d and $d-1$ consisted in at least 1,050 games, and the results showed that the average score decreased from about 0.71 for depth 6 vs. 5, to about 0.62 for depth 12 vs. 11.

At this point, it is interesting to compare the results by Heinz (2000) with our own results. In Table 3 we have the average scores over 200 games between depth d and depth $d-1$ in the cells right below the main diagonal. Figure 6 shows a plot of these values together with the results by Heinz (2000). Even though the later results were obtained in a different setup, with a different engine, and with many more games between each pair of depths, the same trend of a decreasing score with an increasing d can be observed in both cases.

Even though the phenomenon of diminishing returns had been successfully demonstrated with self-play experiments, several authors kept focusing on the original hypothesis by Newborn (1985) and on follow-up experiments to the work by Hyatt and Newborn (1997) and Heinz (1998). In particular, Steenhuisen (2005) carried out a "go deep" experiment where he used CRAFTY (a newer version) up to depth 20 in order to study the probability of a new best move being found at each depth d. His results were convincing in the sense that this probability seems to decrease significantly as d increases (from about 0.40 at depth 2 to only 0.07 at depth 20).

More recently, Guid and Bratko (2007) extended the work by Steenhuisen (2005) and used three different engines

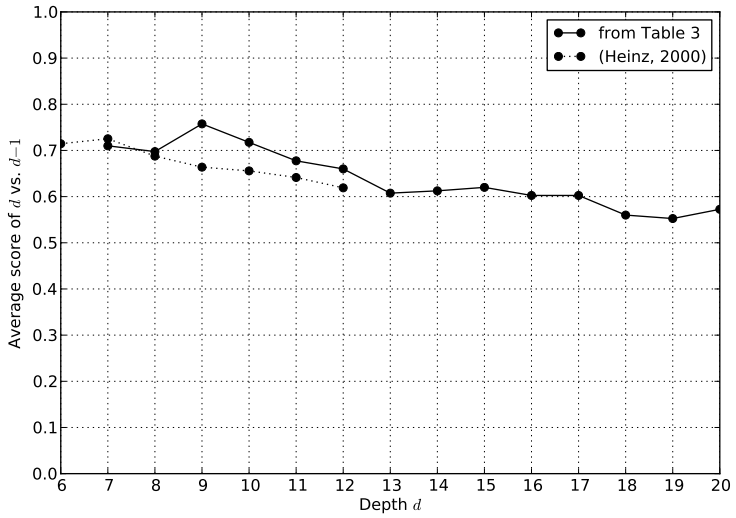

Figure 6: Average scores between depth d and $d-1$ compared with (Heinz, 2000)

as well as an extensive set of board positions to show that the probability of finding a new best move at a given depth d depends on several factors, such as the evaluation function of the engine, the phase of the game (particularly, the probability of finding a new best move seems to decrease towards the end of the game), and also the amount of material on the board. In any case, the effect of diminishing returns (here, in the form of a decreasing probability of finding a new best move) was evident as depth increased.

In our work, the effect of diminishing returns became evident in Table 4, where $\sigma_{(d_2)}$ was found to increase with depth d_2, and also in Figure 3 where the slope of the curve – and therefore the number $\eta_{(d_2)}$ of Elo points per unit depth – was found to decrease with d_2 (the depth of the opponent).

7. CONCLUSION

Although self-play experiments have been extensively used and documented in the literature, to the best of our knowledge this was the first time that the number of Elo points per additional search ply was determined based on a direct fitting of the Elo curve. However, such a fitting is not universal, as it suffers from the effect of diminishing returns. Specifically, the improvement in strength per each additional search ply seems to decrease as the depth (and therefore the strength) of the opponent increases.

Nevertheless, by focusing on one particular curve (the data fitting curve for depth $d_2 = 20$) and by estimating the strength of the engine at such a depth, we were able to find an absolute Elo rating for each depth d_1 in the range $6 \leq d_1 \leq 20$, as shown in Table 8. In our self-play experiment we found that when playing against Houdini at depth $d_2 = 20$, whose strength was estimated to be about 2894 Elo when compared to the world's top players, the engine will gain about $\sigma_{(20)} = 66.3$ Elo points per each additional search ply.

The experiment took about 5 months to complete on a single PC running 24/7 (only interrupted by a few power outages). Despite having taken so long to complete, it should be mentioned that the experimental basis for this work is still rather limited, and that the results are engine-specific. In particular, and as pointed out by an anonymous referee, different engines should be tried, and the estimate for the engine strength should be computed based on multiple tournaments with players in different Elo ranges.

The results reported here are therefore provisional, as they are only a first step towards establishing a relationship between the depth of an engine and a rating in the Elo scale. In the future, I hope it will be possible to carry out similar experiments with other engines, and to go more deeply into greater search depths. This is not only a quest for perfect play, but also for discovering where human skill stands in the immense complexity of this game.

8. ACKNOWLEDGMENTS

The author wishes to acknowledge a number of relevant and interesting issues pointed out by the anonymous referees, which have contributed to describe the approach and the results in a more precise way. The author is also grateful to the Editor for his intermediary role in facilitating the discussion which led to these improvements.

9. REFERENCES

Burgess, G., Nunn, J., and Emms, J. (2010). *The Mammoth Book of the World's Greatest Chess Games*. Running Press.

Condon, J. and Thompson, K. (1983). BELLE. *Chess Skill in Man and Machine* (ed. P. W. Frey). Springer.

de Groot, A. D. (2008). *Thought and Choice in Chess*. Amsterdam Academic Archive. Republication from *Thought and Choice in Chess* (1965, 1978). Translated and augmented by G.W. Baylor, from the original (in Dutch): *Het Denken van den Schaker*. Ph.D. thesis, University of Amsterdam.

Elo, A. (1978). *The rating of chessplayers, past and present*. Arco Pub.

Euwe, M. and Meiden, W. (1994). *Chess Master vs. Chess Amateur*. Dover. Translation of *Meester tegen Amateur* (1983), third edition.

Ferreira, D. R. (2012). Determining the Strength of Chess Players Based on Actual Play. *ICGA Journal*, Vol. 35, No. 1, pp. 3–19.

Glickman, M. E. and Doan, T. (2013). *The USCF Rating System*. United States Chess Federation.

Guid, M. and Bratko, I. (2007). Factors affecting diminishing returns for searching deeper. *ICGA Journal*, Vol. 30, No. 2, pp. 75–84.

Heinz, E. A. (1998). DarkThought Goes Deep. *ICCA Journal*, Vol. 21, No. 4, pp. 228–244.

Heinz, E. A. (2000). A New Self-Play Experiment in Computer Chess. Technical Memo 608, Massachusetts Institute of Technology, Laboratory of Computer Science.

Heinz, E. A. (2001). Self-Play, Deep Search and Diminishing Returns. *ICGA Journal*, Vol. 24, No. 2, pp. 75–79.

Hyatt, R. and Newborn, M. (1997). CRAFTY Goes Deep. *ICCA Journal*, Vol. 20, No. 2, pp. 79–86.

Junghanns, A., Schaeffer, J., Brockington, M., Björnsson, Y., and Marsland, T. (1997). Diminishing Returns for Additional Search in Chess. *Advances in Computer Chess 8* (eds. J. van den Herik and J. Uiterwijk), pp. 53–67. Universiteit Maastricht.

Lee, K.-F. and Mahajan, S. (1990). The development of a world class Othello program. *Artificial Intelligence*, Vol. 43, No. 1, pp. 21–36.

Newborn, M. (1985). A Hypothesis Concerning the Strength of Chess Programs. *ICCA Journal*, Vol. 8, No. 4, pp. 209–215.

Newborn, M. (2011). *Beyond Deep Blue: Chess in the Stratosphere*. Springer.

Ross, P. E. (2006). The Expert Mind. *Scientific American*, Vol. 295, No. 2, pp. 64–71.

Schaeffer, J., Lu, P., Szafron, D., and Lake, R. (1993). A Re-Examination of Brute-Force Search. *AAAI Technical Report FS-93-02*. AAAI Press.

Steenhuisen, J. R. (2005). New Results in Deep-Search Behaviour. *ICGA Journal*, Vol. 28, No. 4, pp. 203–213.

Thompson, K. (1982). Computer Chess Strength. *Advances in Computer Chess 3* (ed. M. R. B. Clarke), pp. 55–56. Pergamon.

Weisstein, E. W. (n.d.). Least Squares Fitting. From MathWorld – A Wolfram Web Resource. http://mathworld.wolfram.com/LeastSquaresFitting.html. Last accessed on 20/01/2013.