Finding Noble Uncertainty, A Study of Uncertainty in Games

T.H.P. van Loo
Department of Data Science and Knowledge Engineering
Maastricht University
Maastricht, The Netherlands

***Abstract*—This paper presents 2 approaches for determining uncertainty in games. The first based on the difference in player strength between AI players, and another based on game refinement theory. Both show promising results and are easily and widely applicable to many different games. Both methods improve previously used ones, by using AI player data instead of human player data to make them more flexible and widely applicable.**

***Index Terms*—Uncertainty, Games, Skill, Chance, Game Refinement, Elo, Player Ranking, Noble Uncertainty**

I. Introduction

Games have intrigued humans for thousands of years, however, being able to assess these games' elements computationally has only been enabled by more recent advancements in computational technology. One such element is the games' ”uncertainty”, or how well the course of the game can be predicted, and therefore also how certain any player can be of their victory or loss. Uncertainty is a very important element for games, Cailios proposes that the outcome of a game should be uncertain for the game to be enjoyable [1]. Similarly, Malone argues that for an activity to be challenging, there needs to be uncertainty [2]. However, too much uncertainty can be detrimental to a game, and the same is true for the wrong kind of uncertainty [3]. Therefore a balance in the amount of uncertainty in a game needs to be struck. In a simple, solved, and deterministic game like tic-tac-toe, there is practically no uncertainty to the outcome of the game when one or both players are competent at the game. Most players also do not enjoy playing tic-tac-toe as much as they do many other games. In other games, like Senet, the outcome can remain very uncertain until the very end. The uncertainty in the games can come from many different aspects such as; randomness, analytic complexity, hidden information, and many more [4].

For this research however, the focus lies not in the source of this uncertainty, but in calculating the total amount of uncertainty computationally. Deursch et al. measure and compare uncertainty by comparing games such as poker to ”50% chess”, which is defined as chess where 50% of games are determined by a coin-flip instead of through regular play [5]. Here players are given a rating in a game, and the size of the distribution of these ratings gives insight into a games' uncertainty. However, these ratings are based on real-world player data, which is a major limitation for research into games that are not played commonly or at all.

Another way to look at uncertainty is by looking at games using game refinement theory [6] [7]. The game refinement score (GR), describes the acceleration of the release of information or solved uncertainty throughout a game. It has been found that Chess, Mahjong, and Go all converged to a certain Game refinement value over time through changes in their rules [8] [9] [10]. These findings suggest a certain *noble uncertainty*. *Noble uncertainty* is defined by Yicong et al. to mean ”a harmonic balance between deterministic and stochastic aspects when playing games” [8]. Here we will use a slightly different definition and say *Noble uncertainty* is the optimal amount of uncertainty in a game, where player enjoyment is maximised. However, as with the research by Deursch et al., this research is limited to only games with lots of playing data available, because it relies on real-world player data.

A. Motivation

The current research on uncertainty is mainly focused on the 2 areas previously discussed, determining uncertainty by analysing the spread of player strength in games, and by analysing games through game refinement theory. However, for both of those fields, the main approach is to analyse playing data from human players. This is likely the most accurate, but limited mainly to games that have large amounts of data available for them, like chess, go, and poker. In this paper, we try to find a more widely applicable way of analysing uncertainty in games. The main difference being that data on how these games are played are generated by using AI players instead. This way, if more general AIs are used, many more games can be analysed. Further, the concept of *noble uncertainty* coined by Yicong et al., is so far only analysed through tracking the changes of games through history. Here we will propose to expand this by also determining the uncertainty in a larger selection of games, and comparing this to user enjoyment scores. This way the concept could be further supported.

B. Problem definition & research questions

Overall, the problem looked at in this paper is: How can uncertainty in games be measured, and is it correlated with

This thesis was prepared in partial fulfilment of the requirements for the Degree of Bachelor of Science in Data Science and Artificial Intelligence, Maastricht University. Supervisor(s): [Cameron Browne]

player enjoyment (i.e. *noble uncertainty*)? This leads us to the following research questions:

- How is uncertainty empirically measured in stochastic and deterministic games?
- Is uncertainty in games related to player enjoyment?
- At what amount of uncertainty is player enjoyment maximised?

The rest of this paper is structured as follows: first related work will be discussed. Second, the methods that were used will be given, and then how those methods are used to get the results in the experiments. Then those results will be shown and discussed. Finally, a conclusion will be given.

II. Related Work

As discussed in section I, 2 main approaches have been identified related to this paper. First is the previously mentioned paper by Deursch et al. [5]. In this paper, they tried to determine for certain games whether they depend predominantly on skill or chance. Their motivation was to help decide whether games like poker are gambling or not. To do this, they look at data from real players playing the game and then giving those player Elo scores [11], which represent the strength of those players in the game. And then the size of the distribution of those scores gives you the uncertainty of the game, measured by the standard deviation. If the standard deviation is small, then player skill has a lower effect on player performance, and therefore the game is taken to be more uncertain. To then determine if skill or chance is the dominant factor, they compare games to "50% chess", in which 50% of games are replaced by the outcome of a coin flip. Games with lower uncertainty than "50% chess" are then seen as predominantly skill-based. From this paper, a few key ideas are taken, like using a ranking system to rank players of different skill and then use the size of distribution of these ranking scores to determine Uncertainty. The benchmark game of "50% chess" is not used, as this benchmark seems rather arbitrary. Only if chess were a game of pure skill, 50% chess would be a game exactly balanced between skill and chance, but as shown in our research, chess is not pure skill, as weaker players still can win against stronger players, even if this is not very common. Within the games researched here, chess also did not achieve the highest uncertainty score, as the sheer complexity of the game leads to some uncertainty. Therefore, rather we rely on purely the standard deviation of the ranking scores to come to a total uncertainty score. Further Deursch et al. use data from real players to generate their player ratings, which while likely more accurate for computing uncertainty for human play, is more limited. The approach taken here is to instead use the same set of AI players for each game and to assign ratings to them instead. This way, one can also try to estimate uncertainty in games that do not have a large amount of human playing data available. In the cases like analysing historical games, or testing different rule sets of the same game, this could be of significant use.

Next, research has been done using game refinement theory to track changes in games through history, and the changes in GR that accompany them [8] [9] [10]. These papers conjectured that the games they studied all seemed to converge towards a $GR \in [0.07, 0.08]$ over time as their rules changed, indicating there is some optimal GR value for a game to have. The GR value is related to uncertainty, in that it encapsulates the acceleration of how information is released, or how uncertainty is solved throughout the game. The assumption there is that a game is better if the outcome remains uncertain until near the end. This is based on the principle of seesaw games [12], where the expected outcome of a game can go back and forth like a seesaw. Games being player for hundreds of year or more can be seen as a strong indicator that the game is enjoyed by its players. The studies referenced here base their evidence of *Noble uncertainty* solely on how these games evolved over time. This study aims to add to that by collecting ratings of games, and comparing that to our found GR values. In these mentioned papers on game refinement, the game refinement score is found based on real-world player data. In this paper, the idea of *noble uncertainty* is explored further, and we will attempt to find further evidence using computational analysis of the game. Instead of using real-world data, we will use AIs playing against each other to determine the GR. We will try to further support the existence of a *noble uncertainty* by comparing GR values and our Elo variance-based uncertainty scores to indicators of user enjoyment in a set of games.

Combining and comparing these approaches together with indicators of game enjoyment/interest could add new insights to the field of game refinement theory and research into uncertainty in games.

III. Methods

A. Ludii

All of the experiments in this study are run using the Ludii General Game System [13]. On its website is describes as follows:

> Ludii is a general game system designed to play, evaluate and design a wide range of games, including board games, card games, dice games, mathematical games, and so on. Download the Ludii player to explore our ever-growing database of games, test your AI search algorithms, and design your own games. Games are described as structured sets of ludemes (units of game-related information). This allows the full range of traditional strategy games from around the world to be modelled in a single playable database for the first time. Ludii is being developed as part of the ERC-funded Digital Ludeme Project.[1]

Ludii has been chosen because of its wide variety of games[2], and its efficiency [14].

B. Player Strength and Elo variance

For calculating an uncertainty value for a given game, an Elo ranking system [11] is used to rank multiple AI players. An

[1] https://ludii.games/index.php (Accessed on 05/06/2021)

[2] https://ludii.games/library.php (Accessed on 05/06/2021)

Elo ranking system gives each player a ranking score, and after each game, this is adjusted based on the difference in rating between the players and who won. Then the standard deviation of these rankings is used to indicate uncertainty. These AIs play against each other a given number of times, this is the number of trials. When the analysis is started, a tournament bracket is generated in which each player plays every other player exactly once. This bracket is then repeated until the given number of trials has been reached. The bracket order is randomised every time it is repeated, as well as the order of who goes first in each game. This is done to minimise the effect of any first player advantage. This way, each AI plays against every other AI roughly the same amount of times or the exact amount of times in cases where the number of trials is divisible by the size of the tournament bracket. The difference in performance between the players is measured by their rank. The difference in Elo ranking between 2 players corresponds directly to the probabilities of winning when those two players play against each other. Therefore, if the difference in ratings is larger (measured by their standard deviation), then the difference in strength for the players in that game is larger. The opposite is also true where if there was a smaller difference in strength between the players, then the difference in ratings would also get smaller. This can be further extended to say that in games that are fully chance-based, the difference between players' ratings will tend to zero, and in games with lower uncertainty, this standard deviation will be higher.

All players / AIs are given the same Elo rating at the start. The starting Elo rating given is not of importance because the relative difference in rating at the end is what is measured. After each played trial/game, both players involved in the trial will have their ratings updated based on who won, and the difference in rating between them and their opponents. After this, the standard deviation of their rankings is calculated according to the following formula:

$$Ratingchange = K * (obtainedscore - expectedscore)^3 \tag{1}$$

where K is the maximum rating adjustment per game, and the expected score for a player is:

$$Expectedscore = \frac{1}{1 + 10^{(rating\_opponent - rating\_player)/400}} \tag{2}$$

C. Game refinement

The GR value is calculated as follows [7]:

$$\frac{\sqrt{D}}{B} \tag{3}$$

where D is the average game length, and B is the average branching factor. These values are estimated using a Ludii function, which runs random trials to calculate its estimation. In Ludii these functions can be found in the Complexity.java class. These functions get a compute time to determine how long the tests should be run for, and then return the average game length and average branching factor after the given amount of time has elapsed.

D. Game Enjoyment

In order to find more evidence of *Noble uncertainty*, our found GR and uncertainty values are compared to how much games are enjoyed by players. For this purpose, data were collected in 2 ways. First, ratings from BoardGameGeek[4] were collected. BoardGameGeek is a website containing a lot of information on many different board games, including more obscure ones. This information includes user ratings for these games, which is what is used for this research. These ratings are given by many different users of the site, and the distribution of ratings can also be found on the game pages linked in section IV.

Second, a survey was run, in which participants were asked to play one or more of the games included in this research using the Ludii player. the included games can be found in sectionIV. Then after playing, the users were asked to rate the games with a score from 1 to 10, which is the same rating range used on BoardGameGeek. The users were asked: ”Try to play the game for at least 10 to 15 minutes, and/or until you feel comfortable with the rules and mechanics of the game. For the more complex games, you might need more playtime to be able to give a rating."

IV. Experiments

To answer the research questions, a number of games have been selected to be investigated. These games have been chosen so that there are both deterministic, and stochastic games. They were also chosen to show off a variety of different complexities, with simpler and more complex games being included, both for deterministic and stochastic games. The games, in alphabetical order, are as follows:

1) Amazons[5]
2) Blue Nile[6]
3) Breakthrough[7]
4) Chess[8]
5) Einstein Wurfelt Nicht[9]
6) Hex[10]
7) Reversi/Othello[11][12]
8) The royal game of Ur[13]
9) Senet[14]

[3] www.gautamnarula.com/rating/ (Accessed on 04/06/2021)

[4] www.boardgamegeek.com

[5] Rating from: https://boardgamegeek.com/boardgame/2125/amazons

[6] Rating from: https://boardgamegeek.com/boardgame/33046/blue-nile

[7] Rating from: https://boardgamegeek.com/boardgame/3825/breakthrough

[8] Rating from: https://boardgamegeek.com/boardgame/171/chess

[9] Rating from: https://boardgamegeek.com/boardgame/18699/einstein-wurfelt-nicht

[10] Rating from: https://boardgamegeek.com/boardgame/4112/hex

[11] In Ludii, the name Reversi is used, but the rules used are that of Othello which is copyrighted

[12] Rating from: https://boardgamegeek.com/boardgame/2389/othello

[13] Rating from: https://boardgamegeek.com/boardgame/1602/royal-game-ur

[14] Rating from: https://boardgamegeek.com/boardgame/2399/senet

10) Tic-Tac-Die
11) Tic-Tac-Toe[15]
12) Yavalath [16]

Tic-tac-die[17] is a version of tic-tac-toe where moves are determined by a dice roll instead of normal play, so it should be a game with maximum uncertainty or a game of pure chance. It was designed for use in this research here. Tic-tac-toe, being a very simple and solved deterministic game, is then used here as a game with minimum uncertainty.

For all of the games, default rule sets in Ludii were used. For Senet this was the ruleset proposed by Kendall [15], and for the Royal Game of Ur, the ruleset proposed by Finkel [16].

The goal of the experiments was to calculate uncertainty values and GR values for all of the games, and then compare these against the game ratings collected from BoardGameGeek and through the user surveys. The first step of this was to calculate the uncertainty scores as described in section III-B. Each game was run with 120 trials using 4 different AIs:

1) One Random AI
2) One UCT AI with an iteration limit of 500
3) One UCT AI with an iteration limit of 1000
4) One UCT AI with an iteration limit of 2000

All 3 UCT AIs are the default UCT AI implemented in Ludii (standard MCTS with UCB child selection and no enhancements). They have an exploration constant of 2, and uniform playouts. These AI were chosen as to represent a range of different skill levels. The random AI always plays a random legal move and is therefore the lowest skill level of the bunch. With a higher iteration limit, the UCT AIs can explore more of the game tree to make their decisions, therefore the UCT AI with an iteration limit of 2000 should perform the best in games where skill is important. For the Elo calculations, the value of K was set to 32. This value is commonly used for Elo systems for beginner chess players[18]. Further values for K have not been tested, as the initial value has worked sufficiently well.

Second, GR values were calculated with function 3 mentioned in section III-C to obtain the estimated average branching factor and game length. These were then put into the formula given in the same section to calculate our final estimated GR value.

Lastly, a test was set up to compare GR values obtained in this research, to those of previous research. The following 4 games were chosen for the comparison:

- Chess
- Chaturanga
- Shatranj
- Go

These games were chosen because they were 2 player games implemented in Ludii, with GR calculations available from previous literature. Therefore these could have GR values both from our own estimates, and other literature obtained with relative ease.

V. Results

Table I shows all the data collected in the experiments described in section IV. In the 3rd and 4th column there are also confidence intervals on the mean of the standard deviations from the Elo ratings assigned to the AIs after playing. These were calculated by running each test multiple times and then using the results from those tests as our samples. The confidence intervals differ between games because both the variance in the scores between tests differs per game, and the number of samples collected differs per game. This is because running chess trials took much longer than running trials for a simpler game like Tic-Tac-Toe. The name BoardGameGeek was shortened to BGG in column 6. Tic-Tac-Die has no rating from BoardGameGeek, as it was designed for use in Ludii for this research, and therefore not widely played and rated. Games for which very few (less than 3) samples/ratings were collected in the survey were omitted, as their mean ratings would not be very meaningful. The Elo standard deviation used as an uncertainty score in columns 2-4 represent the spread of the skill levels of the players in the experiment. Lower scores here indicate higher uncertainty.

Figure 2 shows GR estimate values plotted against game ratings obtained from BoardGameGeek. The numbers represent the games the samples come from, as listed in section IV. Figure 4 shows the same but with mean Elo rating standard deviations against BoardGameGeek ratings. Figure 3 and 5 are the same, but the ratings from BoardGameGeek are swapped out for those obtained from the survey, and therefore only games with more survey rating samples are used.

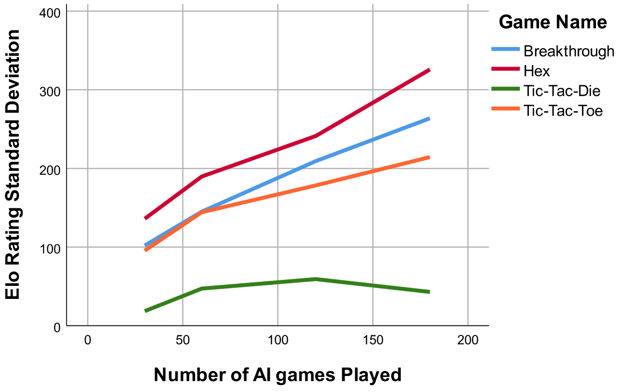

Fig. 1. Elo standard deviation and number of trials for multiple games

Then in figure 1 we show the relation between the number of games played by the AI and the calculated standard deviation in Elo rankings. This is shown for a few different games, to show it does not scale the same way for every game. Tic-Tac-Toe and Tic-Tac-Die were selected as games of pure skill and chance, and the other two for representing different uncertainty

[15] Rating from: https://boardgamegeek.com/boardgame/11901/tic-tac-toe

[16] Rating from: https://boardgamegeek.com/boardgame/33767/yavalath

[17] https://ludii.games/details.php?keyword=Tic-Tac-Die

[18] chess.com/blog/JollyPlayer/details-and-comments-regarding-the-elo-system (Accessed on 05/06/2021)

Game Name	Mean Elo STD/ Uncertainty score	95% Confidence Interval Upper Bound for Mean STD	95% Confidence Interval Upper Bound for Mean STD	GR	Mean BGG Score	Mean Survey Score
1. Amazons	229,33	219,80	238,85	0,067	7,2	
2. Blue Nile	211,85	202,70	220,99	0,168	4,9	
3. Breakthrough	209,35	202,78	215,92	0,079	6,5	4,5
4. Chess	226,32	217,19	235,46	0,011	7,1	9,3
5. Einstein Würfelt Nicht	134,79	123,08	146,49	0,068	6,6	7,5
6. Hex	259,00	253,67	264,87	0,077	6,7	
7. Reversi	219,48	211,00	227,95	0,047	6,1	6,8
8. Royal Game of Ur	193,43	187,59	199,27	0,012	5,9	7,3
9. Senet	168,71	139,27	198,16	0,011	5,8	5,9
10. Tic-Tac-Die	59,25	50,23	68,26	0,133		
11. Tic-Tac-Toe	178,31	173,07	183,55	0,305	2,7	4,3
12. Yavalath	194,60	186,29	202,91	0,385	7,1	9,4

TABLE I
TABLE WITH MAIN EXPERIMENT RESULTS

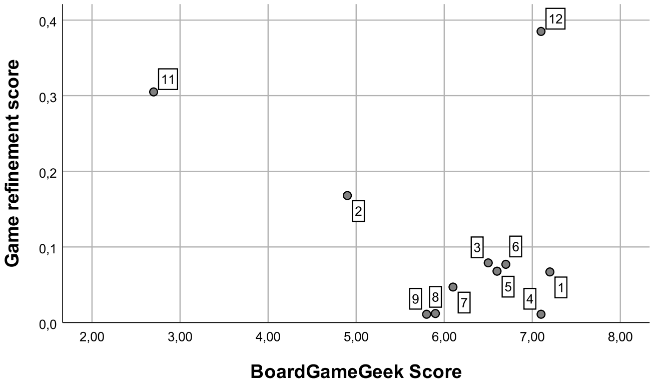

Fig. 2. GR and BoardGameGeek ratings

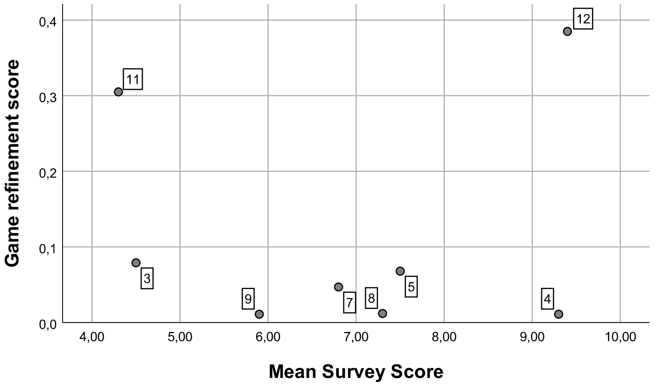

Fig. 3. GR and mean survey ratings

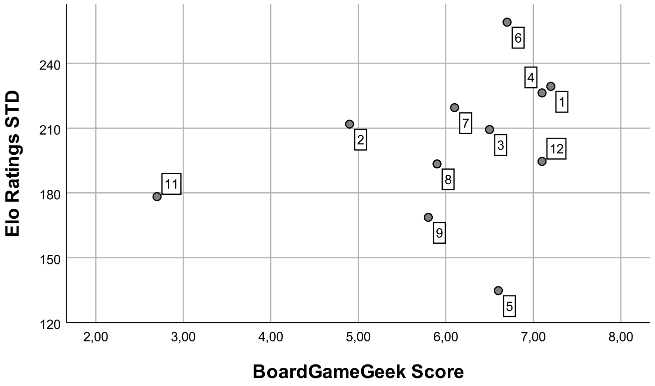

Fig. 4. Elo standard deviation and BoardGameGeek ratings

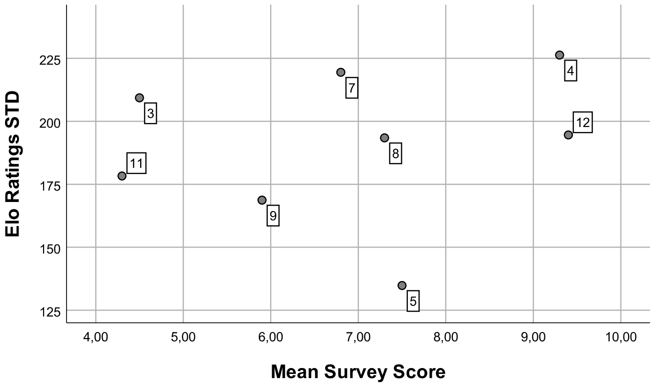

Fig. 5. Elo standard deviation and mean survey ratings

level, without taking long to run the up to 180 trials used (like Chess).

Finally, in table II we show a comparison to the GR values given by Yicong et al. in their analyses of Chess [8] and Go [9].

VI. DISCUSSION

A. Practicality

We will start by discussing the practicality of using the methods described in this paper. The methods for determining both GR values, and Uncertainty values based on skill are easy to use, applicable to virtually all of the 745 games available in Ludii at the time of writing. There are a few exceptions like backgammon. Backgammon trials are very slow to run in Ludii, largely because of the way Ludii computes the valid moves. While it is technically still possible to run the computational method for determining uncertainty for this game, it would take too long to run the 120 iterations used here in a realistic amount of time. Running the test to calculate the GR value is still possible for backgammon, but it would need to

Game Name	Computational GR estimate	GR from related work
Chess	0.011	0.052
Chaturanga	0.009	0.025
Shatranj	0.009	0.020
Go	0.019	0.0758

TABLE II
TABLE COMPARING ESTIMATED GR VALUES, TO GR VALUES OBTAINED FROM RELATED WORK

be given more compute time. Otherwise, the iteration numbers will be too low and give inconsistent results. The distance from the lower and upper bound confidence intervals to then mean is generally not very large, and generally, between the games, there is not much overlap. The most chance-based games like Tic-Tac-Die had many more samples collected than most other games, but since for game as simple as tic-tac-die, running the trials takes a significantly shorter amount of time, this was not a problem. Chess had fewer samples collected due to the higher amount of time required to run the trials, but the variance in the data was also much lower, so the confidence interval is still relatively small.

The GR experiments take far less time to run compared to the other experiment based on the Elo score standard deviations. The Elo rating standard deviation does seem to give results with a stronger correlation to the BoardGameGeek scores, but this is hard to determine conclusively based on the data collected and shown here.

B. Game Ratings

The scores/ratings from the survey are based on a relatively small amount of samples, and therefore analysis will mainly focus on the scores obtained from BoardGameGeek instead of those from the survey. We recognise that taking the survey for participants required a relatively high amount of work, especially if they wanted to give ratings for all of the games. A compromise had to be made in making the survey quick and simple to fill in, while also giving enough explanation and information so that most respondents would be able to use the Ludii player and play the games with just the information given within Ludii and the survey. For privacy reasons, no respondent data was collected, so it is hard to say how many people completed the survey, but it is clear that not all of the respondent played a majority of the games, as some games had very few or even no ratings given.

C. Game Refinement

For calculating the GR values, a novel approach was used. Using AI trial data instead of that from real players. For the most part, this seems to give GR scores which make sense relative to those of other games. One thing we can notice from table II is that the computational estimate GR values obtained from our own methods differ quite significantly to those from related work for the games in the table. One reason for this could be related to the way the GR values are estimated in our research, as we use playouts from random AIs for the estimates. The Trials of these random AIs tend to have much longer average game lengths than humans would have for the more complex games. All 4 of the games in table II are on the more complex side, therefore these results could be skewed. It is expected that this is less of a problem in games of lower complexity. This problem might be able to be partially alleviated by using more advanced AIs instead of using trials with random AI to estimate the game length and branching factor. This would likely require larger amounts of time to calculate these estimate, as to still have high enough trial numbers. We believe this could help, because the average game length differs for random AIs, as they do not necessarily choose moves that bring them closer to winning. In contrast, more advanced AIs do, such as UCT AIs. Therefore their average game length would probably be closer to that of human players. This is, however, just a hypothesis for now and needs to be tested further.

D. Elo Rating and Player Strength

We can see in figure 1 that the number of trials that the AI plays before their final rank, can have a large impact, especially the relative difference in ratings. Our hypothesis for why this happens is that all Elo ratings will converge after a certain number of trials, and the increase in Elo rating standard deviation will slow down as it approaches this point. For a game of pure chance, like Tic-Tac-Die, this point is reached almost immediately, as there is no difference in strength between the players. However, for games where this difference is large, like hex, it might take many more iterations/trials before this point is reached. A solution for this, could be to change the number of trials from a static amount like 120 chosen here, to checking while running the trials for convergence. The downside of that approach might be that for games like chess or hex, the required number of trials before convergence would be very great. This could lead to the computations taking unreasonably long, which would make the test impractical. A noteworthy result to mention, include that tic-tac-toe, even though it intuitively has very low uncertainty, does not get a very high uncertainty score. This is because the game is so simple, that all AI other than the random AI are able to play perfectly, and therefore all games without the random AI end in a tie. There is therefore no difference in strength between the UCT players, and also no significant difference in their Elo ratings. This is a limitation of the method, where games that are easily solved by the AIs might not get accurate uncertainty scores. Another result worth mentioning is that of tic-tac-die, which even though it is completely random does not get a score of 0. We can attribute this to the way the Elo system works, as with the random nature of the game, many times a player does still win. The uncertainty score will therefore trend close to 0, but usually not quite be 0.

E. Noble Uncertainty

From figure 2 we can see that there does seem to be some kind of trend in the data, where most of the highest ratings from BoardGameGeek are for games with lower GR values, with outliers. It is hard to pinpoint a specific range, in which the measured GR is optimal and leads to the highest ratings. Most of the high ratings seem to be in or around the interval of $[0.07, 0.08]$. Generally, the same could be said for figure 3, but here there is less of a visible trend in the data and an optimal interval is not really visible.

Lastly looking at figure 4 we can see that most games get uncertainty scores somewhere in/around the range of $[190, 230]$, indicating that this might be the range where *noble uncertainty* could lie for this way of computing uncertainty. However, there are still outliers, even for the relatively highly rated games, which lie outside of this range.

F. Other Factors

Of course, it cannot be ignored that there are more factors influencing how enjoyable a game is other than uncertainty. To be able to give more clear results, more games would have to be included, but this could be automated relatively easily with the code developed for this research. The methods developed here are easily applicable to a large number of games, even though for some complex games it could take a large amount of time. There was an effort made to make the code run multiple trials in parallel so that on multi-core systems the time to complete the 120 trials could be shortened significantly. This was however not finished as it was no longer feasible within the time span of this research.

G. Future Work

To expand research behind the theory of *noble uncertainty*, further research could collect data for a larger amount of games. The methods described in this paper could be used for this. Since no new methods would have to be developed, this could be done rather efficiently.

To test the accuracy of uncertainty scores and GR values given by the methods in this research, future work could compare the results given in this paper, to results from real-world data based on similar methods to the ones used. Another option would be to alter the number of iterations or run trials until Elo ratings mostly converge as described in section VI-D. Further advances in computational technology would make higher iteration limits more and more viable over time, as computational capacity increases. This could be further amplified if the methods described here would be run in parallel, so multiple trials can run at the same time.

Lastly, another thing that could be changed for any future research for this topic, is the ranking system. There are better ranking systems available than Elo. Most of these, like Microsoft TrueSkill, are patented and need a license to be used outside of Microsoft. The Elo ranking system also has a large amount of public information and documentation available, which is why it was chosen for this research.

VII. Conclusion

The main goal of this paper was to determine how uncertainty in games can be measured, and attempting to find further evidence of *noble uncertainty*, as coined by Yicong et al. [8].

For the first research question on how uncertainty could be measured, we have identified 2 main potential approaches. These were then both implemented and tested. Both the game refinement approach and the AI player strength approach yielded promising results. The Elo approach seems slightly more accurate in determining the uncertainty of games, especially considering the limitation of calculating GR values for more complex games discussed in section VI. We also found that for the Elo approach, care must be given to how the number of trials is selected, as this can significantly influence results. Further, we determined that for some games, as was done for this research, multiple iterations of the experiment needed to be run to collect consistent data. Overall, this not a problem, except possibly for those games which take much longer to run AI trials.

The second question asked whether uncertainty is related to player enjoyment of games. We can answer this question with a very cautious yes, there seems to be some correlation between player enjoyment as given by the ratings on BoardGameGeek and the uncertainty scores obtained for this research. To conclusively say whether there actually is a correlation would most likely require a larger amount of data from more games than were used for this research.

The last research question was on whether there is a range of uncertainty values where player enjoyment is maximised. Unfortunately, there is not sufficient data to actually prove or reject this. The results hint that there might be *noble uncertainty*, but for now, this is not much more than an indication. Together with the results discussed by previous literature, this is already a much stronger indication, as both do show that certain levels of uncertainty might be preferred.

Overall the goals of this research were reached, as 2 viable methods of measuring uncertainty in games were proposed. Both of these methods are applicable to most games, as they do not require any human player analysis, and are therefore very flexible. *Noble uncertainty* can not be found conclusively based on this research alone, but the methods developed here might help in further supporting its theory.

References

[1] R. Caillois, *Man, play, and games*. University of Illinois Press, 2001.

[2] T. W. Malone, "Heuristics for designing enjoyable user interfaces: Lessons from computer games," in *Proceedings of the 1982 conference on Human factors in computing systems*, 1982, pp. 63–68.

[3] C. Klimmt, T. Hartmann, and A. Frey, "Effectance and control as determinants of video game enjoyment," *Cyberpsychology & behavior*, vol. 10, no. 6, pp. 845–848, 2007.

[4] G. Costikyan, *Uncertainty in Games*, ser. Playful thinking. MIT Press, 2013.

[5] P. Duersch, M. Lambrecht, and J. Oechssler, "Measuring skill and chance in games," *European Economic Review*, vol. 127, p. 103472, 2020.

[6] S. Xiong, L. Zuo, and H. Iida, "Possible interpretations for game refinement measure," in *International Conference on Entertainment Computing*. Springer, 2017, pp. 322–334.

[7] A. P. Sutiono, A. Purwarianti, and H. Iida, "A mathematical model of game refinement," in *Intelligent Technologies for Interactive Entertainment*, D. Reidsma, I. Choi, and R. Bargar, Eds. Cham: Springer International Publishing, 2014, pp. 148–151.

[8] W. Yicong, H. P. P. Aung, M. N. A. Khalid, and H. Iida, “Evolution of games towards the discovery of noble uncertainty,” in *2019 International Conference on Advanced Information Technologies (ICAIT)*, 2019, pp. 72–77.

[9] W. Yicong, M. N. A. Khalid, and H. Iida, “Informatical analysis of go, part 1: Evolutionary changes of board size,” in *2020 IEEE Conference on Games (CoG)*, 2020, pp. 320–327.

[10] W. Yicong, C. Liu, M. N. Akmal Khalid, and H. Iida, “Informational analysis of go, part 2: Zuozi and huanqitou,” in *2020 International Conference on Advanced Information Technologies (ICAIT)*, 2020, pp. 111–116.

[11] A. E. Elo, *The rating of chessplayers, past and present*. Arco Pub., 1978.

[12] H. Iida, N. Takeshita, and J. Yoshimura, “A metric for entertainment of boardgames: its implication for evolution of chess variants,” in *Entertainment Computing*. Springer, 2003, pp. 65–72.

[13] M. Stephenson, E. Piette, D. J. Soemers, and C. Browne, “An overview of the ludii general game system,” in *2019 IEEE Conference on Games (CoG)*. IEEE, 2019, pp. 1–2.

[14] E. Piette, D. J. N. J. Soemers, M. Stephenson, C. F. Sironi, M. H. M. Winands, and C. Browne, “Ludii – the ludemic general game system,” 2020.

[15] T. Kendall, *Passing Through the Netherworld: The Meaning and Play of Senet an Ancient Egyptian Funerary Game*. Kirk Game Company, 1978.

[16] I. L. Finkel, *Ancient board games in perspective: papers from the 1990 British Museum colloquium with additional contributions*. British Museum Press, 2007.

Appendix

A. Used Game Rules

All the rules given below are taken directly from Ludii, and are added here for reproduction value, as to make sure any future research uses the same rules if comparisons are made.

1) Amazons: Played on a 10x10 board. Each player has four Amazons (chess queens), with other pieces used as arrows. Two things happen on a turn: an amazon moves like a Chess queen, but cannot cross or enter a space occupied by another amazon or arrow. Then, it shoots an arrow to any space on the board that is along the path of a queen’s move from that place. The last player able to make a move wins.

2) Blue Nile: Played on a hexagonal board with five spaces per side. Players take turns placing stones on an empty space. The stone must be adjacent to the last stone played but cannot be adjacent to any other. The last player to be able to make a legal move wins.

3) Breakthrough: Played on an 8x8 board with a double contingent of chess pawns. Pieces move forward one orthogonally or diagonally. Pieces can capture by moving diagonally. The first player to reach the opponent’s edge of the board wins. A player also can win if they capture all of the opponent’s pieces.

4) Chess: Played on an 8x8 board with pieces with specialized moves: Pawns (8): can move one space forward; Rooks (2): can move any number of spaces orthogonally; Bishops (2): can move any number of spaces diagonally; Knight (2): moves in any direction, one space orthogonally with one space forward diagonally; Queens (1): can move any number of spaces orthogonally or diagonally; Kings (1): can move one space orthogonally or diagonally. Players capture pieces by moving onto a space occupied by an opponent’s piece. Player wins when they capture the other player’s king

5) Einstein Würfelt Nicht: The game is played on a square board with a 5×5 grid. Each player has six cubes, numbered one to six. During setup, each player can arrange the cubes as he or she sees fit within the triangular area of their own color. The players take turns rolling a six-sided die and then moving the matching cube. If the matching cube is no longer on the board, the player moves a remaining cube whose number is next-higher or next-lower to the rolled number. The player starting in the top-left may move that cube one square to the right, down, or on the diagonal down and to the right; the player starting in the bottom-right may move that cube one square to the left, up, or on the diagonal up and to the left. Any cube which already lies in the target square is removed from the board. The objective of the game is for a player to either get one of their cubes to the far corner square in the grid (where their opponent started) or to remove all of their opponent’s cubes from the board.

6) Hex: Players take turns placing a piece of their colour at an empty cell, and win by connecting their board sides with a chain of their pieces. The game is played on an 11x11 board

7) Reversi: Reversi is played on an 8x8 board. Pieces are double-sided, with each side distinct in some way from the other. Each side designates ownership of that pieces to a certain player when face-up. Play begins with the players taking turns placing pieces into the central four squares until they are full. Then players may place their pieces next to an opponent’s piece, a long as a straight line can be drawn between the new piece and an existing piece belonging to that player that goes through the opponent’s piece. The opponent’s pieces between the new piece and the old piece are then flipped and now belong to the player who just played. If a player cannot make a legal move, they pass. Play continues until the board is full or neither player cannot make a legal move. The player with the most pieces on the board wins.

8) Royal Game of Ur: Each player starts play on one of the top corners of the 3x4 grid, proceeding down that row to the opposite corner, and then up the central track, which both players use, and then turning back toward the original side of the track when reaching the top of the central track in the 2x3 grid. If a player lands on an opponent’s spot, they are removed from the board and may reenter on a subsequent turn. A rosette in the center of the central track marks the spot where a player is safe from capture. Rosettes in the four corners allow a player to roll again. A player wins when they remove all seven of their pieces from the board by rolling the exact number of spaces left in the track, plus one.

9) Senet: Seven pieces per player, which begin on the board, alternating spaces from white to black along the track. Four throwing sticks, marked on one side and blank on the other, used as dice. The values of the throws are equal to the number of blank sides up, when no blank sides are up the throw = 5. Throws of 1, 4, and 5 grant the player another throw. All throws are made before moving, and a piece must move the full value of one throw at a time. Players alternate turns throwing the sticks, and the first one to throw 1 plays as white and moves the white piece in front. When a piece lands on a space occupied by the opponent's piece, the opponent's piece is sent back to the space where the piece that captured it moved from. When a player has two or more pieces in consecutive spaces, these pieces cannot be sent backward in this way. If a player cannot use a throw to move a piece forward, it must be used to move a piece backward. If a backward move makes a player's piece land on a space occupied by a piece belonging to the opponent, the opponent's piece is sent to the place where the player's move began. If a player cannot move, the turn ends. Spaces 26-30 provide special rules allowing the player to bear off. To move beyond square 26, the player must first land on it with an exact throw. From there, the player may: bear off with a throw of 5; move to square 30 with a throw of 4 and bear off on any subsequent throw; move to square 29 with a throw of 3, but it must stay there until borne off with a throw of 2; move to square 28 with a throw of 2, but it must stay there until borne off with a throw of 3. Pieces in squares 28 and 29 are never required to move backward and bearing off is not required from any space. When a player lands on squares 28-30 and an opponent's piece is already there, the opponent's piece is sent to square 27 instead of 26. When a piece is in square 27, whether by being sent there as described above or by being forced to use a throw of 1 to move into square 27, the player may either move the piece back to square 15 and lose one turn, or may leave the piece in square 27 until a 4 is thrown, bearing the piece off. A player cannot move any other piece on the board when one remains in square 27 or 15 after being sent back to it, and pieces which normally would be protected from bring sent back because they are next to each other may now be sent back. Pieces in squares 28-30 are safe as long as a piece is in square 27. The player in square 27 may decide to give up trying to throw a 4 on any turn and move this piece back to square 15 and lose their next turn. The first player to successfully bear off all their pieces wins.

10) Tic-Tac-Die: Play occurs on a 3x3 grid. One player places an X, the other places an O and players take turns placing their marks in the grid, attempting to get three in a row. The die is showing the cell index to place a piece.

11) Tic-Tac-Toe: Play occurs on a 3x3 grid. One player places an X, the other places an O and players take turns placing their marks in the grid, attempting to get three in a row of their colour.

12) Yavalath: Players alternate turns placing pieces on one of the spaces. The first player to place four in a row without first making three in a row wins.