Rating Logic Puzzle Difficulty Automatically in a Human Perspective

Hao Wang
National Chiao Tung University, Department of Computer Science
1001, Ta Hsieh Road, Hsinchu City, Taiwan
wanghau.ms89@gmail.com

Yu-Wen Wang, Chuen-Tsai Sun
National Chiao Tung University, Department of Computer Science
1001, Ta Hsieh Road, Hsinchu City, Taiwan
kid7676@hotmail.com, ctsun@cs.nctu.edu.tw

ABSTRACT
Logic puzzle games like Sudoku are getting popular for they are flexible in playing time and space and are useful in education. For puzzles, difficulty is arguably one of the most important factors in problem design. A problem too easy is boring, yet a problem too hard is frustrating. Providing problems with adequate difficulty to avoid boredom or anxiety is thus an important issue. In this paper we rate difficulty level of Sudoku problems with human oriented, general difficulty criteria so that the method can be used to evaluate problems of most logic puzzles. Only few previous Sudoku difficulty research are based on real playing data and the rating methods are limited to Sudoku or at most, constraint satisfaction problems (CSP). We found that the proposed method, despite of its simplicity and generality, can sort Sudoku problems in an order similar to average player solving time, the player perceived difficulty.

Keywords
difficulty, dynamic difficulty adjustment, Sudoku, puzzle, automatic content generation

INTRODUCTION
Puzzle games are becoming more popular and attract many players who do not play games before (Juul 2010). Many players take their time playing puzzles in traffic or spend some time playing puzzles in offices to run away from their work for a short time (FreeCell is a typical example). In addition, some core games like adventure games and RPGs now have logic puzzles interleaved within the gaming course. Considering factors of good puzzles, adequate difficulty is arguably one of the most important. Most classic puzzles offer problems in a wide range of difficulty so that it is easy to start playing and challenging to master. Koster (2005) says that a good game teaches players everything it has to offer before players stop playing. We might say that a problem too simple offers nothing to be learned while a problem too hard can make players stop playing before they learn. To measure the difficulty of a problem in puzzle games like Sudoku, FreeCell, and Sokoban (Warehouse Keeper) is thus getting important for game designers and providers. We hope to be able to sort human designed or even randomly generated problems by their difficulty so that we can offer a problem with adequate difficulty. Even more, we can provide a series of problems in an increasing order of difficulty. In this way, players have

Proceedings of DiGRA Nordic 2012 Conference: Local and Global – Games in Culture and Society.

better chance to learn and improve their skills gradually with less frustration and keep their interest to the game.

As mentioned before, logic puzzles are getting popular and new puzzles are invented every day to be played on PCs and mobile devices like smart phones. It is better to have a more general method to evaluate difficulty which is not limited to a certain game so that we can offer automatically generated problems of a new game with adequate difficulty level and do it without heavy human analysis on the game. Therefore, the goals of this paper are:

1. Rate the difficulty of a classic logic puzzle (we take Sudoku for its popularity and extensive gameplay records) with a method analogous to how human solve the problems. The method should not take specific characteristics of the chosen game so that it may have the potential to apply on other logic puzzles. The rating should be a real number to allow fully ranking and the rank by predicted difficulty should be highly similar to the rank of average time cost on solving them according to human play records.

2. By analyzing the method, learn more about the perceived difficulty in logic puzzle solving. Specifically, how do general factors like back trace and number of possible next moves contribute to difficulty?

PUZZLE DIFFICULTY RATING

Previous researches on logic puzzle difficulty ratings use mostly game-specific criteria. Simonis (2005) took the CSP point of view to analyze Sudoku problems and proposed a difficulty-rating model based on constraint propagation. The rating method was validated with designer-assigned difficulty level for each problem. Pelánek (2011) found that the number of high level strategies required to solve a problem without brute-force is a good measure of Sudoku difficulty. The validation is made with real data and the number of strategies required is highly correlated to average solving time in online gameplay records. The difficulty rating method is potentially extendible to CSP-based puzzles. Sokoban is mostly studied with agent-based search methods and the difficulty is also rated through game-specific criteria (Junghanns and Schaeffer 2001; Jarusek and Pelanek 2010).

There are some researches use computational complexity rather than Sudoku specific criteria. For example, Mantere et al. (2007) use the computational cost of genetic algorithm (GA) on solving a problem to measure Sudoku problem difficulty. Xu et al. (2009) define entropy of a Sudoku problem and measure difficulty according to it. However, the complexity models are not directly related to human play experiences thus cannot show why a problem is hard in human perspective. Lee et al. (2008) studied the human Sudoku problem solving process, providing a psychological view of human perceived difficulty.

In this paper we focus on logic puzzles, which are like combinatorial games except that logic puzzles are single player games. They have the following properties (Browne et al. 2010):

- Finite: There are final states in which problems are solved.
- Discrete: They are turn-based, with no real-time actions.
- Deterministic: There is no chance involved in the game.

- Perfect information: There is no hidden information in the game.

WHAT MAKES A LOGIC PUZZLE DIFFICULT FOR HUMAN

In general, difficulty of puzzles can arise in several aspects. Clarity is an important factor of difficulty and can be largely classified as visual and logic clarity. An example of visual clarity is how obvious the differences are of two almost identical pictures when players try to find all the different parts. It measures how hard for players to find necessary information. The logic clarity, on the other hand, measures how hard for players to figure out an efficient method to solve the problem. A difficult Sudoku problem requires the knowledge of multiple strategies and has lower logic clarity than simpler one.

Intuitiveness also affects difficulty. Imagine that you are solving a maze. It will be more difficult if you have to turn away from the direction of the goal many times in the course to reach the goal. Since players do not always solve problems systematically, especially when the players' perceived logical clarity is low, players rely highly on their intuition.

Another obvious one is the size. A big maze is in general more difficult to go through. However, a maze will also be more difficult if the number of crossroads is larger. So we see another important factor about size: the number of correct choices to be made. The size of required memory also affects difficulty quite much. The number of short term memory chunk of people is very limited (less than 7). Players need to use external memory like pen and paper if there are too much to remember and solving the problem becomes cumbersome.

THE GENERAL DIFFICULTY CRITERIA ON SUDOKU

Our idea about general difficulty criteria that can better reflect human perceived difficulty and are applicable for most logic puzzles is as following. The possible states of a single player logic puzzle can be presented as a search tree. The root node is the start state (the problem itself); the leaf nodes consist of solution states; child nodes of each node consist of all states that can be reached by a legal move from the current state. However, the sizes of the trees are typically too large to do a full analysis. Here we propose a method based on features of the solving path, the part of search tree visited in the solving process. In our experiment, the solver uses only naked single technique and hidden single technique so that the solving process can fit beginners better. We do this because we want the method to be applicable to new invented games and all players are beginners to these games. When there is no certain number to be filled, the solver tries uncertain number with minimal possible choices. Specifically, we use the following features on the solving path considering both computational complexity and possible psychological effects:

1. The number of basic calculations. It measures how many units of calculations are required to solve a problem by the solver. In Sudoku, this unit is to do a naked or hidden single check for a cell. Intuitively, the larger the number is, the harder the problem might be.

2. The number of times the solver guesses. For those problems which are not "simple Sudoku", the solver has to guess. The solver guesses and fills a number in the cell with minimal possible choices and continues. When players have to guess, it creates a feel of uncertainty and can affect their feeling about difficulty and even their speed of solving problems. For example, after having to guess for several times, a player might feel that it is too cumbersome to solve this problem and lose motivation.

3. The number of back trace. We think that this would be a crucial factor on difficulty since it is quite difficult for human to do back traces. Our memory is very limited and using pen and paper is both awkward and time consuming. Although players rarely do back traces in solving Sudoku, we think that the number of back traces done by a solver can be used to measure difficulty because it can reflect how hard it would be to solve a problem "by violence" without advanced skill. Since our goal is to make a method for measuring difficulty for all logic puzzles, it is adequate because for a newly invented game, all players can considered to have no advanced skill.

4. Number of certain branches. It is the number of next moves which can be immediately known to be right or wrong. In Sudoku, the former is the number of cells that can be filled with the two basic techniques in current state. It is an averaged number since every time a state of game is considered (a node is visited), we get a number. The lager the number, the easier the player can find a correct step so that the actual solving time is shorter and players feel the problem to be easier. On the other hand, a large number of certain wrong moves can create a feeling of certainty and reduce the complexity on analyzing the state by that a large portion of a branch is safely pruned.

5. Number of uncertain branches. It is the number of next moves which cannot be immediately known to be right or wrong. In our Sudoku experiment, it is the number of cells that cannot be filled with the two basic techniques in current state. It is also an averaged number. Intuitively, in contrast with the number of certain branches, a large number of uncertain branches creates feeling of uncertainty and increases complexity.

EVOLUTION OF DIFFICULTY MEASUREMENT FUNCTION

The difficulty measurement function we propose is in the form of second order polynomial for simplicity so that we can easily observe how the function is composed. Rather than optimizing result on Sudoku with sophisticate method, our intention is to find simple relationship between variables (the general difficulty criteria) and human perceived difficulty for better extendibility to most logic puzzles. We use standard GA to evolve the constants and variable combinations in each term. A good entity is a polynomial that comes out with a larger number when the values of variables are given by the feature of a harder problem and vice versa. Then we can rank problems according to the number given by the difficulty measurement function.

EXPERIMENT

In the experiment, we take the general criteria described before as the variables in a second order polynomial and use GA to find a polynomial that can sort problems similar to the order of solving time in online play data. The play data records are retrieved from a Sudoku website: http://oddest.nc.hcc.edu.tw/. The variables are:

X_1: The number of basic calculations done in the solving process.

X_2: The number of times the solver guesses made in the solving process.

X_3: The number of back traces made in the solving process.

X_4: The total number of certain branches in the visited part of search tree.

X_5: The total number of uncertain branches in the visited part of search tree.

$X_6 : \frac{X_4}{X_5}$

All variables are normalized so that the values range from 1 to 10 except X_6. We let the values be larger than 1 to make sure that a multiplication of variables always leads to larger values and division leads to smaller values. The constant in each term ranges from -1 to 1. So we may argue that a certain variable has negative contribution to difficulty if the constant is a negative number. The fitness function is the ranking similarity of order by average solving time and order by the entity; this is measured by Spearman's rho (rank correlation coefficient). We use a population size of 500, with 2-point crossover as the crossover method.

The best entity found in the experiment achieves a 0.8 Spearman's rho in GA. The same entity achieves 0.69 in the validation dataset, which constitute of 150 problems with average solving time records. Although the correlation is not as strong as researchers have achieved in previous work (Pelánek 2011) because we do not use Sudoku specific criteria, our method has higher flexibility to be applied to other, even newly invented logic puzzle games.

The entity we found in GA constitutes tens of terms. We reduce the number of terms by removing least significant terms those have less effect on ranking in order to have some insight about which terms contribute to difficulty primarily. The five most contributive terms in the best entity found in the experiment are (in order of significance):

$0.8 \times {X_5}^2$; $-0.52 \times X_4 \times X_6$; $0.05 \times X_2 \times X_3$; $0.34 \times X_1$; $0.64 \times X_1 \times X_5$

The results shed some light on how the general criteria contribute to human perceived difficulty. First of all, we can see that the number of certain and uncertain branches is crucial in human perceived difficulty since they appear in the two most influential terms. ${X_5}^2$ with a positive constant implies that the number of uncertain branches has a positive effect on difficulty. $X_4 \times X_6$, which equals ${X_4}^2/X_5$, is with a negative constant implies that the number of certain branches has a negative effect on difficulty as expected. Second, the $X_2 \times X_3$ term is also with a positive and small constant, which may imply that difficulty is sensitive with number of guessing and number of back trace. Finally, we see that X_1 also has a positive effect on difficulty, but is less influential. So the number of calculations required may not be crucial in solving time compared to other criteria.

DISCUSSION

Trained Sudoku players avoid guessing and thus not often take back traces (these are brute-force play). However, our method can still approach the difficulty they perceive. We think that it is because the general difficulty criteria can also reflect the number of high level techniques required without brute-force play, though the relationship may not be strictly defined.

Although human solve problems with different methods, and thus go through a different path in a search tree, the results show that the proposed difficulty measurement can still evaluate difficulty to some extent. A further improvement might be using several solvers to create multiple solving paths to evaluate average perceived difficulty of a problem better.

Besides difficulty, the structure of the search tree of a problem also influences the fun to play. Althöfer (2003) proposed some criteria for automatic evaluation of interestingness of two player games, and most of them are concerning the feature of search tree played. Browne et al. (2010) further test tens of criteria and successfully invent two player combinatorial games with interestingness measurement automatically. Our results about understanding the general criteria suggest that the number of possible next moves should be adequate. Consecutive uncertain next moves should be limited or the problem loses clarity, which is important in puzzle design (Abbott 1975).

CONCLUSION

In this paper we proposed a method of measuring logic puzzle problem difficulty based on the solving path taken by a beginner level solver. By taking the features like number of back traces and branches as variables in 2nd-order polynomials, we can rank problems similar to their human-perceived difficulty, estimated by average solving time in on-line play records. Also, by observing the characteristics of the difficulty measurement functions found by GA, we can have some idea about how back traces, required guesses, and number of different kinds of branches contribute to overall difficulty. The advantage of the method is that it is extendible for no Sudoku-specific difficulty criteria are used. It is especially suitable for estimating the difficulty of automatic, on-line generated problems of new inventions of logic puzzles, for which game specific difficulty criteria is unknown and human evaluation is impossible. Although the optimal parameters of difficulty measurement functions differ from one game to another, our findings about how the general criteria contribute to difficulty can serve game designers a good start on parameter tuning. Future work would be to study the players when problems are given in the order of modest increasing difficulty to see if it creates the desirable experiences. Another future work might be to further expand the difficulty criteria for games beyond logic puzzles. For example, take visual search difficulty into consideration for games like Bejeweled.

BIBLIOGRAPHY

Althöfer, I. (2003). Computer-aided game inventing. Technical report. Available at http://www.minet.unijena.de.preprints/althoefer_03/CAGI.pdf (accessed Feb. 2012).

Abbott, R. (1975). Under the strategy tree. Games and Puzzles vol. 36, pp. 4-5. Available at http://www.logicmazes.com/games/tree.html (accessed Feb. 2012).

Browne, C. and Maire, F. (2010). Evolutionary game design. IEEE Transactions on Computational Intelligence an AI in Games vol.2, no. 1, pp. 1–16.

Chase, W. G. and Simon, H. A. (1973). Perception in chess. Cognitive Psychology, vol. 4, pp. 55-81.

Csikszentmihalyi, M. (1991). Flow: the Psychology of Optimal Experience. Harper Perennial, New York.

Gobet, F. and Simon, H. A. (1996). Templates in chess memory: A mechanism for recalling several boards. Cognitive Psychology vol. 31, 1-40.

Jarušek, P. and Pelánek, R. (2010). Difficulty rating of sokoban puzzle. In Proc. of the Fifth Starting AI Researchers' Symposium (STAIRS 2010). IOS Press.

Juul, J. (2010). A Casual Revolution. The MIT Press, Massachusetts.

Junghanns, A. and Schaeffer, J. (2001). Sokoban: Enhancing Single-Agent Search Using Domain Knowledge. Artificial Intelligence, 129(1–2), 219–251.

Koster, R. (2005). A theory of fun for game design. Paraglyph Press, Scottsdale AZ.

Lee, N. Y. L., Goodwin, G. P., & Johnson-Laird, P. N. (2008). The psychological problem of Sudoku. Thinking & Reasoning, 14, 342–364.

Lewis, R. (2007). Metaheuristics can solve sudoku puzzles. Journal of Heuristics vol. 13, pp. 387–401.
Mantere, T. and Koljonen, J. (2007). Solving, rating and generating Sudoku puzzles with GA. In IEEE Congress on Evolutionary Computation, 1382-1389.
Pelánek, R. (2011). Difficulty rating of sudoku puzzles by a computational model. In Proceedings of Florida Artificial Intelligence Research Society Conference.
Simonis, H. (2005). Sudoku as a constraint problem. In Proc. 4th Int. Workshop on Modelling and Reformulating Constraint Satisfaction Problems, 13–27.
Xu, C. and Xu, W. (2009). The model and algorithm to estimate the difficulty levels of Sudoku puzzles. Journal of Mathematics Research vol. 1, no. 2, pp.43-46.
Yato, T. and Seta, T. (2002). Complexity and completeness of finding another solution and its application to puzzles. In Proceedings of the National Meeting of the Information Processing Society of Japan (IPSJ).