Effectiveness of Novel GA with Advanced Twin Operator for Solving the Number Puzzles

Anagha P. Khedkar, Dr. Shaila Subbaraman

Abstract— **It is well known that Genetic Algorithm (GA) is widely used for solving the games and number puzzles especially Sudoku puzzles and magic square. This work attempts to solve the interesting but difficult number puzzles that need exponential complexity to optimize with the conventional methods. This work focuses on the effectiveness of advanced twin operator of the novel GA for solving the number puzzles efficiently. In this work, simple GA and the novel GA with advanced twin operator are designed and applied to solve the famous number puzzles viz. a 10-digit self-describing number and last digits of n^2. The results of this implementation show the effectiveness of novel GA to generate the unique solution in terms of convergence time. Besides this, the novel GA with advanced twin operator yields better performance in terms of reduced convergence time than that of the conventional simple GA. Thus the advanced twin operator designed for the novel GA highlights its effectiveness on the overall performance of GA for this puzzle solving application.**

Index Terms— **Advanced twin operator, Genetic algorithm, Puzzle, Performance.**

I. Introduction

GAs, the heuristic computational procedures based on the principles of natural genetics, are known to be widely applicable for solving complex search and optimization problems in almost all the facets of real world. This work highlights the effectiveness of GA in solving the interesting number puzzles. It has been observed that the basic operational flow of GA has strong natural power to find out the optimal solution for various puzzles and games [1]-[9]. Since last three decades GAs are used by practitioners to solve the variety of games, logic puzzles and number puzzles especially sudoku. Many researchers have attempted to create the sudoku puzzles of different difficulty levels using GA [1], [2], [3]. Some researchers found that the efficiency of GA to solve these puzzles degrades with the increased difficulty levels of puzzles and sometimes the problem specific knowledge is required to solve the puzzles [4].

Many researchers have solved the interesting and well known magic square problems using GA [5], [6]. Magic square problem is a rich combinatorial optimization problem having n^2! ways and used as a benchmark to compare GA and similar optimization methods viz. deterministic and nondeterministic [7], [8]. In 2007, Harmony search algorithm, a type of evolutionary algorithm was applied to find out solution for sudoku puzzle [10]. The researchers did analysis

of parameters to test the effectiveness of algorithm. Toyama et al. have solved the jigsaw type puzzles and obtained good results for puzzle only up to 64 pieces [11]. Later D. Sholoman et al. attempted to solve the jigsaw puzzles of large size with efficient GA solver [12]. GA was started with the encoded chromosomes in the form of arrangement of puzzle pieces. They obtained significant improvement in the results and their GA solver provided the efficient solution for jigsaw puzzles of 22,834-piece size. The efficient GA was developed using the novel type of crossover. In addition to this solver, they also provided a benchmark of large images to test the other solvers

This work has attempted to apply simple GA and novel GA based on advanced twin operator to solve the number puzzle problems. In our earlier work, the novel GA was introduced and designed using the novel operator called advanced twin operator [13] [14] [15].

This novel GA was rigorously tested on the standard benchmark optimization test functions to verify its effectiveness. The advanced twin operator design was originated by extrapolating the process of twin offspring formation in natural genetics. This operator develops twin offspring individuals in the simulated environment of GA. This operator is associated with the design parameters viz. twin probability and twin separability. The twin probability resembles the frequency of producing twin offspring in natural reproduction systems and set to low static value. The twin separability parameter is related to the number of unequal genes representing the variation in twin's identical appearance and its phenotypic characteristics.

II. Problem Definition for Puzzles

This work is carried out to solve the number puzzles as defined below:

- Find a 10-digit number where the first digit is how many zeros in the number, the second digit is how many 1's in the number etc. until the tenth digit which is how many 9s in the number.
- Find a one four-digit whole number n, such that the last four digits of n^2 are in fact the original number n.

ISSN: 2278 – 1323

The puzzles are defined and reported on the web [16]. These puzzles have the exponential complexity and seem to be little bit difficult if solved by deterministic methods. GAs are guaranteed to find near optimal solution for such types of problems. Before presenting the application of novel GA, the advanced twin operator designed for novel GA is explained below in brief.

III. Novel GA with Advanced Twin Operator

The research work on the advanced twin operator for GA was carried out in our previous research work [13], [14] [15]. The design and development of the advanced twin operator is described below in brief.

The design of the advanced twin operator is abstracted from the process of twin offspring generation in natural genetics. In this case, the formation of twin offspring due to single ovulation is taken into consideration. The process of generating twin offspring involves the selection of fit parents for reproduction, mating and crossover of these parents. After crossover, the advanced twin operator is applied to form twin offspring individuals. The selection operator of GA is responsible for selecting the fit parents for reproduction. Similar to natural system, one of the fit parents is equivalent to single ovum in the single ovulation based reproduction process. The elitist strategy of GA enforces that the best individual or the individual of first rank of the current generation should be propagated as it is to the next generation. Therefore the second rank individual or individual next to the best individual is selected for reproduction as the one of the parents resembling the single ovum in single ovulation. This parent is referred to as P_1. The other parent is randomly selected from the current generation. This parent is referred to as P_2. The two parents are crossed using any conventional crossover method to generate two offspring say $child_1$ and $child_2$ respectively.

After this crossover, the advanced twin operator is designed and applied as described below. After reproduction, the hamming distance of each child from both the parents is calculated. Hamming distance indicate the unequal genes of the child from the respective parent. Consider that H_1 and H_2 are denoted as the hamming distances of $child_1$ from P_1 and P_2 respectively. The advanced twin operator should randomly select exactly half the number of unequal genes from H_1 as well as H_2 and change only the values of these genes by keeping all other genes same as that of $child_1$. This creates the first twin mate child say $child_3$. In this way, the advanced twin operator generates the first twin pair $child_1$:$child_3$. The locations of unequal genes from H_1 and H_2 have considerable effect on the decoded value of the twin mate child. Similar process is repeated with $child_2$ to generate its twin mate $child_4$. This generates the second twin pair $child_2$:$child_4$. The design parameter of twin operation is the probability of twin operator say 'P_{twin}'. It should be low as it resembles the frequency of twining in natural genetics. The other design parameter is number of unequal genes randomly selected from H_1 and H_2. It affects the twin separability parameter that increases in proportion to the number of unequal genes. The performance of GA varies in accordance with these design parameters.

IV. Application of Simple and Novel GA to Solve Puzzles

The puzzles are solved by designing and implementing the SGA and ATGA. The general structure of SGA is modified with the creation of twin offspring using advanced twin operator with its parameters viz. twin probability and twin separability. The general flow of novel GA with advanced twin operator ATGA is presented below:

ATGA ( )
{

Randomly generate initial population of fixed population size;

Repeat till stopping criteria not met

{

Evaluate each individual as per fitness function;

Carry forward best individual as per elitist strategy;

Select parents by using any selection method on remaining individuals;

Perform crossover with crossover probability and apply advanced twin operator to generate twin offspring with specific twin probability;

Apply mutation with low probability;

}

Simple GA (SGA) and novel GA with advanced twin operator called as ATGA are designed as per the structure to solve each puzzle. In the design, the important GA parameters set are encoding method, population size, crossover method and its probability P_c, mutation and its probability P_m, and the termination criteria. In ATGA, the twin probability P_{twin} is the main design parameter.

For solving 10-digit puzzle, the GA individual is encoded as the string of 10 decimal digits generated randomly. The population size is set to 40 after lots of experimentation trials. The parents for crossover are selected using tournament selection method. The single point crossover with crossover probability P_c of 1 is used to achieve maximum exploration. The mutation probability P_m is set to low static value 0.05. The termination criterion is set as the maximum number of generations. The fitness function is designed to meet the objective of the puzzle and later scaled to yield the appropriate value. In ATGA, all the other GA parameters except twin probability and separability are kept same as that of SGA. The value of twin probability P_{twin} is set to fixed value 0.05 after lots of experimentation trials for P_{twin}. The twin separability is set to 50%.

In the case of a whole number 4-digit puzzle, the GA individual is encoded as the string of 4 decimal digits generated randomly. The tournament selection method is used for selecting parents. The single point crossover with its probability 1 is applied to generate next generation offspring. The rest of the parameters viz. population size, mutation probability, termination criteria are set after carrying out the number of experimentation trials. Further the population size is varied and step by step reduced to 6 as used in the typical micro-GA keeping all other parameters same. The effect of population size is observed on the convergence of GA. It is found that more time is required to converge when the population size is reduced from 20 to 6 in stepwise manner.

The complete experimental set up of GA parameters for the puzzle problems under consideration is displayed in Table 1. For each puzzle, SGA and ATGA are implemented as per the design and executed number of times to find out the puzzle solution in terms of the best individual. Besides best individual, the other performance parameters viz. convergence generation and average fitness are recorded to assess the effectiveness of novel GA.

Table 1: Experimental Set up of GA parameters

Puzzles	**Encoded String Length in bits**	**Pop_size**	**Selection**	**Crossover (p_c=1)**	**Fixed *Pm***	**Max. gens**
10-digit puzzle	10	40	Tournament with replacement	Single point	0.05	25
4-digit puzzle	4	20	Tournament with replacement	Single point	0.05	25
4-digit puzzle	4	6	Tournament with replacement	Single point	0.02 to 0.05	25

Table 2: Performance parameters for SGA and ATGA

Parameters	**Puzzle solution**	**SGA**		**ATGA**	
Puzzles		**Best Convergence Generation**	**Average Convergence Generation**	**Best Convergence Generation**	**Average Convergence Generation**
10-digit puzzle	6210001000	16	22	8	18
10-digit puzzle	7100000100	16	21	7	16
4-digit puzzle	9376	8	18	3	15
4-digit puzzle	0625	6	14	4	10
4-digit puzzle	0001	5	12	4	10

V. Results and Discussions

For solving each puzzle, SGA and ATGA are designed and executed for repeated number of trials 25. The final solution for the puzzle is recorded and presented in terms of the average number of trials. It is interesting to note that for 10-digit puzzle, one extra solution (7100000100) than the unique solution (6210001000) reported is found and displayed in Table 2 [16]. In the case of 4-digit puzzle, the solutions found in terms of the best individual encoded as 4-digits are 9376, 0625 and 0001 as displayed in Table 2. Out of 3 solutions, only 9376 is a valid solution considering 4 digits.

The performance of SGA and ATGA is measured in terms of convergence generation, best individual and average fitness. The results in terms of these performance parameters are displayed in Table 2. It is observed from this Table 2 that the best and average values of the convergence generation are less for ATGA than that of SGA in solving both puzzles. This shows that the convergence occurs earlier in ATGA than that of SGA indicating the positive effects of advanced twin operator. The graphs of the best individual against the number of generations for 4-digit puzzle are displayed in Figure 1 for one of the executed trials.

From the Graph 1, it is observed that the best individual appears earlier in ATGA compared to that of SGA indicating fast convergence. Similar results are also observed for 10-digit puzzle.

For 4-digit puzzle, the effect of varying population size is also observed on the overall performance of GA. As the population size reduces, it takes more time to yield convergence. But for the population size of even 6 similar to that of micro-GA, the final solution is obtained after certain increased number of generations.

ISSN: 2278 – 1323

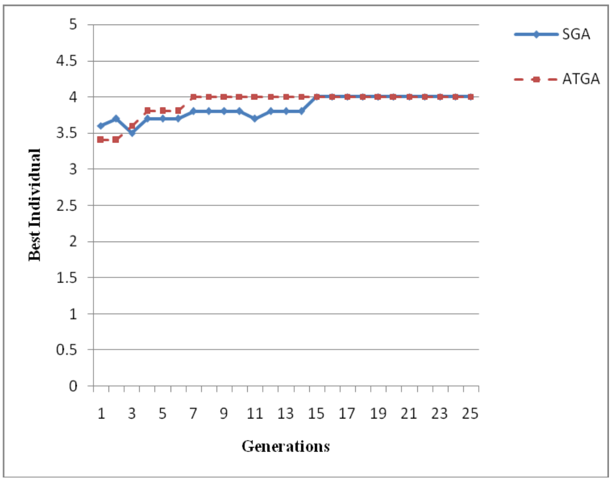

Figure 1: Best Individual Variation according to Generations

By the method of repeated trials, it is also found that for population size of 6, the greater values (>0.05) mutation probability are required to get the solution quicker.

VI. Conclusion and Future Scope

For 10-digit puzzle, the deterministic approach takes exponential number of ways to find out the final solution. Compared with this, GA finds the unique solution in significantly less number of function evaluations. From the results presented in Table 2, it is concluded that the convergence occurs fast for ATGA than that of SGA showing the positive effect of the advanced twin operator in reducing the number of function computations. The graph of best individual shows the generation wise improvement leading to the final solution after reaching the average number of generations. From the execution of SGA and ATGA, it is also observed that the average fitness improves as the number of generations goes on increasing. In the case of 4-digit puzzle, the final solution is obtained for the very small population size 6 similar to that of micro-GA. This indicates the capability of GA to find out the final solution. The extensions of the current work include the application of ATGA to solve more complex puzzles and games. Besides this, it would be interesting to try the adaptive approach of twin probability of advanced twin operator for solving the puzzles and verify its effectiveness on the performance of GA.

REFERENCES

[1] Simonis, H.: Sudoku as a constrain problem. In B. Hnich, P. Prosser, B. Smith (eds.), Proc. 4th Int. Works. Modelling and Reformulating Constraint Satisfaction Problems, 2005, 13–27

[2] Lynce, I., Ouaknine, J.: Sudoku as a SAT problem. In 9th International Symposium on Artificial Intelligence and Mathematics AIMATH'06, January, 2006

[3] Moon, K., Gunther, J.: Multiple constrain satisfaction by belief propagation: An example using Sudoku. In 2006 IEEE Mountain Workshop on Adaptive and Learning Systems, July 2006, 122 - 126

[4] Timo Mantere, Janne Koljionen, "Solving, Rating and Generating Sudoku puzzles with GA", in proc. Of 2007 IEEE congress on Evolutionary Computation, pp. 1382-1389, 2007.

[5] Ardel, D.H.: TOPE and magic squares, a simple GA approach to combinatorial optimization. In J.R. Koza (ed.) Genetic Algorithms in Stanford, Stanford bookstore, Stanford, CA (1994)

[6] Alander, J.T., Mantere T., Pyylampi, T.: Digital halftoning optimization via genetic algorithms for ink jet machine. In B.H.V. Topping (ed.), Developments in Computational mechanics with high performance computing, CIVIL-COMP Press, Edinburg, UK (1999) pp. 211-216

[7] Jose B. Fonseca, "The Magic Square as a Benchmark:Comparing MIP to improved GA and to a High Performance Minimax AI Algorithm.

[8] Technical Report by Sandip Barui, M Sc. Second year project, Dept. Of Mathematics, IIT Bombay.

[9] S. Lucas and G. Kendall, "Evolutionary computation and games," IEEE Computational Intelligence Magazine , vol. 1, no. 1, pp. 10-18, 2006

[10] Zongwoo Geem, "Harmony search algorithm for solving Sudoku", Proceedings of the 11th international conference, KES 2007 and XVII Italian workshop on neural networks conference on Knowledge-based intelligent information and engineering systems: Part I , pp. 371-378, Springer-Verlag Berlin, Heidelberg, 2007

[11] F. Toyama, Y. Fujiki, K. Shoji, and J. Miyamichi. Assembly of puzzles using a genetic algorithm. In IEEE Int. Conf. On Pattern Recognition , volume 4, pages 389–392, 2002.

[12] Dror Sholomon, Omid David, Nathan S. Netanyahu: A Genetic Algorithm-Based Solver for Very Large Jigsaw Puzzles. CVPR 2013: 1767-1774.

[13] Anagha Parag Khedkar and Subbaraman Shaila, "Novel Twin Operator for Genetic Algorithm", proceedings of 3rd International conference on Data Management and Advances in Computer Science and Engineering, pp. 111-117, 2010.

[14] Anagha Parag Khedkar and Subbaraman Shaila, "Effect of Advanced Twin Operator on the performance of Genetic Algorithm", International Journal of Engineering Research and Technology, vol. 3, pp. 721-731, 2010.

[15] Anagha Parag Khedkar and Subbaraman Shaila,"The Novel Approach of Adaptive Twin Probability for Genetic Algorithm", International Journal of Advanced Studies in Computers, Science and Engineering, vol. 2, special issue 2, pp. 31-37, Sept. 2013.

[16] www.mathsisfun.com/puzzles/number-puzzles-index.html

About Authors:

Mrs. Anagha P. Khedkar received her B.E. degree in Electronics Engineering from Shivaji University Kolhapur in 1991. In 2003, she also received her M.E. degree in Electronics Engineering from the same university. Now she is pursing PhD from Shivaji University Kolhapur in the area of 'Evolutionary Computation' under the guidance of experienced professor Dr. Shaila Subbaraman. Presently she is working as an associate professor in the Matoshri college of Engineering and Research Centre, Nashik, Maharashtra (India). She is having 22 years of teaching experience. Her areas of interest are evolutionary computation and embedded systems.

Dr. Mrs. Shaila Subbaraman received her M. Tech in Physical Engineering from Indian Institute of Science, Bangalore in 1975. She worked in manufacturing industries in India related with semiconductor devices and hybrid integrated circuits for a period of 14 years in various capacities. She joined Walchand College of Engineering, Sangli (Maharashtra) in 1989 and was deputed to Indian Institute of Technology, Bombay to pursue her Ph. D. in 1994 which she completed in 1999. She has worked as Ex-Dean Academics and Professor at Walchand College of Engineering, Sangli. She has guided many research students. Her areas of interest are Microelectronics, Low power VLSI Design, Architectures for reconfigurability of hardware etc.

