iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 5 **Issue:** VI **Month of publication:** June 2017

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

Volume 5 Issue VI, June 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Applying Genetic Algorithm on Dynamic Programming Problems

Sangeeta[1], Anshul Bhatia[2], Eisha Jain[3]
[1]*M. Tech Student DCSA, MDU, Rohtak (Haryana)*
[2]*M. Tech Student CSE, RTU, Jaipur(Rajasthan)*
[3]*MCA student Gurukul Kangri Vishwavidyalaya, Haridwar, India*

Abstract- GA is one of the mostly used heuristic search methodology in the present time. Without evolutionary programming and evolutionary algorithms the whole search process is wasted. GA is applied in various fields of search and optimization techniques but now we try to apply this on the Dynamic programming. Various kind of greedy and dynamic programming algorithms are available but we try to enhance their space and time complexity by using these Genetic Algorithms. Sometimes dynamic programming also known as dynamic optimizations, which mean solving the problem by divided the same problem into smaller sub problems for the purpose of better understanding. In this paper we can try to enhance the output of dynamic programming by using Genetic Algorithm in it.
Keywords - Genetic Algorithm, Dynamic Programming, Greedy Programming, LCS, Knapsack, Matrix chain multiplication etc.

I. INTRODUCTION

A. Genetic Algorithm

GA is one of the most effective kinds of heuristic approach which can find out the most optimal solution in the search space. It can be used with various kinds of problems and resultant solution must be better than previous solution. But there is some limitation of simple GA like premature convergence and population diversity. But these limitations can be resolved by using fitness scaling functions with in GA. There are various steps in GA like selection, crossover, mutation etc. In selection phase the GA can select the best suitable individuals of populations which can helps to generation best offspring's in next iteration/generation. In crossover phase of GA the two best selected parents are used for crossover process and the result will be better offspring that are generated are used for further process. The fitness of child offspring's are much better than parents populations. The next step i.e. mutation is also helps to generate more accurate results by mutating the bits of best two individuals of populations. Hence by applying all these processes the GA will assure that the results must be global optimum and not converged to local optimum. Some of the numerical parameter of GA are:-

1) Initial Population Type
2) Population Size
3) Maximum Generation Number
4) Crossover Probability
5) Mutation Probability

B. Dynamic Programming

It solves the problem by storing the value or result of smaller problems and use it into bigger problems. Dynamic programming is itself a very powerful technic which can solve very complex and huge calculations and reduces the coding complexities a lot. It can solve the exponential time problems into polynomial time very effectively. Basically we can say that it uses "divide and conquer" technique. It produces much better results than Greedy Programming approach because sometimes Greedy method select locally best solutions as a sub parts. That's why it produces faster result than Dynamic method but not always optimum. On the other hand Dynamic approach always provides globally optimum solutions. Figure 1 shows the procedure of Dynamic programming.

Volume 5 Issue VI, June 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

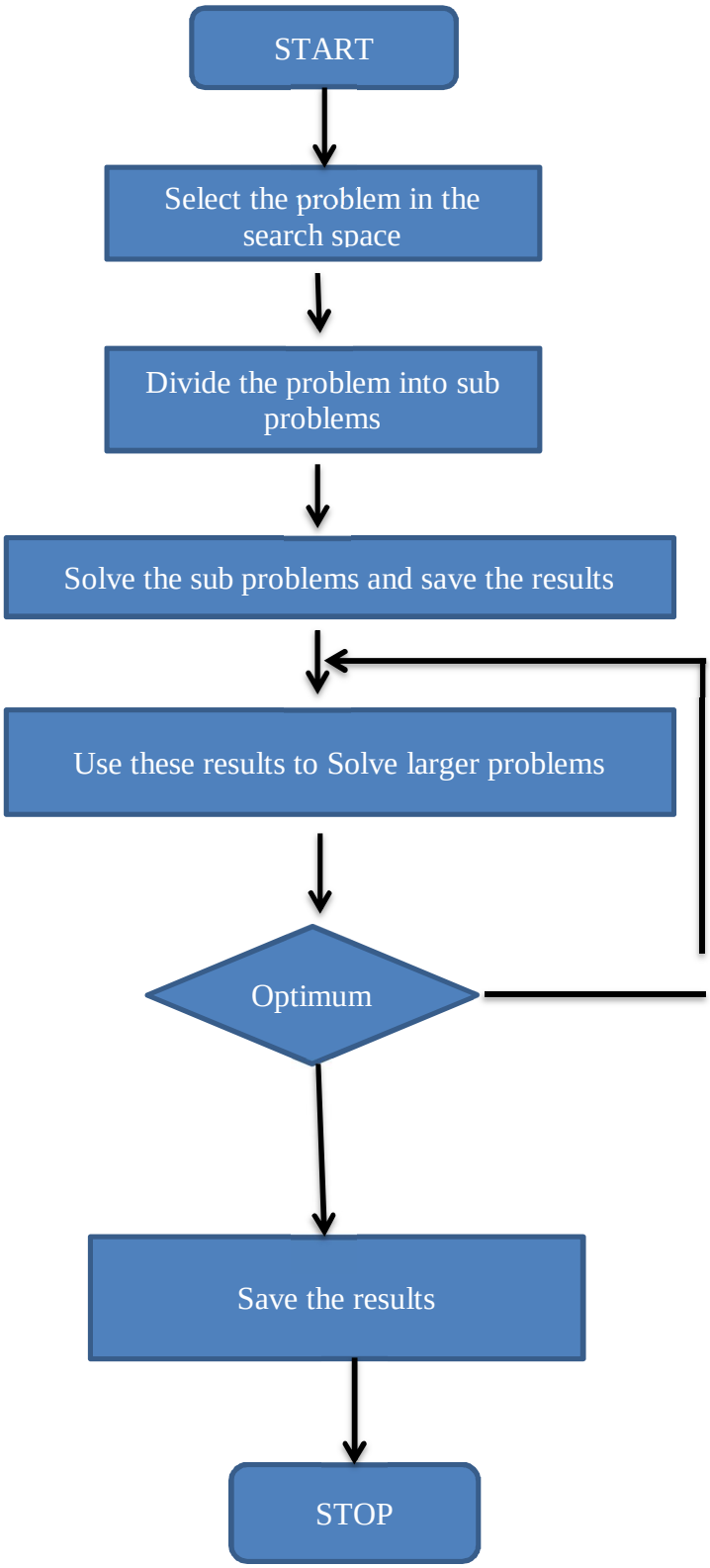


Fig.1 Flowchart of simple Dynamic Programming Problem

II. EXAMPLES OF DYNAMIC PROGRAMMING APPROACHES

There is much kind of Dynamic programming problems but we can only try to implement GA with some of them. These are -

Volume 5 Issue VI, June 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

A. LCS (longest common subsequence

It is one of the dynamic problem which can uses the sub parts of problem and calculate the common part of two inputs. It does not means that the result in contiguous order. In Figure 2 input strings "AGGTAB" and "GXTXAYB". Last characters match for the strings. So length of LCS can be written as L ("AGGTAB", "GXTXAYB") = 1 + L ("AGGTA", "GXTXAY")

	A	G	G	T	A	B
G	-	-	4	-	-	-
X	-	-	-	-	-	-
T	-	-	-	3	-	-
X	-	-	-	-	-	-
A	-	-	-	-	2	-
Y	-	-	-	-	-	-
B	-	-	-	-	-	1

fig.2 Example of LCS

Let the input strings "ABCDGH" and "AEDFHR. Last characters do not match for the strings. So length of LCS can be written as: L ("ABCDGH", "AEDFHR") = MAX (L ("ABCDG", "AEDFH**R**"), L("ABCDG**H**", "AEDFH")). So the LCS problem has optimal substructure property as the main problem can be solved using solutions to sub problems.

B. Knapsack Problem

It is a kind of NP complete problem which means the time period to solve this problem is not polynomial and whenever used with Genetic algorithm then it's greatly affects the output performance. Here the N items are given and their corresponding value Vi and weight Wi is given. We have to select the items in such a way that we have to get maximum benefit. Here also we try to enhance the results by using GA, Branch and bound, dynamic programming etc. with it.

C. Matrixes Chain Multiplication

It is also a dynamic programming problem in which we just try to multiply multi-dimensional array in best effective time. In this when we apply dynamic programming the we first try to solve the entire sub parts then use these sub parts for further calculations by this we can reduce number of calculation and reduce time required for calculations. After that we can compare the results of simple dynamic approach with the results find out by applying GA.

III. RELATED WORK

There is lot of research work done in the field of GA applying with Dynamic Programming Problems but We try to enhance those results by using more advanced methods for enhancing the results. Some of those researches are as follows –

Mateusz Wibig in his paper concluded that the Genetic algorithm is very effectively works with Dynamic programming in the field of business optimization very efficiently. It can uses all the business parameter very reliably without any need to recalculate them from beginning every day. It can focus on each changing parameter which can improve the growth of any business and enhance the profit. Basically it can take care of the variable as well as static parameter of business in advanced which is very difficult by manual procedure[11].

Thomas, Gulter And Engelbert in their paper concluded that the evolutionary and dynamic programming problem helps in the finding of facility layout problem. Which can helps a lot to decrease the cost of material handling, manpower requirement, flow of information, work in process inventory etc. they just can enhance the search space used by Yang and Peter and also improves the results of previous work done [3].

Ameen Shaheen and Azzam Sleit conclude in their paper that whenever knapsack problem is used with dynamic programming, branch and bound, greedy approach, genetic algorithm then it can produce different kind of results. It means if it can applied with

genetic and greedy approach then it can suffer by genetic diversity and effective tine gradually increases exponentially but produces best results with dynamic programming and branch and bound approaches[12].

Maya, Hristakeva and Dipti Shrestha can concluded that after applying brute force, GA, Dynamic programming, Greedy algorithm and branch and bound they find out that dynamic and GA can produces best outputs. But somehow in coding sense dynamic approach proves to be better than genetic algorithm because of its complex paradigm [13].

S. Pothumani in his paper try to evaluate the N queens problem with various kind of heuristic search methods and evaluate results . He can observe that the N queens problem produce better results with ACO (ant colony optimization), tabu search, PSO (particle swarm optimization), GA etc. and get that the results produced are much better than previously used backtracking methods. The output are highly depends on size of board and number of queens. He just try to compare the output generated by all heuristic approaches [4].

IV. CONCLUSION

This paper can be concluded that the GA can greatly enhance the power of simple dynamic programming problems by reducing its space and time complexity at a great extent. Here we can try to apply GA with LCS (longest common subsequence), knapsack problem and Matrix chain multiplications. But the further future scope of this paper is that we can able to apply GA with another kind of Dynamic problems like N queen's problem, Shortest common subsequence, traveling salesman problem etc.

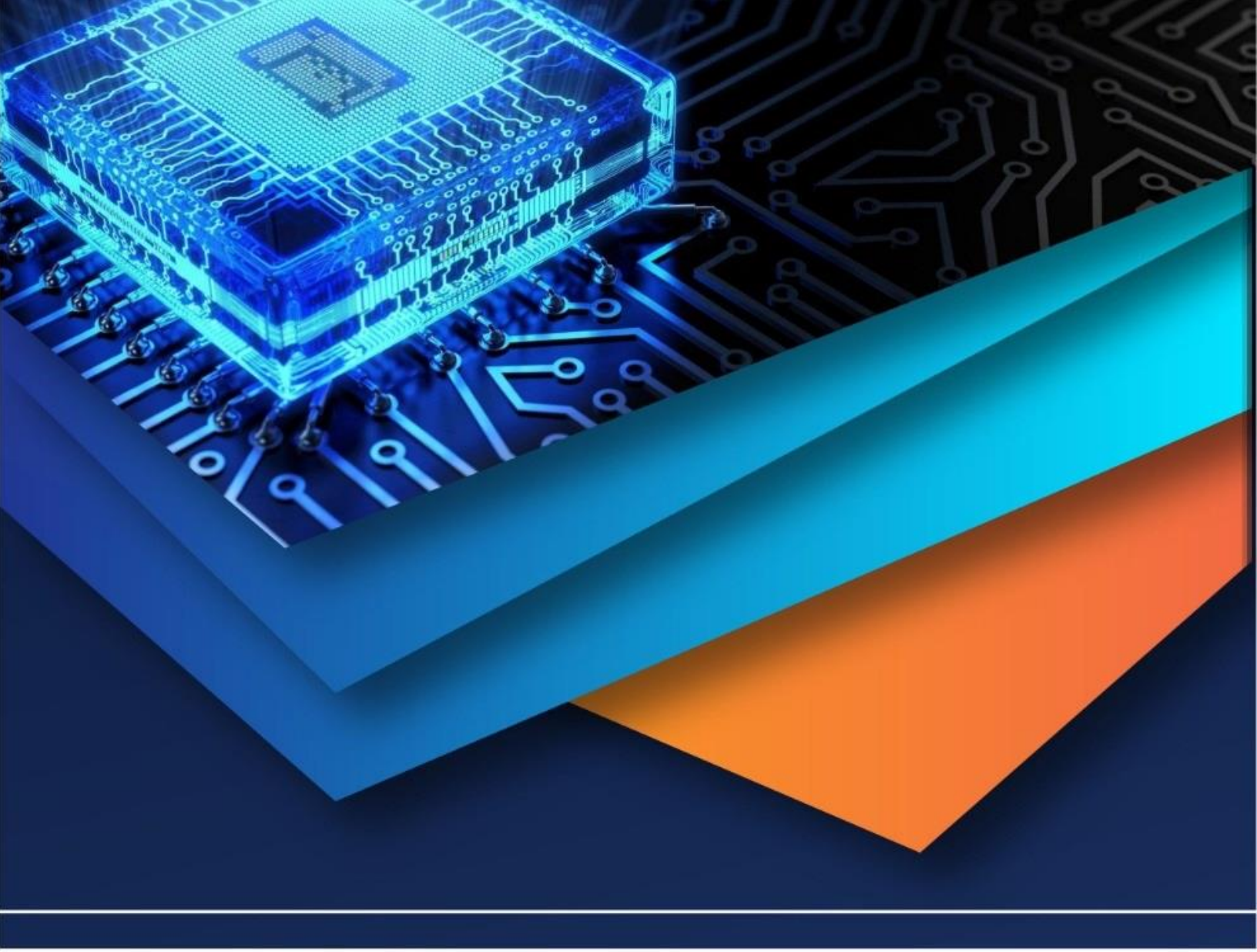
We can also able to try this GA with all kind of Greedy programming problems. So basically we try to enhance the output of any kind of problem by increasing its speed and reduce its processing time and complexity.

V. ACKNOWLEDGEMENT

I am grateful to Mr. Balkishan Singh, Assistant Professor, Dept. of Computer Science and Applications whose insight and advice proved useful during research work. As a supervisor, he has constantly encouraged me to remain focused and his constant support and motivation helped me in achieving this goal. Without him, none of the research work would have been possible. His observation and comments helped me to establish the overall direction of the research and to move forward with investigation in depth. I sincerely thank everyone who has provided me with new ideas, constructive criticism, inspirational words and their invaluable time.

REFERENCES

[1] Held, M., Karp, R.M.: A dynamic programming approach to sequencing problems. Journal of the Society for Industrial and Applied Mathematics. vol. 10, pp. 196--210 (1962)
[2] D.E. Goldberg.: Genetic Algorithms in Search, Optimization and Machine Learning. (Addison-Wesley, New York, 1989)
[3] Thomas, Gulter And Engelbert : Combining Evolutionary Computation with Dynamic programming for solving Dynaic Facility layout problem. Operational Research Proceeding KOI, 2002, pp. 287-29
[4] S. Pothumani: Solving N Queen Problem Using Various Algorithms – A Survey. International Journal of Advanced Research in Computer Science and Software Engineering, vol 3, issue 2, pp 247-250
[5] Ivica Martinjak, Marin Golub ,Comparison of Heuristic Algorithms for the N-Queen Problem (Citations: 5), International Conference on Information Technology Interfaces - ITI , pp. 759-764, 2007
[6] Milena Lazarova: Efficiency of parallel genetic algorithm for solving N-queens problem on multicomputer Platform,EC'08 Proceedings of the 9th WSEAS International Conference on Evolutionary Computing, Pages 51-56
[7] Ana Madureira, Carlos Ramos, Sílvio do Carmo Silva : Using Genetic Algorithms for Dynamic Scheduling. POMS 14th Annual Meeting, 2003
[8] Anitha Rao, Sandeep Kumar Hegde: Literature Survey On Travelling Salesman Problem Using Genetic Algorithms. International Journal of Advanced Research in Eduation Technology, Vol. 2, Issue 1 (Jan. - Mar. 2015), pp 42-45
[9] Veenu Yadav, Ms.Shikha Singh : A Review Paper on Solving 0-1 knapsack Problem with Genetic Algorithms. International Journal of Computer Science and Information Technologies, Vol. 7 (2) , 2016,pp 830-832
[10] V. Kreinovich, C. Quintana, and O. Fuentes, "Genetic algorithms: what fitness scaling is optimal?," Cybernetics and Systems, vol. 24, no. 1, pp. 9–26, 1993
[11] Chen Lin:A Heuristic Genetic Algorithm Based on Schema Replacement for 0-1 knapsack Problem. Fourth International Conference on Genetic and Evolutionary Computing 2010.
[12] Mateusz Wibig: Dynamic Programming and Genetic Algorithm for Business Processes Optimisation. I.J. Intelligent Systems and Applications, 2013, 01,pp 44-51
[13] Ameen Shaheen and Azzam Sleit: Comparing between different approaches to solve the 0/1 Knapsack problem. International Journal of Computer Science and Network Security, VOL.16 No.7, July 2016.
[14] Maya, Hristakeva and Dipti Shrestha: Different Approaches to Solve the 0/1 Knapsack Problem. MICS 2005 proceeding < www.micsymposium.org/mics_2005/papers/paper102.pdf>.
[15] Hristakeva, Maya and Dipti Shrestha:Solving the 0/1 Knapsack Problem with Genetic Algorithms. MICS 2004 Proceedings. <www.micsymposium.org/mics_2004/Hristake.pdf>.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY