Evolutionary Algorithms

A Project Report

Submitted in the partial fulfilment of the requirements for the

award of the degree of

Bachelor of Technology

in

Department of Computer Science and Engineering

By

ERANKI RAMA SUMANTH (2010030051)

SIVA KARTHIK (2010030329)

KAMEPALLI SHENITH (2010030341)

under the supervision of

Madhukar Rao G

Department of Computer Science and Engineering

K L University Hyderabad,

Aziz Nagar, Moinabad Road, Hyderabad – 500 075, Telangana, India.

March, 2022

Declaration

The Project Report entitled “Evolutionary Algorithms” is a record of bonafide work of Siva Karthik Reddy (2010030329), Eranki Summanth (2010030051), Kamepalli Snehith (2010030341), submitted in partial fulfilment for the award of B.Tech in the Department of Computer Science and Engineering to the KL University, Hyderabad. The results embodied in this report have not been copied from any other Departments/ University/ Institute.

Siva Karthik Reddy

Eranki Rama Sumanth

Kamepalli Snehith

Certificate

This is to certify that the Project Report entitled “Evolutionary Algorithms” is being submitted by Siva Karthik Reddy (2010030329), Eranki Rama Sumanth (2010030051), Kamepalli Snehith (2010030341), submitted in partial fulfilment for the award of B. Tech in Computer Science Engineering to the K L University, Hyderabad is a record of bonafide work carried out under our guidance and supervision. The results embodied in this report have not been copied from any other departments University/Institute.

Signature of the Supervisor

Signature of the HOD

Signature of the External Examiner

ACKNOWLEDGEMENT

First and foremost, we thank the lord almighty for all his grace & mercy showered upon us, for completing this project successfully.

We take grateful opportunity to thank our beloved Founder and Chairman who has given constant encouragement during our course and motivated us to do this project. We are grateful to our Principal **Dr. L. Koteswara Rao** who has been constantly bearing the torch for all the curricular activities undertaken by us.

We pay our grateful acknowledgement & sincere thanks to our Head of the Department **Dr. Chiranjeevi Manike** for her exemplary guidance, monitoring and constant encouragement throughout the course of the project. We thank Dr. P Lalitha Surya Kumari of our department who has supported throughout this project holding a position of supervisor.

We whole heartedly thank all the teaching and non-teaching staff of our department without whom we won't have made this project a reality. We would like to extend our sincere thanks especially to our parent, our family members and friends who have supported us to make this project a grand success.

TABLE OF CONTENTS: -

S.No.	Title	Pg. No.
1	Abstract	6
2	Introduction	7
3	Literature Review	8
4	Flow Process	9
5	Methods	11
6	Implementation	12
7	Conclusion	13
8	References	14

Abstract: -

An evolutionary algorithm (EA) is an algorithm that uses mechanisms inspired by nature and solves problems through processes that emulate the behaviors of living organisms. EA is a component of both evolutionary computing and bio-inspired computing. EAs are inspired by the concepts in Darwinian Evolution. In EAs, the solutions play the role of individual organisms in a population. The mix of potential solutions to a problem is populated randomly first. Then the population is tested for fitness -- how well and how quickly it solves a problem. Next, the fittest individuals are selected for reproduction. The cycle begins again as the fitness of the population is evaluated and the least fit individuals are eliminated.

Introduction: -

An evolutionary algorithm is an evolutionary AI based computer application that solves problems in which employing processes that mimic the behaviours of living things. As such, it uses mechanisms that are typically associated with biological evolution, such as reproduction, mutation and recombination.

Evolutionary algorithms function in a Darwinian-like natural selection process; the weakest solutions are eliminated while stronger, more viable options are retained and re-evaluated in the next evolution—with the goal being to arrive at optimal actions to achieve the desired outcomes.

Evolutionary computation is one of the principal methods in what is called ‘nature-inspired computing.’ Nature-inspired computing, which also includes artificial neural networks, swarm intelligence, and fuzzy logic, has also been called ‘soft computing,’ or ‘computational intelligence’ to distinguish it from symbolic artificial intelligence. Technically, evolutionary computation is as an example of heuristic search, i.e., search by trial and error,

Referred documentations for this project: -

Literature review:

SNO	TITLE	AUTHORS	PUBLICATION	CONCLUSION
1	A notion of graph likelihood and an infinite monkey theorem	Christopher Banerji, Toufik Mansour	arxiv	Used graph methods to plot the probalbiti of occurances
2	A million monkeys and Shakespeare	Jesse Anderson	The Royal Statistical Society	Real life experiment of infinte monkey theorum
3	Computational Intelligence and Metaheuristic Algorithms with Applications	Xin-She Yang,	Hindawi	Most metaheuristic algorithms have successful applications in practice, but their mathematical analysis lags far behind

Flow Process: -

Evolutionary algorithms

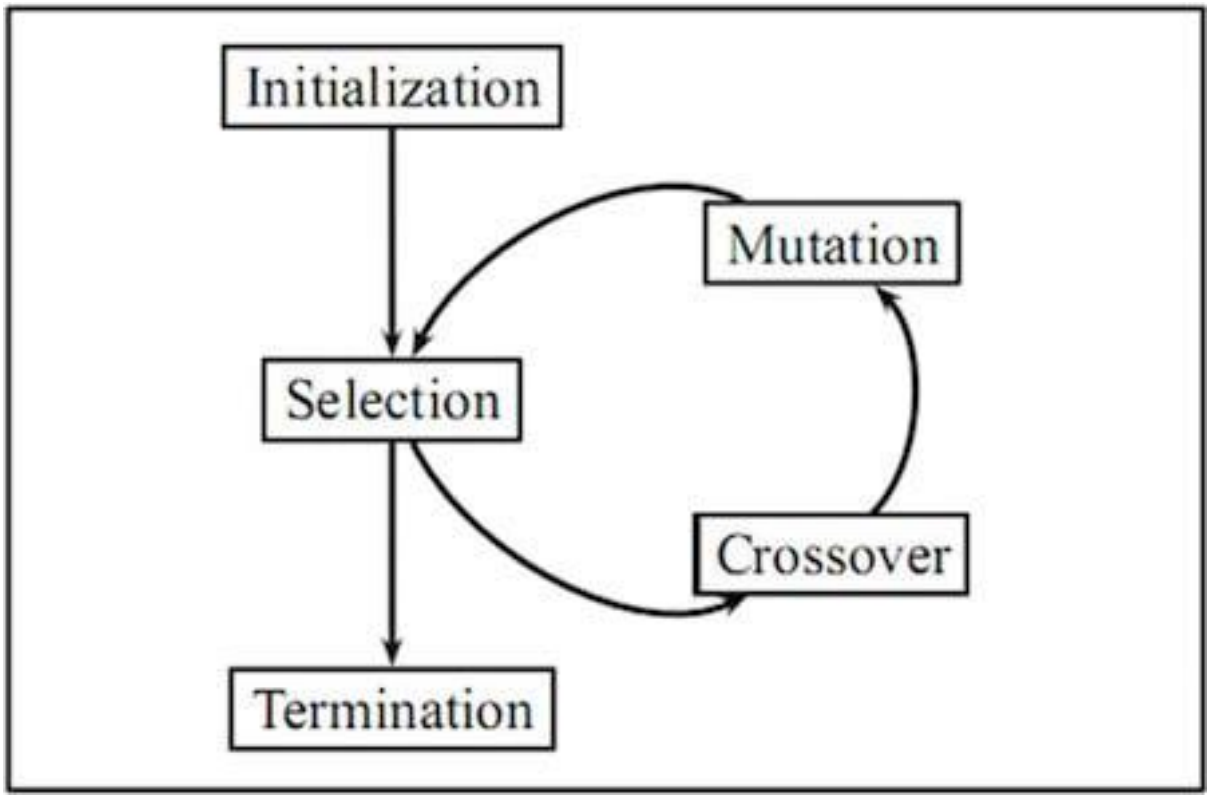

Fig. 1. Flow process of evolutionary algorithms

HARDWARE AND SOFTWARE REQUIREMENTS:

3.1 Software Requirement:

- PYCHARM
- FRONT END: PYTHON
- PLATFORM: WINDOWS 10/11

3.2 Hardware Requirement:

- RAM: 8 GB PROCESSOR: Intel CORE i7 8th gen
- HARD DISK: 128 SSD 1TB HDD

PROPOSED METHODS:

1. Infinite monkey theorem:

The infinite monkey theorem states that if a monkey starts hitting keys at random on a keyboard for an infinite amount of time, he will almost surely type a given text, such as the complete works of William Shakespeare. In fact, the monkey would almost surely type every possible finite text an infinite number of times. However, the probability of this event is so tiny that it will require more time than the estimated age of the universe, but chances of occurrence of this event is not zero.

2. Darwin’s natural selection theory:

Natural selection is defined as a natural process that results in the survival and reproduction of organisms with genetic traits best suited to their environment. A shorter (but no less accurate) definition might be "survival of the fittest." Within any population, the fittest individuals, or the ones who fit the environment best, usually survive and reproduce, passing on their genetic traits to future generations.

Code:

```
import random,string

shakespeare = 'methinks it is a weasel'

quoteLen = len(shakespeare)

def generate():
 char = string.ascii_lowercase+' '
 randchars = ''.join(random.choice(char) for _ in range(quoteLen))
 return randchars

def score():
 scorenum = 0
 randchars = generate()
 shake = shakespeare.split()
 randlist = randchars.split()
 for i,j in zip(shake,randlist):
  if i==j:
   scorenum = scorenum+1
 scorecount = (scorenum / quoteLen) * 100
 return scorecount

def main():
 run = 0
 curScore = 0
 while not(curScore==100):
  curScore = score()
  if (curScore != 0):
    print(run, " = ", curScore)
  run = run + 1;

if __name__ == '__main__':
 main()
```

OUTPUT:

```
8485521  =  4.3478260869565215
8487813  =  4.3478260869565215
8489658  =  4.3478260869565215
8492149  =  4.3478260869565215
8495911  =  4.3478260869565215
8497055  =  4.3478260869565215
8500068  =  4.3478260869565215
8501547  =  4.3478260869565215
8501568  =  4.3478260869565215
8502224  =  4.3478260869565215
8506783  =  4.3478260869565215
8512950  =  4.3478260869565215
8518807  =  4.3478260869565215
8519159  =  4.3478260869565215
8519367  =  4.3478260869565215
8520024  =  4.3478260869565215
8522162  =  4.3478260869565215
8523137  =  4.3478260869565215
8524696  =  4.3478260869565215
8526685  =  4.3478260869565215
8537968  =  4.3478260869565215
8547472  =  4.3478260869565215
8549307  =  4.3478260869565215
8551465  =  4.3478260869565215
8553905  =  4.3478260869565215
8554272  =  4.3478260869565215
8559076  =  4.3478260869565215
8559375  =  4.3478260869565215
8562705  =  4.3478260869565215
```

Conclusion: -

Evolutionary Algorithms are excellent at optimizing solutions. It is important to note though that while Evolutionary Algorithms optimize effectively, they don't necessarily find the optimal solution. Instead, Evolutionary Algorithms constantly find working solutions and measure performance against one another, which may or may not find the absolute best possible solution. Evolutionary Algorithms relatively high computational requirements, which may also be a consideration, are largely due to the complexity of fitness determination. This complexity can be reduced through fitness approximation. As the mechanisms by which Evolutionary Algorithms work are inspired by evolution and living organisms, functions might include selection, reproduction, mutation and recombination. The adaptive process of choosing the best available solutions to a problem where selection occurs according to fitness is analogous to Darwin's survival of the fittest. By testing fitness according to measured performance, optimization occurs over generations through such functions as mutation .

REFERENCES:

[1] Xiufen Zou, Yu Chen, Minzhong Liu, and Lishan Kang". A New Evolutionary Algorithm for Solving Many-Objective Optimization Problems"

[2 Nicholas J. Radcliffe & Patrick D. Surry. "Fundamental Limitations on Search Algorithms: Evolutionary Computing in Perspective"

[3] Darrell Whitley. "An overview of evolutionary algorithms: practical issues and common pitfalls"

[4 David A. Van Veldhuizen and Gary B. Lamont. "Multiobjective Evolutionary Algorithm Test Suites"

[5 Franz Rothlauf." Representations for Genetic and Evolutionary Algorithms"

[6] Christopher Banerji, Toufik Mansour, and Simone Severini. "A notion of graph likelihood and an infinite monkey theorem"

[7 Ferdinando, "Form Darwin's Origin of Species toward a theory of natural history"