International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 09 | Sep 2020 www.irjet.net p-ISSN: 2395-0072

FPGA IMPLEMENTATION OF BAT ALGORITHM FOR OPTIMIZATION

Atharva A. Kankal[1], Neha R. Jagtap[2], Shraddha S. Sankpal[3], Satyashil D.Nagrale[4]

[1,2,3]*Student, B.E. Electronics & Telecommunication*
[4]*Assistant Professor, Dept. of Electronics & Telecommunication, Pimpri Chinchwad College of Engineering, Pune Maharashtra, India*

Abstract -*In this paper, we propose a metaheuristic method, the Bat Algorithm which is implemented on FPGA is presented. The Bat Algorithm is now becoming a powerful algorithm to solve many tough optimization problems and also to perform global optimization. The Bat Algorithm has good performance compared to the other most famous algorithms such as particle swarm optimization(PSO) or the genetic algorithm(GA). The application Bat Algorithm in our research is based on a mathematical model based on the echolocation behavior of Bat. Simulations show that the proposed algorithm seems much superior to other algorithms.*

Keywords: Metaheuristic, BAT Algorithm, FPGA, Optimization, Echolocation.

1. INTRODUCTION

1.1 BAT Algorithm

Bat algorithm is one of the new heuristic optimization algorithms that was recently proposed by Yang. The latest edition of the bat algorithm is the Bat Inspired search(BIS). The algorithm uses the echolocation behavior of bats in searching for an optimum solution. Indeed, bats use several methods to detect their prey or other shapes around them and even in dark places. They achieve the prey by emitting a sound pulse to the air and they listen foe echos reflect from them. Generally, echolocation calls are presented by three important features; the pulse emission rate, the frequency, and the loudness. After that, the information is collected and calculated in the brain to make a virtual image of their surrounding. The bat algorithm gives satisfactory results in solving many dispatch problems related to biology medical, finance, 3d graphics, image processing, and others. The bats and some other animals use echolocation; it is an advanced search based on a navigation system to detect any objects in their surroundings by emitting a sound to the environment and returns to them as an echo. The direction and intensity of the returned signal enable them to locate the direction and distance of potential prey. Besides, they have a surprising ability to quickly differentiate between an obstacle and a prey. At first, the bat flies blindness around the search space while emitting a sound wave of certain amplitude(intensity)and pulse rate. Between the pulse rates, it receives feedback signals (it's signal and possibly signals from other bats swarm) by echolocation and interprets these sound waves. If the received signals have a low intensity or strong rate then it is likely that prey is detected and the bat should run toward it. Gradually, as the bats approach prey, it increases the number of pulses(pulse rate) and at the same time decreases the intensity of the pulses(loudness). But if the received signals are very low levels, it continues its flight blindly without changing the intensity of the emitted sound wave. The emitted sound travels in zones which has the same atmospheric air pressure and at a constant speed to follow the time delay of the returning echoes. Moreover, bats use the loudness of the received signals each time to identify the direction of the shapes. This information allows the bats to make a virtual image of their environment.

2. METHODOLOGY

2.1 Position Vector

To simplify the design of our BA, the algorithm focused on the main characteristics of bats to accelerate finding the prey. In fact, the artificial bats use three vectors; the first vector is the position, the second one is the velocity and the third one is the frequency vector. All vectors are updated throughout the race of iterations by the following equations (1) and (2).

$$X_i(t+1) = X_i(t) + v_i(t+1) \quad \text{.....(1)}$$

$$V_i(t+1)=V_i(t) + (X_i(t) - G_{best})\, f_i \quad \text{.....(2)}$$

Where $X_i(t+1)$ is the position of bats at iteration t+1 and $X_i(t)$ at iterations t. G_{best} is the best value reached and it is obtained Via the optimization process which represents our global best solution found f_i is the i^{th} bat frequency and $v_i(t)$ is the velocity of the bat. Considering that bats move by flying and that the solutions of the search space "S" are positions in space. At each time t, each of N bats in the population has a position in space S and velocity V_i.

2.2 Velocity Vector

Each artificial bat has a velocity vector that is updated during the course of iterations. At initialization, the bats are uniformly distributed in the search space. The initial velocity is reduced around the prey. The prey is detected when the amplitude and the pulse rate perceived by the bat are less than the emitted loudness, therefore higher than the emitted rate.

$$V_i(t + 1) = V_i(t) + (X_i(t) - G_{best})f_i$$

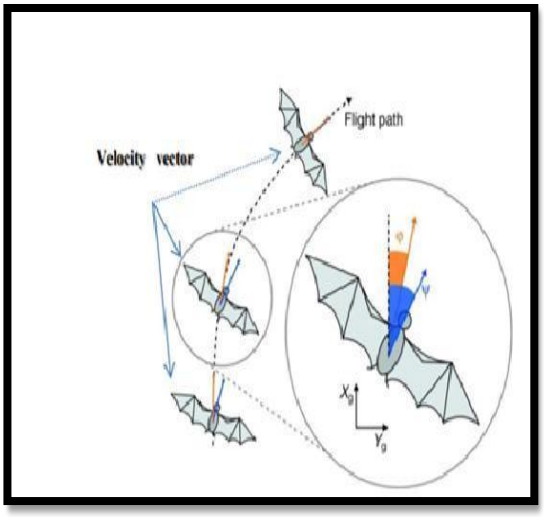

Fig 2.2:- Velocity vector

2.3 Frequency Vector

The frequency f_i is generated uniformly in the range of (f_{min}, f_{max}) and it allows controlling the rhythm of the movement. The new position is obtained by adding the new velocity to the current position. Thus the distance to the prey is estimated by the Doppler effect by varying the frequency of the emitted signal wave if it is the i^{th} frequency of the bat and it is updated after every iteration using the following equation:

$F_i = f_{min} (f_{max} - f_{min})B$ .....(3)

B is a value chosen randomly from a uniform re-partition between [0,1]

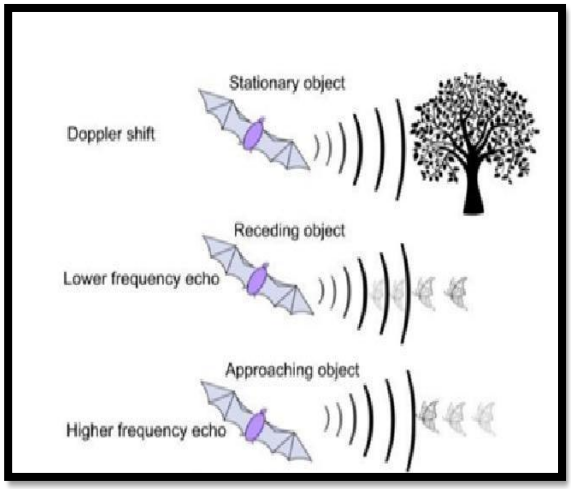

Fig 2.3:- Frequency vector

3. PROPOSED SYSTEM

3.1 Block diagram

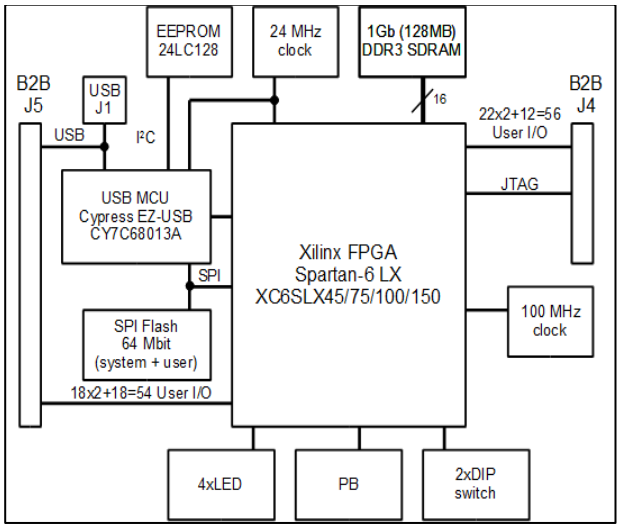

Fig3.1:-Block Diagram

3.2 Explanation

A Field-programmable Gate Array (FPGA) is an integrated circuit designed to be configured by a customer or a designer after manufacturing hence the term "field-programmable". The FPGA configuration is generally specified using a hardware description language (HDL). FPGAs contain an array of programmable logic blocks and a hierarchy of reconfigurable interconnects that allow the blocks to be wired together like many logic gatesthat can be inter-wired in different configurations. Logic blocks can be configured to perform complex combinational functions or merely simple logic gates like AND and XOR. In most FPGA's, logic blocks also include memory elements which may be simple flipflop or more complete blocks of memory. The Xilinx Spartan-6FPGA family delivers an optimal balance of low risk, low cost, low power, and performance for the cost-sensitive application. These FPGAs use a proven low power 45nm process technology. Also, the Spartan-6 series offers advanced power management technology up to 150k logic cells, integrated PCI express blocks, advanced memory support 250MHZ DSP slices, and 3.2 Gbps low power transceivers.

3.3Results

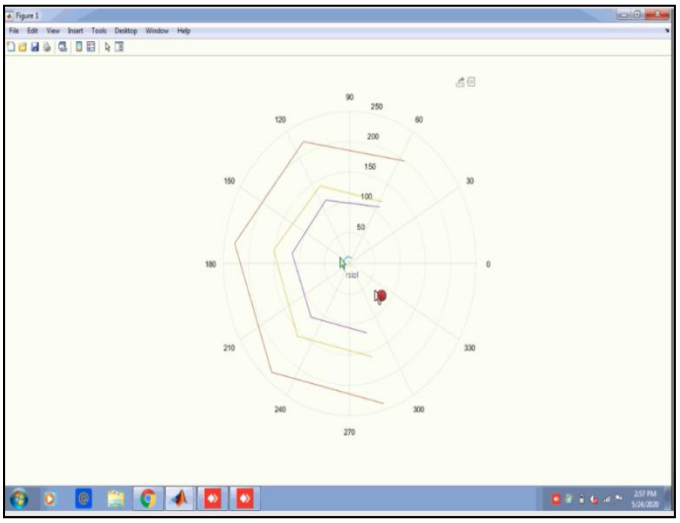

Fig3.3.1:- Simulation results

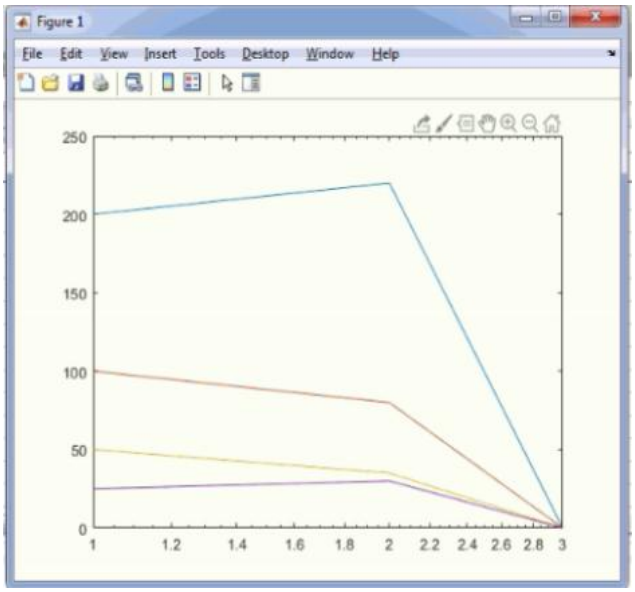

Fig3.3.2:- Simulation results

At the random position, if a random pulse rate is greater than pulse rate RI at current position x_i then it chooses a solution among the best solution and generates a local solution around the best solutions.

Further, it generates new solutions, if loudness is minimum and maximum emission or pulse rate then BAT updates the solutions(increase the pulse rate and reduces loudness) and rank the bats and then finds the current best solution. This process is repeated until it reaches the specified maximum iterations.

4. CONCLUSIONS

Our work will contribute to the implementation of FPGA an optimization algorithm called BA experimented in many benchmark functions. The Bat Algorithm has superior features, including quality of the solution, good computational efficiency, and stable convergence characteristics. The proposed architecture of the bat algorithm will be helpful to prove that it has a favorable convergence speed compared to most other meta-heuristic techniques depends upon the size of design space, it means that the number of bats allocated and the complexity of the problem.

ACKNOWLEDGEMENT

The author would like to thanks many individuals who, as a team member, made a key contribution to the successful execution of the "FPGA IMPLEMENTATION OF BAT ALGORITHM FOR OPTIMIZATION". Team members, in alphabetical order, were **Atharva Kankal, Neha Jagtap, and Shraddha Sankpal.** The authors would like to thank our project guide, **Prof.S.D.Nagarale** who made us aware of various projects and resources to start our project.

REFERENCES

1. Mathematical methods in large-scale computing units. Ann. Computing Lab. Harvard University.1951
2. Adaptation in natural and artificial systems. Arbor, MI: The University of Michigan Press; 1975.
3. Implementation of real coded Genetic Algorithms using FPGA technology 10th international Multi-Conference on Systems,
Signals & Devices (SSD) 2013.
4. Implementation of real coded PSO algorithms using FPGA technology; 15th international Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA), 2014.
5. Binary bat algorithm Seyedali M,
Neural Computation & Application (2014) 25:663–681; 2014.