Solving Numerical Integration Using Firefly Optimization

1Mithun Roy, 2Debaditya Kundu, 3Subhashis Majumder

1,2 Dept. of Computer Science and Engineering, 2 Dept. of Information Technology
3 Siliguri Institute of Technology Siliguri, India
Email: 1 mathmithunroy@gmail.com, 2 debaditya.kundu@gmail.com, 3 subhashis.majumder@heritageit.edu

Abstract—Firefly Algorithm is one of the meta-heuristic algorithms that are capable of performing optimization quite efficiently. It is a relatively new concept being inspired by the flashing behavior of fireflies. In this work, we have used it effectively in calculation of definite integrals, especially when the integrands have primitives that cannot be expressed using elementary functions. Our application gives results that are more exact than the results obtained by classical numerical methods even when the number of intervals used by us is small. Our implementation takes little time to run and hence can be used efficiently for practical purposes.

Index Terms—Definite Integral, Firefly Algorithm, Solution Vector, Fitness Function.

I. INTRODUCTION

Definite Integration is the one of the most important tools that is required in most of the fields like physics, chemistry and other scientific domains. Exact analytical calculation of definite integrals is often impossible. Many of the functions like \( y = e^{\sin(\log(x^2))} \) [4] do not have primitives that can be expressed analytically. By symbolic integration, it is much more difficult to find out the exact value of the given integration, there is no suitable algorithm to solve this problem.

Traditionally, many algorithms are used to calculate definite integral using classical numerical method. However, in most of the cases the integrand \( f(x) \) is replaced by a different function \( g(x) \) which is easier to integrate and that is why results are not exact compared to analytical calculations.

In this paper, we have proposed a new method for performing numerical integration, which does not require the knowledge of the primitive of the integrand as it is based on the Firefly Algorithm.

This is a novel method for performing numerical integration and it also expands the area of applicability of Firefly Algorithm, which is generally used for solving optimization problems.

II. BACKGROUND OF THE PROBLEM

Let \( f(x) \) be a continuous function on which we want to calculate the definite integral \( I \), over the interval \([a, b]\).

\[
I(f) = \int_a^b f(x) \, dx
\]  

(1)

Let us divide the interval \([a, b]\) into \( k \) number of subintervals each of width \( h = (b - a)/k \). Let the intervals be \([x_0, x_1], [x_1, x_2], \ldots, [x_{k-1}, x_k]\) with \( x_r - x_{r-1} = h \) where \( 1 \leq r \leq k \). According to the definition of the definite integral we have

\[
\int_a^b f(x) \, dx = \sum_{r=1}^{k} f(x_r) \, h
\]  

(2)

Let \( F(x) = f(x) \). Then the Mean Value Theorem (MVT) states that there exists \( c \) in \([a, b]\) such that

\[
\frac{F(b) - F(a)}{b - a} = F'(c)
\]

which implies

\[
\int_a^b f(x) \, dx = f(c)(b - a), \quad \text{for some } c \text{ with } a \leq c \leq b
\]  

(3)

This is known as the Mean Value Theorem for integrals. The point \( f(c) \) is then called the average value of \( f(x) \) on \([a, b]\).

Let \( p_i \) be any point in the interval \([x_{i-1}, x_i]\) for each \( i \), then we define the following integral sum \( S(p_1, p_2, \ldots, p_k) \) [4].

\[
S(p_1, p_2, \ldots, p_k) = \sum_{i=1}^{k} f(p_i) \, (x_i - x_{i-1})
\]  

(4)

III. GEOMETRIC INTERPRETATION

The integral sum \( S(p_1, p_2, \ldots, p_k) \) represents the area bounded by the curve \( y = f(x) \). Note that we have chosen a set of \( k \) random points taking one from each interval. However it is approximately equal to the value of the
definite integral given by equation (1), and the calculation error does not exceed the error given by the rectangular method. If another set of points \( p_i \) is selected the integral sum will be different but the approximate value will be still equal to the integral’s value.

![Fig. 1. Geometrical representation of the integral sum \( S \) [4]](image)

For large value of \( N \) the average value of the integral sum \( S' \) will converge to the value of the integral (1). Thus when \( N \) tends to infinity [4],

\[
\int_a^b f(x)dx = \lim_{n \to \infty} S' = \lim_{n \to \infty} \frac{\sum_{i=1}^N S_i}{N} \quad (5)
\]

The integral sum \( S \) is close to the average sum \( S' \) whereas \( S \) is very close to the exact value of the integral given by equation (1). So we have reduced the problem of calculation of integral with reasonable accuracy to an optimization problem of finding some random points \( p_1, p_2, \ldots, p_k \) over the interval \([x_{r-1}, x_r]\) for each \( r \).

IV. DESCRIPTION OF THE FIREFLY ALGORITHM

The Firefly Algorithm [1][6] was introduced by Dr. Xin She yang at Cambridge University in 2007, which was inspired by the mating or flashing behavior of fireflies. There are about two thousand fireflies produce short \& rhythmic flashes. The pattern of flashes is often unique for a particular species. The flashing light produced by a process of bioluminescence and the true functions of such signaling systems are still being debated [6]. Although the algorithm has many similarities with other swarm based algorithms such as Particle Swarm Optimization [2][8] or Artificial Bee Colony Optimization [9], the Firefly Algorithm has proved to be much simpler both in concept and implementation and hence has been accepted by the community as a worthwhile optimization technique.

A. Concept

In Firefly Algorithm, there are three idealized rules:

1. A firefly will be attracted by to all other fireflies regardless of their sex.
2. Attractiveness is proportional to their brightness and brightness decreases as the distance among them increases.
3. If there are no fireflies brighter than a given firefly, it will move randomly.

The brightness should be associated with the objective function. In Proposed Algorithm, we state below our algorithm to find the value of a Definite Integral (Single).

For a maximization problem, the brightness can simply be proportional to the value of the objective function. Other forms of brightness can be defined in a similar way to the fitness function in genetic algorithms [4].

B. Fitness Function

For definition of fitness function, we will calculate the integral sum of each solution vector using (4). In an optimization problem, we want to minimize the difference between the integrated sum and the value of the integral (1). If the value of the integral sum \( S \) is close to the average value \( S' \) of all integral-rated sums, then the objective function is given by equation (5) [4].

So our fitness function \( F \) will be

\[
F(p_1, p_2, \ldots, p_k) = |S(p_1, p_2, \ldots, p_k) - S'|
\quad (6)
\]

Where \( S' \) is the average sum (depends on number of solution vectors, that is number of fireflies) of all the integrated sums (depends on each solution vector).The algorithm stops when the leading firefly does not change during several moves.

**Proposed Algorithm:**

**Solving \( \int_a^b f(x)dx \) using Firefly optimization**

Input: \( y = f(x) \), Lower limit, Upper limit, Number of Intervals

Define the objective function

\[
I = F(p_1, p_2, \ldots, p_k) = |S(p_1, p_2, \ldots, p_k) - S'|
\]

Compute width of each interval.

Initialize the \( N \) solution vectors and each vector has \( k \) random points between each interval.

Set all required parameters such as \( \alpha, \gamma, \delta \)

**while** (\( t \leq \text{MaxGeneration} \)) do

\[
i = 1
\]

\[
\text{while} \ (i \leq N) \ do
\]

\[
j = 1
\]

\[
\text{while} \ (j \leq N) \ do
\]

Calculate the distance between two fireflies.

\[
\text{if} \ (|l_j|) \ \text{then}
\]

Calculate the variation of the distance.

Update the solution vector.

**end if**

**end while**

**end while**

Evaluate new solutions and update light intensity.

Rank the fireflies and find the current best.

**end while**

Calculate the definite integral value for the best solution vector.
C. Initial Solution Vector
The solution is the ordered set of k points within interval [a, b] and the length of the each interval \( h = \frac{b-a}{k} \). It is important that there is only one point \( p_i \) lying in the interval \( [x_i, x_{i+1}] \) for each \( i \) where \( 1 \leq i \leq k \).

Fig. 2. Initial solution vector
So each solution vector is a sequence of \( k \) numbers, such as \( p_1, p_2, ..., p_k \).

D. Formulation of the Light Intensity and Attractiveness
We formulate the light intensity \( I \) as the objective function \( F \). In our case, we assume \( I = F \), i.e. equation (6).

In the Firefly Algorithm, there are two important issues including variation of light intensity and the formulation of the attractiveness [5]. We know, the light intensity varies according to the inverse square law, i.e.

\[ I(r) = \frac{I_0}{r^2} \]  
where \( I(r) \) is the light intensity at a distance \( r \) and \( I_0 \) is the intensity at the source. To avoid the singularity at \( r = 0 \) in equation (7) [5], the equations can be approximated in the following Gaussian form:

\[ I(r) = I_0 e^{-\gamma r^2} \]  
As we know, that the attractiveness of a firefly is proportional to the light intensity seen by adjacent fireflies, the attractiveness \( \beta \) of a firefly is determined by equation (9) where \( \beta_0 \) is the attractiveness at \( r = 0 \) and \( \gamma \) is the light absorption coefficient at the source [5].

\[ \beta = \beta_0 e^{-\gamma r^2}, \quad r \geq 1 \]  
E. Distance
The Cartesian distance between any two fireflies \( i \) and \( j \) present at \( x_i \) and \( x_j \) respectively is determined by equation (10) [5].

\[ r_{ij} = \sqrt{\sum_{k=1}^{N}(x_{ik} - x_{jk})^2} \]  
where \( N \) is the number of dimensions or number of solution vectors.

F. Movement
The movement of a firefly \( i \) that is attracted to another more attractive firefly \( j \) is determined by [5]

\[ x_{i+1} = x_i + \beta_0 e^{-\gamma r^2}(x_j - x_i) + \alpha (\text{rand} - \frac{1}{2}) \]  
where the second term is for attraction, while the third term is for randomization including randomization parameter \( \alpha \) and \( \text{rand} \) is the random number generator, whose generated numbers are uniformly distributed in the interval \([0, 1]\).

G. Convergence
For any large number of fireflies \( N \), if \( N > M \), where \( M \) is the number of local optima of an optimization problem, the convergence of the algorithm can be achieved. Here, the initial location of \( N \) fireflies is distributed uniformly in the entire search space [1][6]. As the iterations of the algorithm continue, fireflies converge into all the local optima. By comparing the best solutions among all these optima, the global optimum can be achieved.

V. TESTING OF OUR PROPOSED ALGORITHM
The quality of result of a proposed algorithm depends on the number of points in a partition of a segment and also the parameters of the Firefly Algorithm, such as the number of fireflies, different other parameter values and the stopping criteria. The relative error of any result is calculated by the formula:

\[ \Delta = \frac{(E_A - E_T)}{E_T} \times 100 \]  
where \( E_A \) is the value received from our algorithm and \( E_T \) is the value received from some classical numerical method. During the testing only one parameter of the Firefly Algorithm is changed. It is the number of points in a partition of a segment, all other parameters remain fixed.

As a typical example, we consider an integral with a monotonic continuous integrand such as \( \int_{2}^{3} x^3 \, dx \). It is important to note that the result (correct up to 4 decimal places) given by our algorithm is much closer than the ones given by known numerical algorithms such as top-right corner approximation, top-left corner approximation, midpoint approximation, trapezoidal rule, simpson’s 1/3 rule, especially when the number of intervals is small.

(a) Area bounded by the curve \( y = x^3 \) from \( x = 2.3 \) to \( 3 \).
(b) Graph representing percentage of error given by proposed algorithm and other numerical methods.
In case of quickly oscillating functions our Firefly Algorithm gives more exact result than the other known classical numerical integration methods. For example, we consider \( \int_{0.01}^{1} \cos \left( \frac{1}{x^2} \right) \, dx = -0.0925 \)

We briefly present the results of our experiments in Table 1 and before also show the comparison with an implementation using Genetic Algorithm (GA). Our proposed algorithm performs consistently better both in terms of accuracy as well as running time. In Table 2, we show a few cases with Double Integrals, where also the results are encouraging.

<table>
<thead>
<tr>
<th>Integration</th>
<th>Approx. Value</th>
<th>Firefly Algorithm</th>
<th>Genetic Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Error</td>
<td>Runtime (in sec)</td>
<td>Error</td>
</tr>
<tr>
<td>( \int_{0}^{1} x^3 , dx )</td>
<td>0.2496</td>
<td>0.0004</td>
<td>0.89</td>
</tr>
<tr>
<td>( \int_{0}^{0.6} \sin \left( \frac{1}{x} \right) , dx )</td>
<td>0.1277</td>
<td>0.0000</td>
<td>0.08</td>
</tr>
<tr>
<td>( \int_{0}^{1} e^{-x^2} , dx )</td>
<td>0.7068</td>
<td>0.0000</td>
<td>0.14</td>
</tr>
<tr>
<td>( \int_{0}^{1} \sin(x) + \cos(x^2) , dx )</td>
<td>2.5658</td>
<td>0.0001</td>
<td>0.06</td>
</tr>
<tr>
<td>( \int_{0}^{1} \sqrt{\sin(x)} , dx )</td>
<td>2.3962</td>
<td>0.0001</td>
<td>0.64</td>
</tr>
</tbody>
</table>

Table 1: Comparison of results for Single Integrals

<table>
<thead>
<tr>
<th>Integration</th>
<th>Approx. Value</th>
<th>Firefly Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Error</td>
<td>Runtime (in sec)</td>
</tr>
<tr>
<td>( \int_{0}^{1} \int_{0}^{1-x} xy , dy , dx )</td>
<td>0.0417</td>
<td>0.0000</td>
</tr>
<tr>
<td>( \int_{0}^{2} \int_{0}^{1} (x + y^3) , dy , dx )</td>
<td>17.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>( \int_{0}^{2} \int_{0}^{1} (x^2 + y^2) , dy , dx )</td>
<td>2.1394</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

Table 2: Results on Double Integrals

VI. CONCLUSION

In this paper, we have successfully used the Firefly Algorithm to calculate Definite Integrals. For brevity, we did not show the comparisons with the results obtained by classical numerical methods or other soft computing techniques [4].

However, it was observed that our implementation using Firefly Algorithm gave more accurate results compared to all of them.

Although, the Firefly Algorithm has advantages of being easily implementable, precise, robust, and accommodates parallel implementation, it also has
disadvantages like chances of being trapped into local optima and lack of memorization capability.

REFERENCES


