

Available online at <http://www.mecs-press.net/ijeme>

An Improved Flower Pollination Algorithm with Chaos

*Osama Abdel-Raouf ^a, Mohamed Abdel-Baset ^b and Ibrahim El-henawy ^c

^aDepartment of Operations Research, faculty of Computers and Information, Menoufia University, Menoufia, Shebin-el-Kome, Egypt, Postal code: 32511.

^bDepartment of Operations Research, faculty of Computers and Informatics, Zagazig University, El-ZeraSquare, Zagazig, Sharqiyah, Egypt, Postal code: 44519.

^cDepartment of Computer Science, faculty of Computers and Informatics, Zagazig University, El-ZeraSquare, Zagazig, Sharqiyah, Egypt, Postal code: 44519

Abstract

Flower pollination algorithm is a new nature-inspired algorithm, based on the characteristics of flowering plants. In this paper, a new method is developed based on the flower pollination algorithm combined with chaos theory (IFPCH) to solve definite integral. The definite integral has wide ranging applications in operation research, computer science, mathematics, mechanics, physics, and civil and mechanical engineering. Definite integral has always been useful in biostatistics to evaluate distribution functions and other quantities. Numerical simulation results show that the algorithm offers an effective way to calculate numerical value of definite integrals, and it has a high convergence rate, high accuracy and robustness.

Index Terms: Flower pollination algorithm; definite integral; optimization; nature-inspired algorithm.

© 2014 Published by MECS Publisher. Selection and/or peer review under responsibility of the Research Association of Modern Education and Computer Science.

1. Introduction

The definite integral is the study of how the numeric value of an integral can be found. Which refers to finding a square whose area is the same as the area under a curve, it is one of the classical topics of numerical analysis [1]. The basic problem considered by numerical integration is to compute an approximate solution to a definite integral $\int_a^b f(x)dx$. Situations arise which the analytical method we have developed so far cannot be used to evaluate a definite integral for example an integrand may not have an obvious antiderivative such as $\cos x^2$ and $\frac{1}{\ln x}$ or maybe the integrand represented by individual data points, which makes finding an antiderivative impossible. When analytical methods fail, we often turn to numerical methods [2], which are typically done on

* Corresponding author
E-mail address:

a calculator or computer.

These methods do not produce exact values of definite integrals, but provide approximations that are generally quite accurate. Some of the more advanced methods for software is widely available are rectangle rule method, trapezoidal rule method, Simpson's rule method, Newton-Cotes method, Romberg method, Gauss method and so on [3-6]. However, these traditional methods have limitations: the rectangle rule method, trapezoidal rule method, Simpson's rule method are suitable for the bad smooth integrand, but their precisions are low, Newton-Cotes method is one of the constructing integrand based on the interpolating functions, but the convergence is not guaranteed for higher order Newton-Cotes method, for the Romberg method and Gauss method, their convergent speeds are quick and the computational precisions are high, but their computations are complex.

Flower pollination is an intriguing process in the natural world. Its evolutionary characteristics can be used to design new optimization algorithms. The algorithm obtained good results were dealing with lower-dimensional optimization problems, but may become problematic for higher-dimensional problems because of its tendency to converge very fast initially. This paper introduced an improved Flower pollination algorithm by integrating it with chaos to improve the reliability of the global optimality, and also enhances the quality of the results.

This paper is organized as follows: after introduction, the original Flower pollination algorithm is briefly introduced. Then in section 3 introduces the meaning of chaos. In section 4, the proposed algorithm is described, while the results are discussed in section 5. Finally, conclusions are presented in section 6.

2. The Flower pollination Algorithm

Flower Pollination Algorithm (FPA) was founded by Yang in the year 2012. Inspired by the flow pollination process of flowering plants are the following rules [7]:

- Rule 1.* Biotic and cross-pollination can be considered as a process of global pollination process, and pollen-carrying pollinators move in a way that obeys Le'vy flights.
- Rule 2.* For local pollination, a biotic and self-pollination are used.
- Rule 3.* Pollinators such as insects can develop flower constancy, which is equivalent to a reproduction probability that is proportional to the similarity of two flowers involved.
- Rule 4.* The interaction or switching of local pollination and global pollination can be controlled by a switch probability $p \in [0,1]$, with a slight bias toward local pollination .

In order to formulate updating formulas, we have to convert the aforementioned rules into updating equations. For example, in the global pollination step, flower pollen gametes are carried by pollinators such as insects, and pollen can travel over a long distance because insects can often fly and move in a much longer range [7]. Therefore, Rule 1 and flower constancy can be represented mathematically as:

$$x_i^{t+1} = x_i^t + \gamma L(\lambda)(x_i^t - B) \quad (1)$$

Where x_i^t is the pollen i or solution vector x_i at iteration t , and B is the current best solution found among all solutions at the current generation/iteration. Here γ is a scaling factor to control the step size. In addition, $L(\lambda)$ is the parameter that corresponds to the strength of the pollination, which essentially is also the step size. Since insects may move over a long distance with various distance steps, we can use a Le'vy flight to imitate this characteristic efficiently. That is, we draw $L > 0$ from a Levy distribution:

$$L \sim \frac{\lambda \Gamma(\lambda) \sin(\pi\lambda/2)}{\pi} \frac{1}{S^{1+\lambda}}, (S \gg S_0 > 0) \quad (2)$$

Here, $\Gamma(\lambda)$ is the standard gamma function, and this distribution is valid for large steps $s > 0$. Then, to model the local pollination, both Rule 2 and Rule 3 can be represented as

$$x_i^{t+1} = x_i^t + U(x_j^t - x_k^t) \quad (3)$$

Where x_j^t and x_k^t are pollen from different flowers of the same plant species. This essentially imitates the flower constancy in a limited neighbourhood. Mathematically, if x_j^t and x_k^t comes from the same species or selected from the same population, this equivalently becomes a local random walk if we draw U from a uniform distribution in $[0, 1]$. Though Flower pollination activities can occur at all scales, both local and global, adjacent flower patches or flowers in the not-so-far-away neighbourhood are more likely to be pollinated by local flower pollen than those faraway. In order to imitate this, we can effectively use the switch probability like in Rule 4 or the proximity probability p to switch between common global pollination to intensive local pollination. To begin with, we can use a naive value of $p = 0.5$ as an initially value. A preliminary parametric showed that $p = 0.8$ might work better for most applications [7]. The basic steps of FP can be summarized as the pseudo-code shown in Figure 1.

Flower pollination algorithm

Define Objective function $f(x)$, $x = (x_1, x_2, \dots, x_d)$
Initialize a population of n flowers/pollen gametes with random solutions
Find the best solution \mathbf{B} in the initial population
Define a switch probability $p \in [0, 1]$
Define a stopping criterion (either a fixed number of generations/iterations or accuracy)
 while ($t < \text{MaxGeneration}$)
 for $i = 1 : n$ (all n flowers in the population)
 if $\text{rand} < p$,
Draw a (d -dimensional) step vector L which obeys a L évy distribution
Global pollination via $x_i^{t+1} = x_i^t + L(\mathbf{B} - x_i^t)$
 else
Draw U from a uniform distribution in $[0, 1]$
Do local pollination via $x_i^{t+1} = x_i^t + U(x_j^t - x_k^t)$
 end if
Evaluate new solutions
If new solutions are better, update them in the population
 end for
Find the current best solution \mathbf{B}
 end while
Output the best solution found

Fig. 1 Pseudo code of the Flower pollination algorithm

3. Chaos Theory

Generating random sequences with longer periods and good consistency is very important for easily

simulating complex phenomena, sampling, numerical analysis, decision making and especially in heuristic optimization [8]. Its quality determines the reduction of storage and computation time to achieve a desired accuracy [9]. Chaos is a deterministic, random-like process found in a nonlinear, dynamical system, which is non-period, non-converging and non-bounded. Moreover, it depends on its initial condition and parameters [10-17]. Applications of chaos has several disciplines including operations research, physics, engineering, economics, biology, philosophy and computer science [18-25].

Recently chaos has been extended to various optimization areas because it can more easily escape from local minima and improve global convergence in comparison with other stochastic optimization algorithms [26]. Using chaotic sequences in flower pollination Algorithm can be helpful to improve the reliability of the global optimality, and they also enhance the quality of the results.

3.1 Chaotic maps

At random-based optimization algorithms, the methods using chaotic variables instead of random variables are called chaotic optimization algorithms (COA) [20-24]. In these algorithms, due to the non-repetition and ergodicity of chaos, it can carry out overall searches at higher speeds than stochastic searches that depend on probabilities [20-26]. To resolve this issue, herein one-dimensional and non-invertible maps are utilized to generate chaotic sets. We will illustrate some of well-known one-dimensional maps as:

3.1.1 Logistic map

The Logistic map is defined by:

$$Y_{n+1} = \mu Y_n(1 - Y_n) \quad Y \in (0,1) \quad 0 < \mu \leq 4 \quad (4)$$

3.1.2 The Sine map

The Sine map is written as the following equation:

$$Y_{n+1} = \frac{\mu}{4} \sin(\pi Y_n) \quad Y \in (0,1) \quad 0 < \mu \leq 4 \quad (5)$$

3.1.3 Iterative chaotic map

The iterative chaotic map with infinite collapses is described as:

$$Y_{n+1} = \sin\left(\frac{\mu\pi}{Y_n}\right) \quad \mu \in (0,1) \quad (6)$$

3.1.4 Circle map

The Circle map is expressed as:

$$Y_{n+1} = Y_n + \alpha - \left(\frac{\beta}{2\pi}\right) \sin(2\pi Y_n) \quad \text{mod } 1 \quad (7)$$

3.1.5 Chebyshev map

The family of Chebyshev map is written as the following equation:

$$Y_{n+1} = \cos(k \cos^{-1}(Y_n)) \quad Y \in (-1,1) \quad (8)$$

3.1.6 Sinusoidal map

This map can be represented by

$$Y_{n+1} = \mu Y_n^2 \sin(\pi Y_n) \quad (9)$$

3.1.7 Gauss map

The Gauss map is represented by:

$$Y_{n+1} = \begin{cases} 0 & Y_n = 0 \\ \frac{\mu}{Y_n} \bmod 1 & Y_n \neq 0 \end{cases} \quad (10)$$

3.1.8 Sinus map

Sinus map is formulated as follows:

$$Y_{n+1} = 2.3(Y_n)^2 \sin(\pi Y_n) \quad (11)$$

3.1.9 Dyadic map

Also known as the dyadic map bit shift map, $2x \bmod 1$ map, Bernoulli map, doubling map or saw tooth map. Dyadic map can be formulated by a mod function:

$$Y_{n+1} = 2Y_n \bmod 1 \quad (12)$$

3.1.10 Singer map

Singer map can be written as:

$$Y_{n+1} = \mu(7.86Y_n - 23.31Y_n^2 + 28.75Y_n^3 - 13.3Y_n^4) \quad (13)$$

μ between 0.9 and 1.08

3.1.11 Tent map

This map can be defined by the following equation:

$$Y_{n+1} = \begin{cases} \mu Y_n & Y_n < 0.5 \\ \mu(1 - Y_n) & Y_n \geq 0.5 \end{cases} \quad (14)$$

4. The Proposed Algorithm (IFPCH) for Solving Definite integral

Suppose that segmentation S splits an integral interval $[a, b]$ into n -subintervals: $[x_0, x_1], [x_1, x_2], \dots, [x_{k-1}, x_k], [x_{n-1}, x_n]$, where $x_j < x_{j+1}$ for $j = 1, 2, \dots, n-1$; $x_0 = a$, and $x_n = b$, also define $\Delta x_k = x_k - x_{k-1}$ for $k = 1, 2, \dots, n$. Using this notation, the integral $f(x)$ in $[a, b]$ can be approximated as [27]:

$$\int_a^b f(x) \approx \sum_{k=1}^n \frac{1}{6} \left(f(x_{k-1}) + 4f\left(\frac{x_{k-1} + x_k}{2}\right) + f(x_k) \right) \Delta x_k \quad (15)$$

In the proposed chaotic Flower pollination algorithm, we used chaotic maps to tune the Flower pollination algorithm parameter and improve the performance [11]. The steps of the proposed chaotic Flower pollination

algorithm for solving definite integral are as follows:

- Step 1.** Define the objective function and initializes a population and find the best solution B in the initial population.
- Step 2.** Calculate p by the selected chaotic maps.
- Step 3.** If ($\text{rand} < p$) then global pollination via $x_i^{t+1} = x_i^t + (f\gamma)L(\lambda)(x_i^t - B) // (f\gamma)$ chaotic Levy flights else do local pollination via selected chaotic map.
- Step 4.** Evaluate new solutions if better, update them in the population.
- Step 5.** Find the current best solution B .
- Step 6.** Output the best solution found.

5. Numerical Results

Several examples have been conducted to verify the weight of the proposed algorithm. The initial parameters are set at $n = 50$ and the number of iterations is set to $t = 1000$. The results of IFPCH algorithm are conducted from 30 independent runs for each integrand. The selected chaotic map for all examples is the Sinusoidal map, whose equation is shown below:

$$Y_{n+1} = \mu Y_n^2 \quad (16)$$

Where n is the iteration number. All the experiments were performed on a Windows 7 Ultimate 64-bit operating system; processor Intel Core i5 760 running at 2.81 GHz; 4 GB of RAM and code was implemented in MATLAB.

Table 1. Integral values of selected functions

	F(x)	Monte Carlo Method	Midpoint Rule	Trapezoidal Rule	Simpson's Rule	IFPCH	Exact Value
F1	$x^2 e^{-x}, 0 \leq x \leq 2$	0.646649	0.646659	0.646633	0.646651	0.646647	0.646647
F2	$\sin^2 x, 0 \leq x \leq 2$	1.1892	1.19047	1.18667	1.1890	1.1892	1.1892
F3	$\frac{1}{1+x^2}, 0 \leq x \leq 1$	0.785396	0.786231	0.783732	0.785395	0.785398	0.785398
F4	$e^x, 1 \leq x \leq 2$	6.3892	6.3761	8.3891	6.4207	6.3891	6.3891
F5	$e^{-x^2}, 0 \leq x \leq 1$	0.74684	0.74678	0.74621	0.74683	0.74682	0.74682
F6	$\frac{\ln(x)}{1+x}, 1 \leq x \leq 2$	0.1576	0.148278735	0.14509553	0.147175616	0.14722067	0.14722067
F7	$\cos^4(x)\sin^5(x), 1 \leq x \leq 4$	0.0212	0.017374155	0.02089606	0.01438904	0.03711515	0.03711515
F8	$e^{3x}\sin(2x), 1 \leq x \leq 2$	2.6916	0.3111514	0.0891849	0.2418799	-35.3891295	-35.3891295
F9	$\frac{x^3}{\sqrt{x^5-2}}, 3 \leq x \leq 3.5$	1.2444	0.903734158	0.90370447	0.903724293	0.90372426	0.90372426
F10	$\frac{x^2 \ln(x)}{\sin^3(x)}, 1 \leq x \leq 2$	1.0184	1.168303150	1.24248120	1.196126966	1.192783710	1.192783710

Table 1 shows the results of IFPCH algorithm are privileged compared with the results of Standard Monte Carlo method, trapezoidal rule, Simpson's rule and midpoint rule. In comparison with exact values we find that the results of IFPCH algorithm very close to the exact values of selected functions under study. Furthermore at test problem number eight (F8), the proposed algorithm found the optimal solution better than all other

numerical integration methods. If a large number of well-behaved one-dimensional integrands are to be integrated, and the user is willing to do some analytic analysis to obtain efficiency, then it is hard to go past the classical methods. More usually, though, users will choose to use IFPCH algorithm, to save their own time and to gain reliability.

The reason for getting better results than the other algorithms considered is that the search power of Flower pollination algorithm. To add to this, using chaos helps the algorithms to escape from local solutions.

6. Conclusions

This paper introduced an improved Flower pollination algorithm by blending with chaos for calculation the numerical value of definite integrals. This algorithm has the ability to trounce the shortage that the segmentation points are uniform in traditional methods. Several simulation examples show that the algorithm can converge to the best solution, and it has a high convergence rate and high accuracy making it easier for engineers to use.

References

- [1] Rao, S. S. (2001). *Applied numerical methods for engineers and scientists*. Prentice Hall Professional Technical Reference.
- [2] Kreyszig, E. (2007). *Advanced engineering mathematics*. John Wiley & Sons.
- [3] Davis, P. J., & Rabinowitz, P. (2007). *Methods of numerical integration*. Courier Dover Publications.
- [4] Forsythe, G. E., Moler, C. B., & Malcolm, M. A. (1977). *Computer methods for mathematical computations*.
- [5] Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1992). *Numerical Recipes: The art of scientific computing* (Cambridge).
- [6] Hildebrand, F. B. (1987). *Introduction to numerical analysis*. Courier Dover Publications.
- [7] Abdel-Raouf, O., Abdel-Baset, M., & El-henawy, I. (2014). A New Hybrid Flower Pollination Algorithm for Solving Constrained Global Optimization Problems. *International Journal of Applied*, 4(2), 1-13.
- [8] Abdel-Raouf, O., El-henawy, I., & Abdel-Baset, M. (2014). Chaotic Harmony Search Algorithm with Different Chaotic Maps for Solving Assignment Problems. *International Journal of Computer Applications*, 86(10), 8-13.
- [9] Yang, D., Li, G., & Cheng, G. (2007). On the efficiency of chaos optimization algorithms for global optimization. *Chaos, Solitons & Fractals*, 34(4), 1366-1375.
- [10] Gandomi, A. H., Yun, G. J., Yang, X. S., & Talatahari, S. (2013). Chaos-enhanced accelerated particle swarm optimization. *Communications in Nonlinear Science and Numerical Simulation*, 18(2), 327-340.
- [11] Alatas, B. (2010). Chaotic harmony search algorithms. *Applied Mathematics and Computation*, 216(9), 2687-2699.
- [12] Gong, W., & Wang, S. (2009, December). Chaos ant colony optimization and application. In *Internet Computing for Science and Engineering (ICICSE), 2009 Fourth International Conference on* (pp. 301-303). IEEE.
- [13] Alatas, B. (2010). Chaotic bee colony algorithms for global numerical optimization. *Expert Systems with Applications*, 37(8), 5682-5687.
- [14] Gandomi, A. H., Yang, X. S., Talatahari, S., & Alavi, A. H. (2013). Firefly algorithm with chaos. *Communications in Nonlinear Science and Numerical Simulation*, 18(1), 89-98.
- [15] Mingjun, J., & Huanwen, T. (2004). Application of chaos in simulated annealing. *Chaos, Solitons & Fractals*, 21(4), 933-941.
- [16] Coelho, L. D. S., & Mariani, V. C. (2008). Use of chaotic sequences in a biologically inspired algorithm for engineering design optimization. *Expert Systems with Applications*, 34(3), 1905-1913.

- [17] Tavazoei, M. S., & Haeri, M. (2007). Comparison of different one-dimensional maps as chaotic search pattern in chaos optimization algorithms. *Applied Mathematics and Computation*, 187(2), 1076-1085.
- [18] Corron, N. J., Hayes, S. T., Pethel, S. D., & Blakely, J. N. (2006). Chaos without nonlinear dynamics. *Physical review letters*, 97(2), 024101.
- [19] He, D., He, C., Jiang, L. G., Zhu, H. W., & Hu, G. R. (2001). Chaotic characteristics of a one-dimensional iterative map with infinite collapses. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 48(7), 900-906.
- [20] Erramilli, A., Singh, R. P., & Pruthi, P. (1994). *Modeling packet traffic with chaotic maps*. KTH.
- [21] May, R. M. (1976). Simple mathematical models with very complicated dynamics. *Nature*, 261(5560), 459-467.
- [22] Wolf, A. (1986). Quantifying chaos with Lyapunov exponents. *Chaos*, 273-290.
- [23] Devaney, R. L. (1989). *An introduction to chaotic dynamical systems* (Vol. 6). Reading: Addison-Wesley.
- [24] Peitgen, H. O., Jürgens, H., & Saupe, D. (2004). *Chaos and fractals: new frontiers of science*. Springer.
- [25] Ott, E. (2002). *Chaos in dynamical systems*. Cambridge university press.
- [26] Letellier, C. (2013). *Chaos in nature* (Vol. 81). World Scientific.
- [27] Qu, L., & He, D. (2010). Solving Numerical Integration by Particle Swarm Optimization. In *Information Computing and Applications* (pp. 228-235). Springer Berlin Heidelberg.

Author(s) Profile

Ibrahim El-henawy received the M.S. and Ph.D. degrees in computer science from State University of New York, USA in 1980 and 1983, respectively. Currently, he is a professor in computer science and mathematics department, Zagazig University. Ibrahim is a member of IEEE and AAEE (Australasian Association for Engineering Education), He is also a reviewer in different international journals and conferences. He has published more than 200 refereed articles in National and international journals, edited books, and conference proceedings. His current research interests are mathematics, operations research, statistics, networks, optimization, Intelligent Computing, Computer Theory, digital image processing, and security

Osama Abdel-Raouf received the M.S. and Ph.D. degrees in operations research and decision support systems from Monofia University. Currently, he is an associate professor in operations research department, Monofia University. His current research interests are evolutionary algorithms, artificial intelligence, and decision support systems.

Mohamed Abd El-Baset received the B.Sc. degree in information system and technology from Zagazig University in 2006, and he obtained his M.S. degree in operations research and decision support systems from Zagazig University, in 2011. Currently, he is a teaching assistant in operations research department, Zagazig University. His current research interests are characterization of probability distribution, optimization, Intelligent Computing, Evolutionary Computation and decision support systems.

How to cite this paper: Osama Abdel-Raouf, Mohamed Abdel-Baset, Ibrahim El-henawy, "An Improved Flower Pollination Algorithm with Chaos", *IJEME*, vol.4, no.2, pp.1-8, 2014. DOI: 10.5815/ijeme.2014.02.01