

Improvement of Gregory's formula using Particle Swarm Optimization

N. Khelil, L. Djerou, A. Zerarka and M. Batouche

Abstract—Consider the Gregory integration (G) formula

$$\int_0^n f(x) dx = \sum_{k=0}^n f(k) + \sum_{k \geq 0} \left(\frac{a_k}{k!} \right) (\Delta_{h_k}^k f(0) + \Delta_{-h_k}^k f(n)) \quad (1)$$

with end corrections where Δ_h is the forward difference operator with step size h . In this study we prove that (1) can be optimized by minimizing some of the coefficient a_k in the remainder term by particle swarm optimization. Experimental tests prove that (1) can be rendered a powerful formula for library use.

Keywords—Numerical integration, Gregory Formula, Particle Swarm optimization.

I. INTRODUCTION

NOTE that for $h_k=1$ (1) reduces to the classical Gregory integration formula. To justify our formula, we shall use the umbral methods developed by Rota and his school [1]-[2] instead of classical generating function technique.

When $f(x)$ is replaced by $1, x, x^2, x^3 \dots$ we find.

$$\begin{aligned} a_0 &= -\frac{1}{2} \\ a_1(h_1) &= \frac{1}{12h_1} \\ a_2(h_1, h_2) &= \frac{-h_1}{24h_2} \\ a_3(h_1, \dots, h_3) &= \left(\frac{1}{720h_3^3} \right) (-10h_1^2 + 30h_2h_1 - 1) \\ a_4(h_1, \dots, h_4) &= \left(\frac{-1}{480h_4^4} \right) \left(\frac{5}{3}h_1^3 - 10h_3h_1^2 + 30h_3h_2h_1 - h_3 \right) \\ a_5(h_1, \dots, h_5) &= \left(\frac{1}{60480h_5^5} \right) (-42h_1^4 + 630h_2^3h_1 + 1050h_3^2h_2^2 - 3150h_3^2h_2h_1 \\ &\quad - 420h_4h_1^3 - 2520h_4h_3^3h_1 - 2940h_4h_2^2h_1 + 7560h_4h_3h_2h_1 + 105h_5^2 - 252h_4h_3 + 2) \end{aligned}$$

So, at the order 5 we can write:

Dr N.Khelil is with the mathematics department, university Med khider at Biskra, Algeria. (Corresponding author to provide phone +213 33 742186; e-mail: khelilna@yahoo.fr).

LDjerou is PhD student at Labo of Physics and Applied Mathematics, University Med Khider at Biskra, Algeria (e-mail: ldjerou@yahoo.fr).

Pr. A. Zerarka is with the Physics department, University Med Khider at Biskra, Algeria, (e-mail: azerarka@hotmail.com).

Pr. M. Batouche computer Science Departement CCIS-King Saud University, Riyadh Saudi Arabia, (e-mail: mbatouche@ccis.ksu.edu.sa).

$$\begin{aligned} \int_0^n f(x) dx &\approx \sum_{k=0}^n f(k) + a_0[f(0) + f(n)] + a_1(h_1)[f(h_1) - f(0) + f(n-h_1) - f(n)] \\ &\quad + a_2(h_1, h_2)[f(2h_2) - 2f(h_2) + f(0) + f(n-2h_2) - 2f(n-h_2) + f(n)] \\ &\quad + a_3(h_1, h_2, h_3)[\Delta_{h_3}^3 f(0) + \Delta_{-h_3}^3 f(n)] \\ &\quad + a_4(h_1, h_2, h_3, h_4)[\Delta_{h_4}^4 f(0) + \Delta_{-h_4}^4 f(n)] \\ &\quad + a_5(h_1, h_2, h_3, h_4, h_5)[\Delta_{h_5}^5 f(0) + \Delta_{-h_5}^5 f(n)] \end{aligned} \quad (2)$$

This formula has a sense so $n \geq 2$. In the contrary case an appropriate variable change will permit us to do the integral without no difficulty.

Our goal is to optimize the remainder. For do it, we try to determine h_1, h_2, h_3, h_4 and h_5 that returns a_3, a_4 and a_5 as small as possible. a_3, a_4 and a_5 is a system non linear of 3 equations to 5 unknowns h_1, h_2, h_3, h_4 and h_5 ; we take $h_4, h_5 (=1, \text{ in this study})$ as parameters and let's solve this system by PSO.

II. PARTICLE SWARM OPTIMIZATION

Overview and strategy of particle swarm optimization

Recently, a new stochastic algorithm has appeared, namely 'particle swarm optimization' PSO. The term 'particle' means any natural agent that describes the 'swarm' behavior. The PSO model is a particle simulation concept, and was first proposed by Eberhart and Kennedy [3]. Based upon a mathematical description of the social behavior of swarms, it has been shown that this algorithm can be efficiently generated to find good solutions to a certain number of complicated situations such as, for instance, the static optimization problems, the topological optimization and others [4]-[5]-[6]-[7]. Since then, several variants of the PSO have been developed [8]-[9]-[10]-[11]-[12]-[13]-[14]. It has been shown that the question of convergence of the PSO algorithm is implicitly guaranteed if the parameters are adequately selected [15]-[16]-[17]. Several kinds of problems solving start with computer simulations in order to find and analyze the solutions which do not exist analytically or specifically have been proven to be theoretically intractable.

The particle swarm treatment supposes a population of individuals designed as real valued vectors - particles, and some iterative sequences of their domain of adaptation must be established. It is assumed that these individuals have a social behavior, which implies that the ability of social conditions, for instance, the interaction with the neighborhood, is an important process in successfully finding good solutions to a given problem.

The strategy of the PSO algorithm is summarized as follows: We assume that each agent (particle) i can be represented in a N dimension space by its current position $x_i = (x_{i1}, x_{i2}, \dots, x_{iN})$ and its corresponding velocity. Also a memory of its personal (previous) best position is represented by, $p = (p_{i1}, p_{i2}, \dots, p_{iN})$ called (pbest), the subscript i range from 1 to s , where s indicates the size of the swarm. Commonly, each particle localizes its best value so far (pbest) and its position and consequently identifies its best value in the group (swarm), called also (sbest) among the set of values (pbest).

The velocity and position are updated as.

$$v_{ij}^{k+1} = w_j v_{ij}^k + c_1 r_1^k [(pbest)_{ij}^k - x_{ij}^k] + c_2 r_2^k [(sbest)_{ij}^k - x_{ij}^k] \quad (3)$$

$$x_{ij}^{k+1} = v_{ij}^{k+1} + x_{ij}^k \quad (4)$$

where are the position and the velocity vector of particle i

respectively at iteration $k + 1$, c_1 et c_2 are acceleration coefficients for each term exclusively situated in the range of

2--4, w_{ij} is the inertia weight with its value that ranges from

0.9 to 1.2, whereas r_1 , r_2 are uniform random numbers between zero and one. For more details, the double subscript in the relations (2) and (3) means that the first subscript is for the particle i and the second one is for the dimension j . The

role of a suitable choice of the inertia weight w_{ij} is important in the success of the PSO. In the general case, it can be initially set equal to its maximum value, and progressively we decrease it if the better solution is not reached. Too often, in the relation (1), w_{ij} is replaced by w_{ij} / σ , where σ denotes the constriction factor that controls the velocity of the particles. This algorithm is successively accomplished with the following steps [18]-[19]:

1. Set the values of the dimension space N and the size s of the swarm (s can be taken randomly).
2. Initialize the iteration number k (in the general case is set equal to zero).
3. Evaluate for each agent, the velocity vector using its memory and equation (3), where pbest and sbest can be modified.
4. Each agent must be updated by applying its velocity vector and its previous position using equation (4).
5. Repeat the above step (3, 4 and 5) until a convergence criterion is reached.

The practical part of using PSO procedure will be examined in the following section, where we'll optimize a_3 , a_4 and a_5 using PSO.

The PSO algorithm is applied, with parameter setting (Table I).

III. SIMULATION STUDY

To test the performance of this algorithm we took various

TABLE I
PARTICLE SWARM PARAMETER SETTING

Parameter	Setting
Population Size	20
Number of Iterations	500
c_1 and c_2	0.5
Inertial Weight	1.2 \rightarrow 0.4
Desired Accuracy	10^{-5}

functions and we looked for an approximation with Gregory (G) and the new formula (GP) (Table II).

IV. CONCLUSION

A Considering results we note that:

- the middle value turns around 0.3
- some well-known functions badly to integrate by the methods classic ($1/1+x, \dots$) give good results with this formula .
- in general the GP formula improves the order of precision in a considerable way in relation to Gregory, without using as many assessments.

TABLE II
COMPARISON

Function	Interval	Formula	h_1	h_2	h_3	R. Error.
$1/1+x$	[0, 1]	G	1	1	1	0.063
		GP	0.3	0.3	0.3	-2.4732×10^{-4}
e^x	[0, 2]	G	1	1	1	0.0102
		GP	0.3	0.3	0.3	-7.9663×10^{-5}
$x(1-x)$	[0, 1]	G	1	1	1	0
		GP	0.3	0.3	0.3	0
$x/1+x^2$	[0, 2]	G	1	1	1	-0.0652
		GP	0.3	0.3	0.3	7.1755×10^{-4}

ACKNOWLEDGMENT

This work was sponsored, in part, by the MERS (Ministère de l'Enseignement et de la recherche Scientifique): Under contrat N° B*0142/0/0016.

REFERENCES

- [1] G.C.ROTA, *finite operator calculus*, Academic press, Inc., 1975
- [2] M.K.Belbahri, Generalized Gregory formula, Doctoral Thesis, Stevens Institute of Technology (1982).
- [3] Eberhart, R.C. and Kennedy, J. (1995). A new optimizer using particles swarm theory', *Sixth International Symposium on Micro Machine and Human Science*, pp.39-43, Nagoya, Japan.
- [4] K. E. Parsopoulos and M. N. Vrahatis, Modification of the Particle Swarm Optimizer for Locating all the Global Minima, V. Kurkova et al., eds., *Artificial Neural Networks and Genetic Algorithms*, Springer, New York, (2001), pp. 324-327.

- [5] K. E. Parsopoulos et al., Stretching technique for obtaining global minimizers through particle swarm optimization, Proc. of the PSO Workshop, Indianapolis, USA, (2001b), pp. 22-29.
- [6] P.C. Fourie, and Groenwold, A.A. Particle swarms in size and shape optimization', *Proceedings of the International Workshop on Multidisciplinary Design Optimization*, Pretoria, South Africa, August 7–10, (2000), pp.97–106.
- [7] P.C. Fourie, and Groenwold, A.A. Particle swarms in topology optimization', *Extended Abstracts of the Fourth World Congress of Structural and Multidisciplinary Optimization*, Dalian, China, June 4–8, (2001), pp.52, 53.
- [8] R.C. Eberhart, et al. Computational Intelligence PC Tools, *Academic Press Professional*, Boston. 1996.
- [9] J.J. Kennedy, The behaviour of particles, *Evol. Progr.* VII (1998), 581-587.
- [10] J. Kennedy and R. C. Eberhart, *Swarm Intelligence*, Morgan Kaufmann Publishers, San Francisco, 2001.
- [11] Y. H. Shi and R. C. Eberhart, Fuzzy adaptive particle swarm optimization, *IEEE Int. Conf. on Evolutionary Computation*, pp. 101-106, (2001).
- [12] Y. H. Shi and R. C. Eberhart, A modified particle swarm optimizer, *Proc. of the 1998 IEEE International Conference on Evolutionary Computation*, Anchorage, Alaska, May 4-9, 1998a.
- [13] Y. H. Shi and R. C. Eberhart, Parameter selection in particle swarm optimization, *Evolutionary Programming VII, Lecture Notes in Computer Science*, (1998b), pp. 591-600.
- [14] M. Clerc, The swarm and the queen: towards a deterministic and adaptive particle swarm optimization, *Proceedings of the 1999 IEEE Congress on Evolutionary Computation, Washington DC (1999)*, pp.1951--1957.
- [15] R.C. Eberhart, and Shi, Y.. Parameter selection in particle swarm optimization, in Porto, V.W., 1998
- [16] T. I. Cristian, The particle swarm optimization algorithm: convergence analysis and parameter selection, *Information Processing Letters*, Vol. 85, No. 6, (2003), pp.317-- 325.
- [17] K. E. Parsopoulos et al., Objective function stretching to alleviate convergence to local minima, *Nonlinear Analysis TMA* 47 (2001a), 3419-3424.
- [18] A. Zerarka and N. Khelil, A generalized integral quadratic method: improvement of the solution for one dimensional Volterra integral equation using particle swarm optimization, *Int. J. Simulation and Process Modelling* 2(1-2) (2006), 152-163.
- [19] L. Djerou, M. Batouche, N. Khelil and A. Zerarka. *Towards the Best Points of Interpolation Using Particles Swarm Optimisation Approach*, *IEEE Congress on Evolutionary Computation*. Singapore September 25- 28, 2007

N. Khelil is a Doctor at Department of Applied Mathematics, University Med Khider at Biskra, Algeria. He holds a Graduate Diploma, DES in Mathematics and Post Graduate Diploma: Magister. He obtained his Doctorat from the University Biskra. His domain of interest is: numerical analysis and applied mathematics.

L. Djerou is a PhD student at Department of Applied Mathematics, University Med Khider at Biskra, Algeria. She holds a Graduate Diploma, Engineer in Informtics and Post Graduate Diploma: Magister. Her domain of interest is: Emergent computing and Complex Systems, Image processing and computer vision, Metaheuristics and Nature Inspired Computing.

A. Zerarka is a Professor of Physics and Applied Mathematics at the University Med Khider at Biskra, Algeria. She obtained his PhD from the University Bordeaux, France. Her domain of interest is physics and applied mathematics. He has been teaching and conducting research since 1981 and has widely published in international journals and the reviewer of numerous papers. He is an active member of the Academy of Science, New York.

M. Batouche is a Professor at computer Science Departement CCIS-King Saud University, Riyadh Saudi Arabia.. He obtained his PhD from the National Polytechnic Institute of Lorraine (INPL), University of Nancy,