CHEMICAL ENGINEERING TRANSACTIONS
VOL. 51, 2016

A publication of AIDIC
The Italian Association of Chemical Engineering
Online at www.aidic.it/cet

Guest Editors: Tichun Wang, Hongyang Zhang, Lei Tian

ISBN 978-88-95608-43-3; **ISSN** 2283-9216
DOI: 10.3303/CET1651084

The Design and Application of Quantum-Behaved Particle Swarm Optimization Based on Levy Flight

Yaya Liu

Shangluo University
Lyy_yyl@163.com

Both Particle Swarm Optimization (PSO) and its improved version of Quantum-behaved Particle Swarm Optimization (QPSO) are the novel swarm intelligence optimization algorithms. However, the above two algorithm makes the search process easy to fall into local optimum and premature convergence because of the existence of particle waiting effect. For overcoming the shortcoming of QPSO and improving its search ability, considering that the feature of Levy flight is different that of Brown motion used by QPSO, we proposed a method of based on Levy flight quantum-behaved particle swarm optimization (LFQPSO). In order to test the performance of the proposed algorithm in our work, we apply it to the benchmark function test and compare it with standard PSO and QPSO algorithm, which shows that the LFQPSO outperforms the standard PSO and QPSO algorithm.

1. Introduction

Inspired by the movement of birds, Kennedy and Eherhart (1995) proposed the Particle Swarm Optimization (PSO) which is a kind of evolutionary computation technology based on swarm intelligence. PSO takes a habitat of bird swarm motion analogous to the space to be optimized, takes each bird to position of feasible solution, and thus guides the bird swarm to move toward to a better location or solution through the information transfer between individuals. For convenience, the bird or individual is abstracted as the particle which is not of the size and quality. PSO has the following features: The particle is moving in the form of a track which is determined by the position and speed of the particle at the same time, this is, when a particle moves at a certain speed, its trajectory is determined. So the search range of PSO is limited and not covers the total space of feasible solution, which makes it impossible that the global optimal solution can be searched. Considering that the quantum motion has great uncertainty and makes the range of motion enlarge since quantum motion obeys the two laws of wave-particle dualism and the uncertainty principle in the quantum phase-space. Based on the above feature, Sun et al. (2004) proposed the method of Quantum-behaved Particle Swarm Optimization (QPSO) in which its evolution equation is different that of PSO, and it uses the iterative equation with completely randomization which overcomes the shortcoming of PSO and is a novel swarm intelligence optimization algorithm.

However, both the PSO and the QPSO algorithm have particle waiting effect which makes the search process easy to fall into local optimum and premature convergence. Due to the appearance of the particle waiting effect, the result is that the individual is close to the better optimal solution according to the Brown motion. In fact, in nature the flight or foraging mode of the individual is not entirely random, but is subject to Levy flight (Yang and Deb, 2009). The so-called Levy flight is a random walk step size obeys Levy distribution, in detail, most of the individual flies or forages only in a small area, but a small part of the individual can suddenly fly far away, so this behavior is very conducive to the search process and used in the CS algorithm (Yang and Deb, 2010).

For improving the performance of QPSO, combined with the feature of Levy flight, we proposed a method of based on Levy flight quantum-behaved particle swarm optimization (LFQPSO), and its evolution equation is completely different that of QPSO and its convergence did not happen much change. We will first test it against a benchmark function. In order to test the performance of the proposed algorithm in our work, we apply it to the benchmark function test and compare it with standard PSO and QPSO algorithm.

Please cite this article as: Liu Y.Y., 2016, The design and application of quantum-behaved particle swarm optimization based on levy flight, Chemical Engineering Transactions, 51, 499-504 DOI:10.3303/CET1651084

2. Description of LFQPSO

2.1 QPSO

The basic principle of the QPSO algorithm can be described as: let the dimension of the solution space of an optimization problem is assumed to d, in this the solution space, there exists a group of particles $X = \{X_1, X_2, \cdots, X_m\}$ in which are composed of *M* particles, and each particle represents a potential solution to the optimization problem. Let at a certain moment t, the location of the ith particles is$X_i(t) = \left(X_{i,1}(t), X_{i,2}(t), \cdots, X_{i,d}(t)\right)$, the best location of the ith particles is $P_i(t) = \left(P_{i,1}(t), P_{i,2}(t), \cdots, P_{i,d}(t)\right)$ and the global best location of the X is $G(t) = \left(G_1(t), G_2(t), \cdots, G_d(t)\right) = P_g(t), g \in \{1,2,\cdots,m\}$ in which g is the subscript of global best location in the X. QPSO only considers the positions of the particle and does not consider the velocity of the particle, which is reason that it is easy to understand and solve.

For the minimization problem, the objective function value is smaller, the corresponding adaptation value is better. The best location $\mathrm{P_i}(t)$of the ith particlesPbest is determined as following:

$$P_i(t) = \begin{cases} X_i(t), if\ f(X_i(t)) < f(P_i(t-1))) \\ P_i(t-1), if\ f\left(X_i(t)\right) \geq f(P_i(t-1))) \end{cases} \tag{1}$$

The global best position of the group is determined by the following formula:

$$g = argmi\,n\left\{f\left(P_i(t)\right)\right\}, i \in \{1,2,\cdots,m \tag{2}$$

$$G(t) = P_g(t) \tag{3}$$

The particle position update equation of the QPSO algorithm is:

$$P_{i,j}(t) = \varphi_j(t) \cdot P_{i,j}(t) + \left(1 - \varphi_j(t)\right) \cdot G_j(t),\ and\ \varphi_j(t) \sim U(0, \tag{4}$$

$$X_{i,j}(t+1) = P_{i,j}(t) + \alpha \cdot \left|C_j(t) - X_{i,j}(t)\right| \cdot ln\left(\frac{1}{u_{i,j}(t)}\right) and\ u_{i,j}(t) \sim U(\tag{5}$$

$$C(t) = (C_1(t), C_2(t), \cdots, C_d(t) = {\sum_{i=1}^{m} P_i(t)}\big/_{m}. \tag{6}$$

Based on the above analysis, QPSO algorithm process is as follows:

Step 1: Let $t = 0$, initializing the particles $X(t) = \{X_1, X_2, \cdots, X_m\}$in the particle swarm in the problem space and use $P(t) = \{P_1, P_2, \cdots, P_m\}$ as the best particles of each particle which are initialized by $X(t)$. In addition, let $G(t)$ is the global best swarm and is initialized by computing the subscript g of the best swarm according to (2).

Step 2: Let $t = t + 1$, according to (4), (5) and (6), computing the fitness of $f(X(t))$ and for $i = 1$ to m, if $f\left(X_i(t+1)\right) < f(P_i(t))$ then $P_i(t) = X_i(t+1)$.

Step 3: computing the subscript g of the best swarm according to (2), if $f(P_g(t) < G(t)$ then $G(t) = P_g(t)$.

Step 4: updating the $P_i(t)$ and $X(t)$ according to (4) and (5).

Step 5: if no meeting the stop condition then go to step 2.

2.2 Levy flight

Animals foraging path was considering as a random or quasi-random manner in nature. However, various studies have shown that the flight behavior of many animals and insects obeys the typical characteristics of Levy flights. Broadly speaking, Levy flights are a random walk whose step length is drawn from the Levy distribution, often in terms of a simple power-law formula $L(s) \sim |s|^{-1-\beta}$ where $0 < \beta < 2$. Obviously, the generation of step sizes samples is not trivial using Levy flights. A simple scheme discussed in detail can be summarized as following (Yang, 2010 and Walton, 2011):

$$L(s) \sim \frac{u}{|v|^{\frac{1}{\beta}}} \tag{7}$$

in which $u \sim N(0, \sigma_u^2)$, $v \sim N(0, \delta_v)$ are normal distribution.

2.3 The description of LFQPSO

For further improving the performance of QPSO, we use levy flight to change the evolution equation of each particles. So the equation (7) is introduced in (4), (5) and (6), then the following evolution equation is following:

$$P_{i,j}(t) = \varphi_j(t) \cdot P_{i,j}(t) + \left(1 - \varphi_j(t)\right) \cdot G_j(t),\ and\ \varphi_j(t) \sim levy(\beta) \tag{8}$$

$$X_{i,j}(t+1) = P_{i,j}(t) + \alpha \cdot \left|C_j(t) - X_{i,j}(t)\right| \cdot ln\left(\frac{1}{u_{i,j}(t)}\right) and\ u_{i,j}(t) \sim levy(\beta) \tag{9}$$

In which $levy(\beta) \sim \frac{u}{|v|^{\frac{1}{\beta}}}(G(t) - P_i(t))$ and $u \sim N(0, \sigma_u^2)$, $v \sim N(0,1)$, $\sigma_u = \frac{\Gamma(1+\beta)sin(\beta\pi/2)}{\Gamma\left(\frac{1+\beta}{2}\right)\beta 2^{(\beta-1)/2}}^{1/\beta}$. Based on the above step, Figure 1 presents the flowchart of LFQPSO.

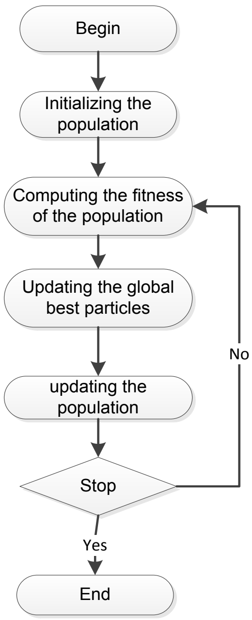

Figure 1: The flowchart of LFQPSO

3. Function Optimization

Function Optimization is often expressed in the following form:

$$\min f(X) \quad \text{subject to} \quad L \leq X \leq U \tag{10}$$

In which $X = \{x_1, x_2, x_3, \dots, x_n\}$ is a vector, and $L = \{l_1, l_2, l_3, \dots, l_n\}$is the lower bound of X, while $U = \{u_1, u_2, u_3, \dots, u_n\}$ is the upper bound of X. In this work, we test and verify the effectiveness of the proposed LFQPSO algorithm in this chapter by using the following 10 benchmark functions of which its definition, domains and optimal value are respectively defined as:

(1) The definition of f_1 is following :

$$f_1(x) = \sum_{i=1}^{n} x_i^2 \tag{11}$$

Its domains is $-5.12 \leq x_i \leq 5.12, i = 1,2,3, \dots, n$
Its argument and optimal value are $x^* = (0,0,0, \dots,0), f(x^*) = 0$
(2) The definition of f_2 is following :

$$f_2(x) = \sum_{i=1}^{n} (ix_i^2) \tag{12}$$

Its domains is $-5.12 \leq x_i \leq 5.12, i = 1,2,3, \dots, n$
Its argument and optimal value are $x^* = (0,0,0, \dots,0), f(x^*) = 0$
(3) The definition of f_3 is following :

$$f_3(x) = \sum_{i=1}^{n-1} [100\left(x_i^2 - x_{i+1}\right)^2 + (x_i - 1)^2] \tag{13}$$

Its domains is $-2.048 \leq x_i \leq 2.048, i = 1,2,3, \dots, n$
Its argument and optimal value are $x^* = (1,1,1, \dots,1), f(x^*) = 0$
(4) The definition of f_4 is following :

$$f_4(x) = 10n + \sum_{i=1}^{n} (x_i^2 - 10cos(2\pi x_i)) \tag{14}$$

Its domains is $-5.12 \leq x_i \leq 5.12, i = 1,2,3, \dots, n$
Its argument and optimal value are $x^* = (0,0,0, \dots,0), f(x^*) = 0$
(5) The definition of f_5 is following :

$$f_5(x) = \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \tag{15}$$

Its domains is $-600 \le x_i \le 600, i = 1,2,3, \dots, n$
Its argument and optimal value are $x^* = (0,0,0, \dots, 0), f(x^*) = 0$
(6) The definition of f_6 is following：

$$f_6(x) = 20 + e - 20e^{-\frac{1}{5}\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^{n} cos(2\pi x_i)} \quad (16)$$

Its domains is $-32.768 \le x_i \le 32.768, i = 1,2,3, \dots, n$
Its argument and optimal value are $x^* = (0,0,0, \dots, 0), f(x^*) = 0$
(7) The definition of f_7 is following：

$$f_7(x) = sin^2(\pi y_1) + \sum_{i=1}^{n-1}[((y_i - 1)^2(1 + 10sin^2(\pi y_i + 1)))] + (y_n - 1)^2\left(1 + sin^2(2\pi y_n)\right), y_i = 1 + \frac{x_i - 1}{4},\ i = 1,2,3, \dots, n \quad (17)$$

Its domains is $-10 \le x_i \le 10, i = 1,2,3, \dots, n$
Its argument and optimal value are $x^* = (1,1,1, \dots, n), f(x^*) = 0$
(8) The definition of f_8 is following:

$$f_8(x) = 1 - cos\left(2\pi\sqrt{\sum_{i=1}^{n} x_i^2}\right) + 0.1\sqrt{\sum_{i=1}^{n} x_i^2} \quad (18)$$

Its domains is$-100 \le x_i \le 100, i = 1,2,3, \dots, n$
Its argument and optimal value are $x^* = (0,0,0, \dots, n), f(x^*) = 0$
(9) The definition of f_9 is following:

$$f_9(x) = 0.1n - \left(0.1\sum_{i=1}^{n} cos(5\pi x_i) - \sum_{i=1}^{n} x_i^2\right) \quad (19)$$

Its domains is$-1 \le x_i \le 1, i = 1,2,3, \dots, n$
Its argument and optimal value are $x^* = (0,0,0, \dots, n), f(x^*) = 0$
(10) The definition of f_{10} is following:

$$f_{10}(x) = -\sum_{i=1}^{n-1}\left(\exp\left(\frac{-\left(x_i^2 + x_{i+1}^2 + 0.5x_i x_{i+1}\right)}{8}\right)\cos\left(4\sqrt{x_i^2 + x_{i+1}^2 + 0.5x_i x_{i+1}}\right)\right) + (n - 1) \quad (20)$$

Its domains is$-5 \le x_i \le 5, i = 1,2,3, \dots, n$
Its argument and optimal value are $x^* = (0,0,0, \dots, n), f(x^*) = 0$

4. Experiment analysis

The experimental environment is as follows: software is MATLAB and hardware environment is windows 7 of PC which uses 3.3GHz Core Duo processor and 4GB of RAM. The simulation experiment was run for 100 times and the average experiment results were statistically. The parameters of LFQPSO are α=0.6, β=1.5 m=100. Each function is iterative 1000 times for each function and the total iterative process is divided into 6 sub interval, the initial iteration times of the sub intervals and terminates the iterations respectively 1~100,100~200, 200~400, 400~600, 600~800 and 800~1000. Table 1, Table 2 and Table 3 respectively give the result of LFQPSO, QPSO and PSO which reflect their search ability. Table 4 presents the runtime of 1000 iteration for 10 benchmark function which measure the time performance of LFQPSO, QPSO and PSO.

Table 1: The result of 10 benchmark function obtained by PSO on the different iteration

function	1	100	200	400	600	800	1000
f_1	277.39	188.54	136.32	48.36	18.32	13.20	1.01
f_2	2987.21	2213.31	1001.47	621.58	284.19	152.43	17.81
f_3	6856.45	3853.82	2103.59	412.25	262.88	144.12	56.36
f_4	424.94	389.45	330.31	713.11	266.21	124.74	117.59
f_5	728.13	704.21	205.19	139.15	87.21	21.25	8.21
f_6	30.85	28.66	25.36	14.23	6.32	4.21	3.01
f_7	163.18	155.21	146.28	136.11	80.96	60.32	7.11
f_8	44.85	40.52	30.48	21.36	12.77	7.29	3.23
f_9	90.22	65.32	58.85	40.74	18.97	20.23	12.66
f_{10}	24.63	23.75	23.75	20.84	19.52	19.02	18.89

As shown table 1, Table 2 and Table 3, the LFQPSO performs better than QPSO and PSO for all 10 benchmark functions, such as, for the benchmark function f_2 the result of LFQPSO is 3.68, of QPSO is 12.87 and of PSO is 17.81 after 1000 iteration, this is, the final result of LFQPSO is far better than that of QPSO and PSO. The above situation is applicable to other 9 benchmark functions. The simulations for these benchmarks suggest that LFQPSO is a very efficient algorithm for function optimization. It can deal with highly nonlinear problems and multi peak problem with complex constraints.

Table 2: The result of 10 benchmark function obtained by QPSO on the different iteration

function	1	100	200	400	600	800	1000
f_1	234.66	107.66	87.96	29.36	11.22	9.36	0.02
f_2	2831.00	2099.39	988.62	455.35	241.87	93.61	12.87
f_3	6841.25	3769.33	1999.45	522.62	313.45	193.75	66.52
f_4	486.14	408.94	353.19	225.91	179.88	144.37	106.71
f_5	709.58	689.14	311.84	110.25	50.69	14.27	6.36
f_6	30.12	28.55	24.32	11.25	4.55	3.00	0.89
f_7	162.36	150.23	140.33	135.36	75.69	58.32	4.21
f_8	41.25	36.32	28.21	19.27	10.27	5.68	2.34
f_9	85.26	61.77	56.21	39.88	20.71	19.31	9.63
f_{10}	23.56	22.52	21.75	20.11	19.01	18.62	17.98

Table 3: The result of 10 benchmark function obtained by LFQPSO on the different iteration

function	1	100	200	400	600	800	1000
f_1	144.86	97.73	68.95	18.88	3.14	2.09	0.00
f_2	2520.21	2068.39	966.62	385.53	184.19	52.43	3.68
f_3	6230.88	3440.11	1690.14	412.25	262.88	144.12	56.36
f_4	424.94	389.45	330.31	214.29	168.81	124.74	97.41
f_5	618.13	505.32	205.19	38.16	11.94	1.64	0.00
f_6	20.77	20.52	19.04	5.49	1.88	0.14	0.01
f_7	131.53	131.53	131.53	131.53	55.39	41.53	1.85
f_8	24.83	24.32	22.03	11.55	2.93	1.61	0.82
f_9	83.25	60.72	51.85	38.74	15.77	12.02	8.36
f_{10}	23.63	21.12	19.51	16.44	14.02	12.22	9.91

As shown table 4, the runtime of LFQPSO is higher than that of QPSO which is higher than that of PSO. However, the income of new proposed algorithm LFQPSO, its time cost is worth. Take f_1 as an example, relative QPSO, although the time of LFQPSO increased by 40.01%, the performance increased by 90%. So, the above analysis shows that as far as the time performance and the solution quality is concerned, the new algorithm is far superior to the other two algorithms.

Table 4: The runtime of 10 benchmark function

function	PSO	QPSO	LFQPSO
f_1	2.48	5.24	7.34
f_2	2.56	5.40	7.57
f_3	2.63	5.56	7.78
f_4	2.78	5.88	8.23
f_5	2.74	5.79	8.10
f_6	2.74	5.79	8.11
f_7	2.61	5.52	7.73
f_8	2.87	6.06	8.49
f_9	2.68	5.66	7.93
f_{10}	2.71	5.73	8.02

5. Conclusion

A new improved quantum-behaved particle swarm optimization based on the levy flight (LFQPSO) has been presented. For all of the benchmark function that has been used in this work, the LFQPSO has been shown to outperform the QPSO and PSO. The superior performance of the LFQPSO is due to its ability to improve a local search while improve global search at the same time which is mainly because that LFQPSO uses the levy flight instead of the Brown motion so that it overcoming the particle wait effect of QPSO. Experiment result of 10 benchmark functions shows our proposed algorithm performs better than QPSO and PSO which shows that LFQPSO proposed in this work is not only suitable for the linear and single peak function but also for the nonlinear and multi peak function.

Reference

Kennedy J., Eberhart R., 1995, Particle swarm optimization, IEEE International Conference on Neural Networks, 4, 1942-1948, DOI: 10.1109/ICNN.1995.488968.

Qin A. K., Huang V. L., Suganthan P. N., 2009, Differential evolution algorithm with strategy adaptation for global numerical optimization, IEEE Transactions on Evolutionary Computation, 13(2): 398-417, DOI: 10.1109/TEVC.2008.927706.

Sun J., Feng B., Xu W., 2004, Particle swarm optimization with particles having quantum behavior. IEEE Conference of Evolutionary Computation, 1, 1571-1580, DOI: 10.1109/CEC.2004.1330875.

Sun J., Xu W., Fang W., Chai Z., 2010, Quantum-behaved particle swarm optimization with binary encoding, Control and Decision, 25(1), 376-385.

Walton S., Hassan O., Morgan K., Brown M.R., 2011, Modified cuckoo search: a new gradient free optimization algorithm. Chaos, Solitons & Fractals, 2011, 44(9):710-718, DOI: 10.1016/j.chaos.2011.06.004.

Yang X.S., Deb S., 2009, Cuckoo Search via lévy flights. Proceeding of world congress on Nature & Biologically Inspired Computing, India, 210-214.

Yang X.S., Deb S., 2010, Engineering optimization by Cukcoo Search, Int. J. Mathematical Modeling and Numerical Optimization, 1(4), 330-343.

Yang X.S., 2010, Engineering optimization: an introduction with meta heuristic applications, John Wiley and Sons.

Zhou Y. H., Huang L., Mao-Long X. I., 2010, Quantum-behaved particle swarm optimization algorithm with random selection of optimal individual. Wase International Conference on Information Engineering, 4, 189 – 193.