**TELKOMNIKA**, Vol. 11, No. 9, September 2013, pp. 4925~4934
ISSN: 2302-4046



# Local Memory Search Bat Algorithm for Grey Economic Dynamic System

**[1,2]Mo Yuanbin*, [1]zhao Xinquan, [1]xiang Shujian**
[1]Statistics and Mathematics Institute, Zhongnan University of Economics and Law, Wuhan 430073, China
[2]Guangxi Key Laboratory of Mixed Computing and Integrated Circuit Design and Analysis, Guangxi University for Nationalities, Nanning 530006, China
*Corresponding author, e-mail: moyuanbing@263.net

***Abstract***

*Control system is a pattern for describing microeconomic performance, so it can provide theory basis for policy-making to make economic performance well and continuously by analyzing and solving the model of economic control system. After analyzing the characteristics of Bat Algorithm (BA), the method to adjust each step of BA is proposed. In the method, each bat took advantage of the optimal location that it had found to guide the direction of search. The result of the case study showed that the proposed algorithm was efficient, then the proposed algorithm was used to solve the grey economic dynamic system, and the results further showed that the method was valid for solving economic control problems.*

***Keywords:*** *grey economic dynamic system, bat algorithm, local search, optimal control*



## 1. Introduction

Normally, a system can establish a clear economic background or structure with specific economical parameters and variables. A system with completely clear information is called white system. If a system structure is unclear and it's hard to clearly demonstrate the action principle, while some documents and information can be retrieved from its final results, such an abstract system with part of clear information, which is called grey system.

Grey system is usually applied in economic field. On one hand, as economy grows, economic system becomes more complex with increasingly harsher requirements for mathematical methods; on the other hand, our understanding of information has been deepened from simple certain information to complex uncertain information. Some random and fuzzy information is more and more frequently applied in economy. So far there's no good solution in the research of grey system [1-2]. As economy prevails in daily lives, it's of theoretical and practice significance in the research of grey economic control system. The research of numerical solution in grey control system using improved bat algorithm provides a new concept in solving this problem.

## 2. Grey Economic System

The influence of any economic policy is far from being limited to the period when it's established. It'll continue for a certain period of time. In mathematical model, constraints in economic development shall be expressed in dynamic equation, which is called dynamic model. Dynamic optimization controls manipulated variables in dynamic model to optimize performance indexes. Therefore, dynamic optimization is a basic problem in economic system engineering [3] of great theoretical and practice significance and under great attention [2]. Its mathematical model is:

$$\min\ J(\boldsymbol{u}) = \Phi(\boldsymbol{x}(t_f)) + \int_0^{t_f} \Psi[\boldsymbol{x}(t), \boldsymbol{u}(t), t]dt \quad \text{s.t.} \ \frac{d\boldsymbol{x}}{dt} = \boldsymbol{f}[\boldsymbol{x}(t), \boldsymbol{u}(t), t], \ \boldsymbol{x}(0) = \boldsymbol{x}_0$$

*Received February 10, 2013; Revised May 7, 2013; Accepted June 12, 2013*

Where $J$ is target function, $\boldsymbol{x}$ is state variable, $\boldsymbol{u}$ is control variable (policy control variable), $t_f$ is terminal moment, $\boldsymbol{f}(\cdot)$ is function vector, $\Phi(\cdot), \Psi(\cdot)$ is normal function relationship.

As the society develops and economic system becomes increasingly complex, our understanding has been deepened from simple certain information to complex uncertain information. The system with some uncertain factors is called grey system. The research of grey system has theoretical meaning (possibly grey mathematics in the future) and practical meaning. For example, big system has a lot of parameters but limited amount of information to acquire or process. So it's necessary to conduct research from grey parameters or even grey area. To some extent, only by researching grey system can we gradually deepen our research in big system and solve its problem. As for the national material allocation system, it covers a wide allocation area without full information about productivity, consuming, storing and distributing capacities in each area. However, it's necessary to establish a comprehensive real-time allocation plan [3] according to some certain parameters (white parameters). The terminal grey constraint is a grey control system, which can be expressed as:

$$\min \ J(\boldsymbol{u}) = \Phi(\boldsymbol{x}(t_f)) + \int_0^{t_f} \Psi[\boldsymbol{x}(t), \boldsymbol{u}(t), t]dt \ \text{s.t.} \frac{d\boldsymbol{x}}{dt} = \boldsymbol{f}[\boldsymbol{x}(t), \boldsymbol{u}(t), t],$$
$$\boldsymbol{x}(0) = \boldsymbol{x}_0, \boldsymbol{x}(t_f) = \boldsymbol{x}_f + \Delta\boldsymbol{b}, \Delta\boldsymbol{b} \in D^N$$

The value of such system terminal can be set within certain range. It's more flexible and practical comparing with fixed terminal. For example, usually the terminal shall be close to one value in economic operation. Fixed terminal can't suffice to solve this problem. Therefore, research in solving this problem has both theoretical and practical meaning.

Due to terminal grey constraint, it's difficult to solve this problem using normal traditional numerical algorithm. In recent years, as new intelligent bionic algorithms occur and become increasingly mature, they are more and more frequently applied in dynamic optimization. The biggest advantage of these algorithms is that they don't need to calculate gradient information. Therefore, more attentions have been given to them in the research of these problems. As basic swarm optimization algorithm has insufficient local extreme value, it's of practical significance to propose a new algorithm with better searching capacity to solve this problem.

## 3. Bat Algorithm

### 3.1. Bat's Behaviour and Bat Algorithm

During hunting for prey, bats emit about 10 to 20 sound pluses, which could be as loud as 110dB. The loudness also varies from the loudest when searching for prey and to a quieter base when homing towards the prey, while the rate of pulse emission increases, and it can be sped up to about 200 pulses per second. The loudness of pulse can help bat to detect further distance and frequency help bat to accurately grasp the variation of the space position of the prey. Such echolocation behavior that bat uses such a type of sonar, called, echolocation, to detect prey, avoid obstacles, and locate their roosting crevices in the dark can be associated with the objective function to be optimized, so bat algorithm was proposed by Yang X.S [4].

For it is simple to realize, BA have been successfully applied to engineering field. BA's basic iteration formula is as Formula (1) , (2) and (3).

$$f_i = f_{\min} + (f_{\max} - f_{\min})\beta \tag{1}$$

$$\boldsymbol{v}_i^t = \boldsymbol{v}_i^{t-1} + (\boldsymbol{x}^{t-1}_i - \boldsymbol{x}^*) f_i \tag{2}$$

$$\boldsymbol{x}_i^t = \boldsymbol{x}_i^{t-1} + \boldsymbol{v}_i^t \tag{3}$$

Where $\beta \in [0,1]$ is a random vector drawn from a uniform distribution, $f_i$ is i-bat frequency, $f_i \in [f_{\min}, f_{\max}]$, depending the domain size of the problem of interest, $\boldsymbol{x}^*$ is the current global best location (solution) which is located after comparing all the solutions among all the n bats.

During implementation, once a solution is selected among the current best solutions, a new solution for each bat is generated locally using random walk.

$$\boldsymbol{x}_{new} = \boldsymbol{x}_{old} + \boldsymbol{\varepsilon} A^t \quad (4)$$

Where $\varepsilon \in [-1,1]$ is a random number, while $A^t =< A_i^t >$ is the average loudness of all the bats at this time step. The process is called local search.

During hunting prey, the loudness usually decreases once a bat has found its prey, while the rate of pulse emission increases, the loudness and the rate can be updated by follow:

$$\boldsymbol{A}^{t+1}(i) = \alpha \boldsymbol{A}^t(i),\ r^{t+1}(i) = r^t(i)[1-\exp(-\gamma t)] \quad (5)$$

Where $\boldsymbol{A}(i)$ is i-bat loudness, $r(i)$ is the rate, $0 < \alpha < 1, \gamma > 0$ are constants, and we have used $\alpha = \gamma = 0.9$ in our simulations. $\boldsymbol{A}(i) = 0$ means that a bat has just found the prey and temporarily stop emitting any sound. We have:

$\boldsymbol{A}^t(i) \to 0$,

$r^t(i) \to r^0(i)$ as $t \to \infty$

We have used $A_i^0 = 0.5$, $r_i^0 = 0.5$.The loudness and emission rates will be updated only if the new solutions are improved, which means that these bats are moving towards the optimal solution.

### 3.2. Process for the Bat Algorithm

$\min f(\boldsymbol{x})$

Set the number of bat as $m$, $\boldsymbol{x}(i) = (x_{i1}, x_{i2}, \cdots, x_{id})$, $i = 1,2,\cdots,n$ is the location of i-bat, the pseudo code of the bat algorithm is as follows:

Initialize the bat population $\boldsymbol{x}(i)$, $i = 1,2,\cdots,n$ and $v_i$ ,define pulse frequency $f_i$ at $\boldsymbol{x}_i$, initialize pulse rates $r(i)$ and loudness $A(i)$.

$fitness(i) = f(\boldsymbol{x}(i))$, $i = 1,2,\cdots,n$

while (t<Max number of iterations)
  for *i*=1:*n*
  generate new solutions $\boldsymbol{x}(i)$ by(1)-(5)
    If *rand* > *r*(i)
      Generate a local solution $\boldsymbol{x}_{new}$ by(4)
    end if
    $fnew = f(\boldsymbol{x}_{new})$
    if *rand* < *A*(*i*)&& *fnew*< *fitness*(*i*)
      $\boldsymbol{x}(i) = \boldsymbol{x}_{new}$
      $fitness(i) = fnew$
      Update *R*(*i*) and *A*(*i*) by(5)
    end if
  end for
Rank the bats and find the current best $\boldsymbol{x}_*$

end while

## 4. Analysis Of Bat Algorithm

From the formula (1),(2) and (3), it is easy to see that BA is similar to Greedy algorithm, bat aims at moving towards the optimal position, while (1) only gives a variation step style. So it is easy to trap into local extreme value only by (1), (2) and (3) to update the location. The key of BA is that BA simulate pulse loudness and emission rate to construct a style local search. This local search can help BA jump out local extreme and make the algorithm can find a satisfied solution at acceptable time at a certain degree.

The local search that BA given is simple random walk, so it would be improve the algorithm Convergence speed and targeted search capacity.

## 5. Local Memory Search Bat Algorithm

A lots of function value were computed out in the process iteration, and ever function value reflects some characteristics of optimization function. These characteristics, if being fully used, will improve targeted search capacity. Particle swarm optimization (PSO) can jump out local extreme at a certain degree due to its introduction to local extreme. The basic iteration formula of PSO is as follow:

$$\boldsymbol{v}_{id}(t+1)=\omega\boldsymbol{v}_{id}(t)+c_1\phi_1(\boldsymbol{p}_{id}(t)-\boldsymbol{x}_{id}(t))+c_2\phi_2(\boldsymbol{p}_{gd}(t)-\boldsymbol{x}_{id}(t)) \quad (6)$$

$$\boldsymbol{x}_{id}(t+1)=\boldsymbol{x}_{id}(t)+\boldsymbol{v}_{id}(t+1) \quad (7)$$

It is easy to see that PSO is similar to Greedy algorithm too, without the adjustment for searching by local extreme. Just by local extreme adjust search direction, PSO has been applied in lots of field.

By the above analysis, we have concluded that it would improve BA, if the information that has been found out in the process of the implementation was fully combined and applied in BA. So Local Memory Search Bat Algorithm (LMSBA) was posed, and LMSBA was constructed by introducing local extreme search in BA local search, and the time complexity of LMSBA is same to BA. Its basic steps are as follows:

$\min f(\boldsymbol{x})$

Set the number of bat as $m$, $\boldsymbol{x}(i)=(x_{i1},x_{i2},\cdots,x_{id})$, $i=1,2,\cdots,n$ is the location of i-bat, the pseudo code of the bat algorithm is as follows:

Initialize the bat population $\boldsymbol{x}(i)$, $i=1,2,\cdots,n$ and $v_i$ ,define pulse frequency $f_i$ at $\boldsymbol{x}_i$, initialize pulse rates $r(i)$ and loudness $A(i)$.

$fitness(i)=f(\boldsymbol{x}(i))$, $i=1,2,\cdots,n$

while (t<Max number of iterations)

for *i*=1:*n*

generate new solutions $\boldsymbol{x}(i)$ by(1)-(5)

If *rand* > *r*(i)

generate a local solution $\boldsymbol{x}_{new1}$ by (4)

generate a local solution $\boldsymbol{x}_{new2}$ by (8)

$$\boldsymbol{x}_{new2}=\boldsymbol{x}_{old}+c\cdot rand\cdot(\boldsymbol{x}_{*d}-\boldsymbol{x}_{old}) \quad (8)$$

$\boldsymbol{x}_{*d}$ is the current best that i-bat has found.

compare $\boldsymbol{x}_{new1}$ and $\boldsymbol{x}_{new2}$ ,and note the better one $\boldsymbol{x}_{new}$

end if

$fnew=f(\boldsymbol{x}_{new})$

if *rand* < *A*(*i*)&& *fnew*< *fitness*(*i*)

$\boldsymbol{x}(i)=\boldsymbol{x}_{new}$

$fitness(i) = fnew$

Update *R*(*i*) and *A*(*i*) by(5)
end if
end for
Rank the bats and find the current best $\boldsymbol{x}_*$
end while

## 6. Simulation Experiment

Several typical function optimizations are applied to test the performance of the proposed algorithm:

a) Schaffer F6 function

$$\min f_1(\boldsymbol{x}) = \frac{\sin^2(\sqrt{x_1^2 + x_2^2} - 0.5)}{[1 + 0.001(x_1^2 + x_2^2)^2]^2} - 0.5, \quad x_i \in [-100, 100]$$

b) Hansen function

$$\min f_2(\boldsymbol{x}) = \sum_{i=1}^{5} i \times \cos((i-1) \times x_1 + i) \sum_{j=1}^{5} j \times \cos((j-1) \times x_2 + j), \quad x_i \in [-10, 10]$$

c) Rastrigin function

$$\min f_2(\boldsymbol{x}) = \sum_{i=1}^{10} [x_i^2 - 10\cos(2\pi x_i) + 10], \quad x_i \in [-5.12, 5.12]$$

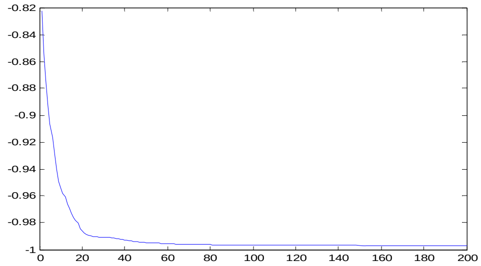
Among that, a) Schaffer F6 is a multimodal function of strong fluctuation with a theoretical value of -1. Optimization is completed if the optimal value is less than -0.9999 after search; b) Hansen function is also a multimodal function with 760 local minimal points and a theoretical optimal value of -176.5417. Optimization is completed if the optimal value is less than -176.5 after search; c) Rastrigin function is a highly multimodal function, with about 10n local minimal values (n is the dimension of solution space) in solution space and a theoretical optimal value of 0. Optimization is completed if the optimal value is less than 1 after search.

Parameter setting: particle swarm optimization: $c_1 = 0.3, c_2 = 0.2, c_3 = 0.2$; Firefly algorithm: $\gamma = 1.0$, $\beta_0 = 1.0$ and $\alpha = 0.02$; Bat algorithm and Local Memory Search Bat Algorithm: $\alpha = \gamma = 0.9$, $A_i^0 = 0.5$, $r_i^0 = 0.5$, $c = 1.8$, Refer to the following table for test results:

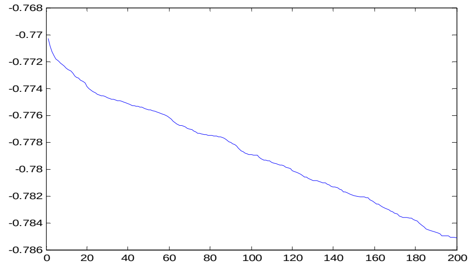
Table 1. Test Result

| Function | Dimension | Number of groups | Average optimal value / (variance) | | | |
|---|---|---|---|---|---|---|
| | | | PSO | FA | BA | LMSBA |
| $f_1(x)$ | 2 | 100 | -0.9968/ (7.9895e-006) | -0.7732/ (0.0100) | -0.9892/ 9.1197e-006 | -0.9915/ 8.2141e-006 |
| $f_2(x)$ | 2 | 100 | -173.6571/ (56.2349) | -168.9492/ (233.9582) | -175.7298 / 0.4984 | -176.0380/ 0.2178 |
| $f_3(x)$ | 10 | 100 | 20.9903/ （81.7098） | 57.1848/ (54.7250) | 63.0017/ 38.6860 | 47.2325/ 49.9077 |

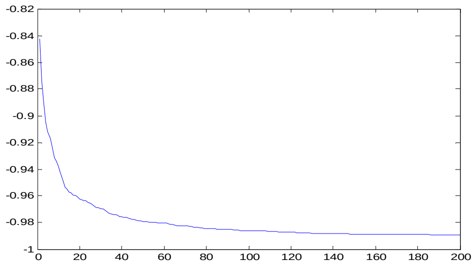
Refer to the following curves for the variation of average value:



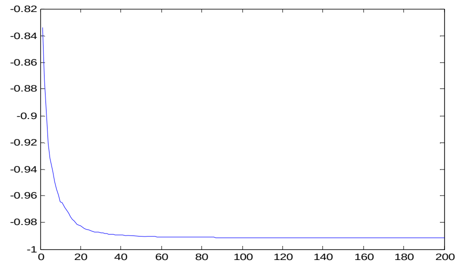

(1) PSO




(2) FA

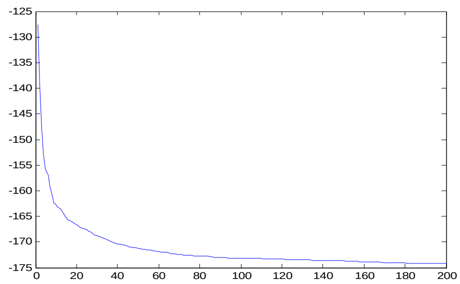



(3) BA

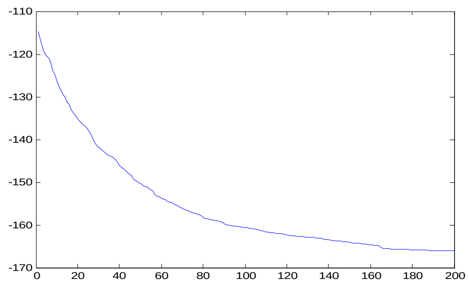



(4) LMSBA

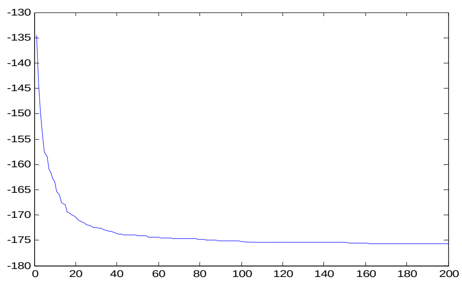
Figure1. Variation Figure of Average Value in Function 1



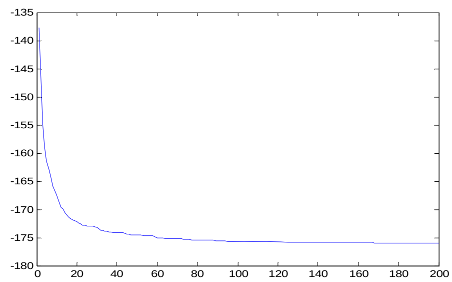

(1) PSO




(2) FA




(3) BA




(4) LMSBA

Figure 2. Variation Figure of Average Value in Function 2

**Algorithm analysis:** according to the results in Table 1, LMSBA is more stable than other algorithms and has better optimization ability than firefly algorithm. In the test of example 2, its optimization is better than particle swarm. The main reason is to adjust search direction through local individual extreme value after introducing memory, thus moving towards optimal direction. LMSBA is better than PSO、FA、BA at a certain degree.

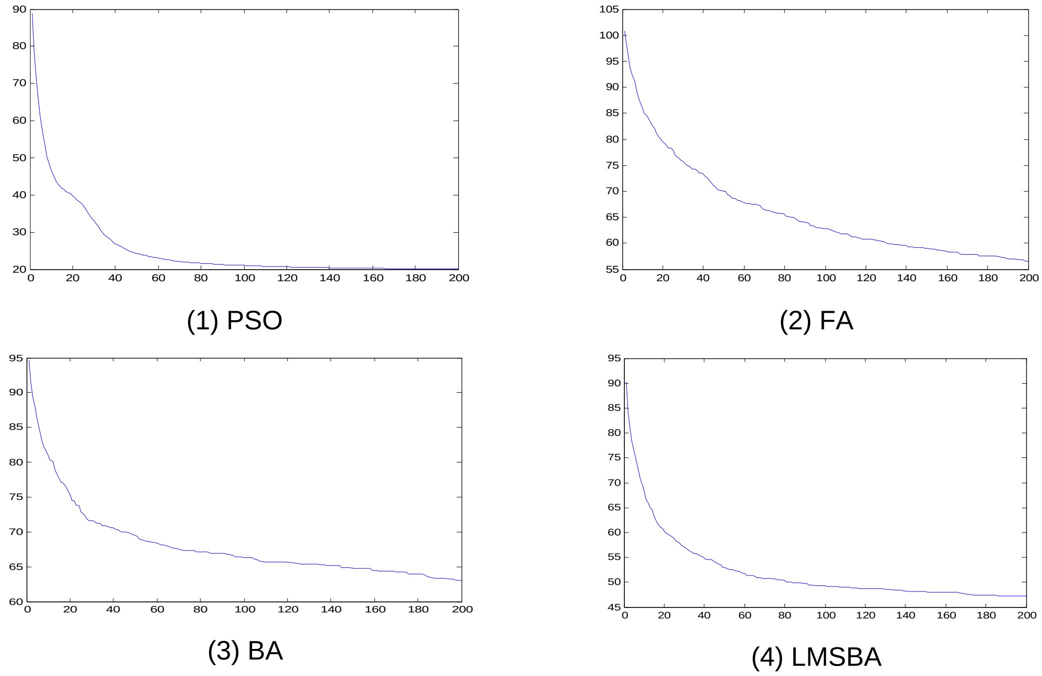



Figure 3. Variation Figure of Average Value in Function 3

## 7. Application in Optimal Control

### 7.1. Application of Swarm Intelligence Algorithm and Its Improved Algorithm in Solving Dynamic Optimization Problem

Divide controlled time interval $[t_1, t_2]$ into $m$ intervals with constant control variable and state variable. The value of control variables at each interval form a sequence $u_0, u_1, ...u_{m-1}$, turning dynamic control into optimization problem with dimension $m$. For this, the solution steps by applying various swarm intelligence algorithms are:

Step 1. Initialize a group of individuals and form an initial group of certain scale. Each individual is a randomly produced sequence of control variable $u_0, u_1, ...u_{m-1}$.

Step 2. Apply the sequence of control variables of each individual in controlled object to get related state trajectory, and calculate individual performance index according to the target function, including numerical calculation of simultaneous differential equations.

Step 3. Determine if it meets end conditions. If it does, end calculation and output optimal result; otherwise perform the next step.

Step 4. Calculate the group of next generation according to each step of the algorithm, return to step 2 and continue operation.

### 7.2. Test

Take the optimal control of production inventory system in literature [3] as a test example.

If an enterprise produces a product whose output and sales volume are respectively $u(t)$ and $s(t)$ in quarter $t$, and inventory at the beginning of quarter $t$ is $x(t)$. So the inventory at the beginning of quarter $t+1$ equals the inventory at the beginning of quarter $t$ plus output and minus sales volume in quarter $t$ (suppose there's no waste in inventory). So the state equation of this production inventory system is:

$$x(t+1) = x(t) + u(t) - s(t)$$

Given that the product's operating cost is proportional to the square of output, and the proportionality factor is 0.005; inventory cost is 1 yuan in each quarter for each product. So the total annual cost of four quarters is:

$$J_4 = \sum_{t=0}^{3}[0.005u^2(t) + x(t)]$$

Suppose $t=0$ is 1st quarter, $t=1$ is 2nd quarter, and this enterprise has no inventory at the beginning of the year and no overstock at the end of the year, namely $x(0)=0, x(4)=0$. Order quantity of 4 quarters is:
$s(0)=600, s(1)=700, s(2)=500, s(3)=1200$ Management problem of the production inventory system is: calculate the output of 4 quarters $u(0),u(1),u(2),u(3)$, so as to minimize total cost $J_4$.
Solution: $u(0)=600, u(1)=700, u(2)=800, u(3)=900$, and $J_4$=11800 yuan.

For boundary problem, adopt penalty function method and calculate for 100 times using BA and LMSBA. All solutions meet constraints, while the solution by BA 12798, and 12785 by LMSBA. These results showed that LMSBA is better than BA. The evolving curves for average target function value of 2 algorithms are as follows:

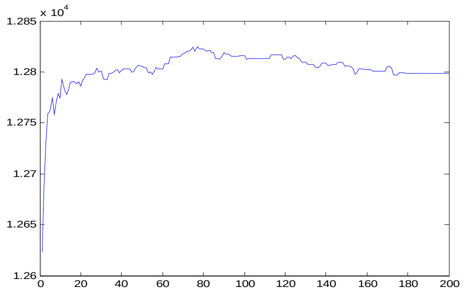



Figure 4. Variation Curve of Average Adaptive Value by BA

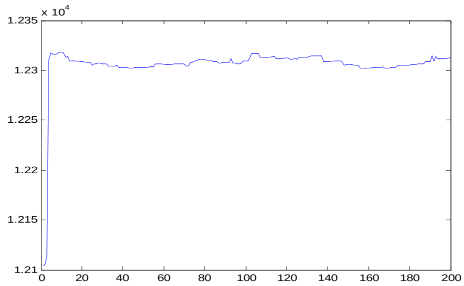



Figure 5. Variation Curve of Average Adaptive Value by LMSBA

## 8. Application in Economic System with Fixed Terminal

Apply the proposed algorithm in the grey economic control system in literature [1], as described below:

$$\begin{cases} \boldsymbol{x}^0 = (1126, 2010, 3500) \\ \boldsymbol{x}(k+1) = \boldsymbol{x}(k) + \boldsymbol{u}(k), k = 0,1,2 \\ \boldsymbol{x}(3) = (1340 + \Delta b, 4502 + \Delta b, 4085 + \Delta b) \\ \boldsymbol{s}(k) = (\boldsymbol{E} - \boldsymbol{A})x(k) - \boldsymbol{B}u(k) \\ \min\ J = \sum_{k=0}^{2}[\boldsymbol{d}(k) - \boldsymbol{s}(k)]^T \boldsymbol{Q}[\boldsymbol{d}(k) - \boldsymbol{s}(k)] \end{cases}$$

Where:

$$\boldsymbol{A} = \begin{bmatrix} 0.15 & 0.27 & 0.0398 \\ 0.05 & 0.27 & 0.0502 \\ 0.1 & 0.1 & 0.4 \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} 0.1 & 0.21 & 0.0401 \\ 0.02 & 0.201 & 0.205 \\ 0.2 & 0.003 & 0.2055 \end{bmatrix}$$

$\boldsymbol{d}(k) = (1010+10k, 1805+20k, 2155+15k)^T, k=0,1,2$ $-550 \le \Delta b \le 550$, It's a grey economic control system due to a grey value at the terminal instead of a specific value.

For the grey economic control system, perform 100 calculations using BA and LMSBA respectively. Result of BA algorithm is:

$$J = 17.3340,$$

$$u = \begin{pmatrix} 976.1 & 252.9 & -1403.8 \\ 400.4 & 452.3 & 1197.8 \\ 475.5 & -51.1 & -93.7 \end{pmatrix},$$

x(4)=(952, 4060.5 3830.7).

Variation curves of average adaptive value in the two algorithms are:

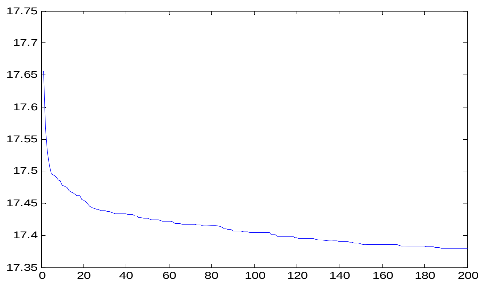


Figure 6. Variation Curve of Average Adaptive Value by BA

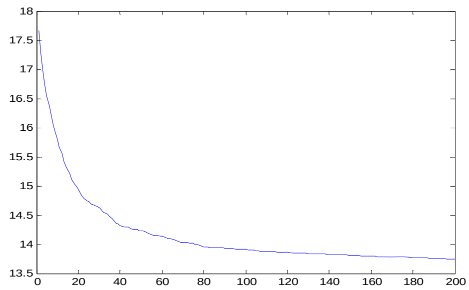


Figure 7. Variation Curve of Average Adaptive Value by LMSBA

Result in literature [1] using membership method is: 22.53; result in literature [7] using particle swarm optimization is 21.903; result in literature [8] using fuzzy method is 22.71; From the results above, the result by LMSBA is best.

## 9. Conclusion

In this paper, the LMSBA is introduced in economic control field, test and simulation results are ideal, and programming of method is concise. This algorithm is suitable for numerical solution in practical dynamic economic control, providing numerical theoretical foundation for steady, healthy and optimal economic growth.

## Acknowledgments

This work was supported by Research Projects of 1. China Postdoctoral Foundation (2012M511711), 2. Guangxi Department of Education Item (201204LX082), 3. Item of Guangxi University for Nationalities (2011MDYB030).

## References:

[1] Liu Guozhi, Went Dongri. Optimal Grey Control Model from Dynamic Input and Output. *Mathematics in Practice and Theory*. 2006; 36(3): 15-18.
[2] Zhang Junliang. Research on Predictive Parsing using Grey Control System Model. *Computer and Network*. 2012; 11: 69-70.
[3] Gong De'en. Economic Cybernetics. *Higher Education Press*. 2008.
[4] Yang Xinshe. A new metaheuristic bat-inspired algorithm. GONZÁLEZ JR, PELTA DA. Nature Inspired Cooperative Strategies for Optimization. Berlin: Springer- Verlag,2010:65-74.
[5] LEMMA TA, HASHIM BM. *Use of fuzzy systems and bat algorithm for exergy modeling in a gas turbine generator.* Proceedings of IEEE Colloquium on Humanities, Science and Engineering. Penang, Malaysia. 2011; 305-310.
[6] BORA TC, COELHO LS, LEBENSZTAJN L. Bat-inspired optimization approach for the Brushless DC wheel motor problem. *IEEE Transactions on Magnetics*. 2012; 48(2): 947-950.

[7] Miao Chen, Liu Guozhi. Particle Swarm Optimization of Optimal Control Model from Dynamic Input and Output. *Scientific Technology and Engineering*. 2009; 15(9): 4497-4499.
[8] Mao Yunying, Yang Zhengfang, Wang Xiuping, Han Lixing. Optimal Fuzzy Control Model from Dynamic Input and Output", *Theory and Practice of System Engineering*. 1996; 7: 1-4.