Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/072L24

A Glowworm Swarm Optimization Algorithm Based Tribes

Yongquan Zhou[1,2], *Guo Zhou*[3], *Yingju Wang*[1] *and Guangwei Zhao*[1]

[1]College of Information Science and Engineering, Guangxi University for Nationalities, Nanning, 530006, China
[2]Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis, Nanning 530006, China
[3]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100081, China

Received: 20 Oct. 2012, Revised: 8 Jan. 2013, Accepted: 11 Dec. 2013
Published online: 1 Jun. 2013

Abstract: This paper based on the metaphor of specialization and cooperation in steppe tribes of the human society, tribe glowworm swarm optimization (TGSO) algorithm was presented to solve the problem of low precision and easy to fall into local optimization of the glowworm swarm optimization (GSO) algorithm. In the proposed tribe glowworm swarm optimization approach, all glowworms are divided into a certain tribes. Finally, the flow shop scheduling problem is utilized to illustrate the effectiveness of the proposed tribe glowworm swarm optimization approach.

Keywords: Glowworm swarm optimization, tribe glowworm swarm optimization, flow shop scheduling problem.

1 Introduction

Glowworm Swarm Optimization (GSO) algo-rithm which is a novel swarm intelligence algo-rithm was advanced by Indian scholars K. N. Krishnanand and D. Ghose in 2005 years. The inspiration of the basic GSO algorithm comes from the phenomenon that one glowworm can be attracted by another one which has a higher luciferin value and then moves toward it. In the nature, glowworms communicate with each other by releasing luciferin. Glowworms release luciferin when they are flying, so they can give out luciferin light. Glowworms attract others around them by giving out fluorescent light. The higher the concentration of luciferin, the greater the intensity of fluorescence, then glowworm can be able to attract more other glowworms. By simulating this phenomenon, the characteristics of the artificial glowworm swarm optimization algorithm can be drawn: the search strategy of this algorithm is multi-point parallel global randomly search based on population and without the evolution of complex operations. It according to the individual glowworm's decision range determines the search path, and then achieves the optimal effect. So far, GSO algorithm has been successfully used in the noise test, simulation of the sensor machine crowd, clustering analysis, numerical optimization calculation, knapsack problem, etc.

Although GSO algorithm has a strong versatility, but this algorithm still exist some shortcomings, such as low precision and easy to fall into local optimization. To solve these shortcomings, this article refers to human society that classification management in the tribes of the grassland then put the classification management structure into the basic GSO algorithm and proposed a glowworm swarm optimization algorithm based tribes. In order to verify the effectiveness of the proposed algorithm in this paper, successfully applied it to the flow shop scheduling problem. The experimental results show that the proposed algorithm has obvious advantages in the convergence and in terms of overall.

The paper is organized as follows. The basic glowworm swarm optimization algorithm is given in Section 2. The glowworm swarm optimization algorithm based tribes is presented in Section 3. Section 4 deals with the effect of algorithm's parameters on TGSO's performance. The result shows that have higher computation accuracy and convergence speed. This paper concludes with some remarks in Section 5.

* Corresponding author e-mail: yongquanzhou@126.com

2 The Basic Glowworm Swarm Optimization Algorithm

In the basic GSO algorithm, a swarm of glowworms are randomly distributed in the search space of object functions. Accordingly, these glowworms carry a luminescent quantity called luciferin along with them and they have their own decision domain $r_d^i(0 < r_d^i \leq r_s)$. The glowworms emit light which intensity is proportional to the associated luciferin and interact with other glowworms within a variable neighborhood. The glowworms' luciferin inten-sity is related to the fitness of their current loca-tions. The higher the intensity of luciferin, the better the location of glowworm, in other words, the glowworm represents a good target value. Otherwise, the target value is poor. A glowworm i considers another glowworm j as its neighbor if j is within the neighborhood range of i and the luciferin level of j is higher than that of i. In particular, the neighborhood is defined as a local-decision domain that has a variable neighborhood range r_d^i bounded by a radial sensor range $r_s(0 < r_d^i \leq r_s)$. Each glowworm selects, using a probabilistic mechanism, a neighbor that has a luciferin value higher than its own and moves toward it. That is, glowworms are attracted by neighbors that glow brighter. In addition, the size of the neighborhood range of each glowworm is influenced by the quantity of glowworms in the neighborhood range. The neighborhood range of the glowworm is propor-tional to the density of its neighbors. If the neighborhood range covers low density of glowworms, the neighborhood range will be increased. On the contrary, the neighborhood range will be reduced.

Generally, GSO algorithm includes four stages as follows:

[(1)]*The initial distribution of glowworms phase* The initial distribution of glowworms phase, in other words, it is initialization phase. Purpose is to make the glowworms randomly distribute in the search space of object functions. Accordingly, these glowworms carry the same intensity luciferin and they have the same decision domain r_0. *Luciferin-update phase* The glowworms' luciferin intensity is related to the fitness of their current locations. The higher the intensity of luciferin, the better the location of glowworm, in other words, the glowworm represents a good target value. Otherwise, the target value is poor. In the algorithm of each iteration process, all the glowworms' position will change, and then the luciferin value also follows updates. At time t, the location of the glowworm i is $x_i(t)$, corresponding value of the objective function at glowworm i's location at time t is $J(x_i(t))$, put the $J(x_i(t))$ into the $l_i(t)$, $l_i(t)$ represents the luciferin level associated with glowworm i at time t. The formula as follows

$$l_i(t) = (1-\rho)l_i(t-1) + \gamma J(x_i(t)), \tag{1}$$

where ρ is the luciferin decay constant $(0 < \rho < 1)$, γ is the luciferin enhancement constant. *Movement-phase* At the movement phase, every glowworm se-lects a neighbor and then moves toward it with a certain probability. As the glowworm i's neighbor need to meet two requirements: one, the glowworm within the decision domain of glowworm i; two, the luciferin value is larger than the glowworm i's. Glowworm i moves toward a neighbor j which comes from $N_i(t)$ with a certain probability, the probability is $p_{ij}(t)$. Using the formula (2) calculates it:

$$p_{ij}(t) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} l_k(t) - l_i(t)} \tag{2}$$

Glowworm i after moving, the location is updated, the location update formula is

$$x_i(t+1) = x_i(t) + st * \left(\frac{x_j(t) - x_i(t)}{\|x_j(t) - x_i(t)\|} \right) \tag{3}$$

where st is the step size. *Neighborhood range update phase* With the glowworm's position updating, it neighborhood range also follow update. If the neighborhood range covers low density of glow-worms, the neighborhood range will be increased. On the contrary, the neighborhood range will be reduced. The formula of neighborhood range update as follows:

$$r_d^i(t+1) = \min\{r_s, \max\{0, r_d^i(t) + \beta(n_t - |N_i(t)|)\}\} \tag{4}$$

where β is a constant parameter and n_t is a parameter used to control the number of neighbors. The Basic GSO algorithm as follows [1]:

Set number of dimensions $= m$ Set number of glowworms $= n$ Let s be the step *size* Let $x_i(t)$ be the location of glowworm i at time t *Deploy_agents_randomly*; for i=1 to n do $l_i(0) = l_0$ $r_d^i(0) = r_o$ set maximum iteration number=$iter\_max$; while ($t < iter\_max$) do { for each glowworm i do: % Luciferin-update phase; $l_i(t) = (1-\rho)l_i(t-1) + \gamma J(x_i(t))$ %See(1) for each glowworm i do :% Movement-phase { $N_i(t) = \{j : d_{ij}(t) < r_d^i(t); l_i(t) < l_j(t)\}$; for each glowworm $j \in N_i(t)$ do; $p_{ij}(t) = \dfrac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} l_k(t) - l_i(t)}$; %See(2) $k = select\_glowworm(\vec{p})$ $x_i(t+1) = x_i(t) + st * \left(\frac{x_j(t) - x_i(t)}{\|x_j(t) - x_i(t)\|} \right)$; %See(3) $r_d^i(t+1) = \min\{r_s, \max\{0, r_d^i(t) + \beta(n_t - |N_i(t)|)\}\}$ %See(4) } $t \leftarrow t+1$: }

Implementation at the GSO at the individual agent level gives rise to two major phases at the group level: Formation of dynamic networks that results in splitting of the swarm into sub-swarms and local convergence of glowworms in each subgroup to the peak locations. In this paper, the glowworm swarm optimation algorithm based tribes are briefly described below.

3 Glowworm Swarm Optimization Algorithm Based Tribes

In this section, we make them imitate community-the organizational structure of the human society and then make the basic GSO further improve. In the vast grassland of the human society, there are many small tribes. Each tribe has a tribal leader who has very strong ability to lead them, and the grassland is always to the leadership by the total head, while the total head was selected from the tribal leaders. Based on this kind of mechanism, a glowworm swarm optimization algorithm based tribes (TGSO) was proposed in this paper.

3.1 Tribal Structure

Assume that there are totally $l \times m$ glow-worms in the population P. In the initiation step of tribe-GSO algorithm, the population P is divided into l sub-populations, called as tribes. Each tribe has the same structure of the basic GSO model: it has m glowworms and the best glowworm from them is called tbest. In each tribe, glowworms obey the formula of basic GSO to select the tbest, so every tribe $i(i = 1,2,..,l)$ has a tribal optimum $tbest_i$. The global optimum of the population *gbest* get from the tribal optimum $tbest_i(i = 1,2,..,l)$. The tribal structure is shown in Figure 1.

In the tribe-GSO algorithm, it has two layers: all tribes form the first layer; the glowworms with the brightest light of each tribe form the second layer, so the global optimum comes from the second layer according to the basic GSO. In the first layer, tribes operate independently, each tribe do not have any information exchange. In the second layer, tribes communicate with each other using the glowworm with the brightest light of each tribe, and then obtain the global optimum.

3.2 The Glowworm Swarm Optimization Algorithm Based Tribes (TGSO)

The steps of the glowworm swarm optimization algorithm based tribes can be described as follows:

Step 1: Initialize the population P, dimension m, the number of glowworms $l \times m$, step size *st* and so on.
Step 2: The population P is divided into several sub-populations. The number of sub-population is l. The size of sub-populations is m. Each sub-population is a tribe.
Step 3: Placing the glowworms randomly in the search space of the object function.

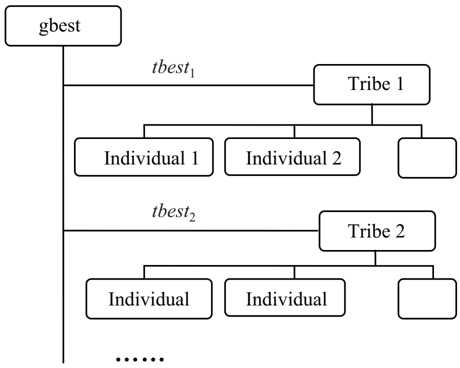

Fig. 1 Based tribes structure of Tribe-GSO

Step 4: Tribes operate independently. Using the formula (1) put the $J(x_i(t))$ into the $l_i(t)$, $l_i(t)$ represents the luciferin level associated with glowworm i at time t. $J(x_i(t))$ represents the value of the objective function at glowworm i's location at time t.
Step 5: Each glowworm in the tribes selects a neighbor that has a luciferin value higher than its own to make up the $N_i(t)$.
Step 6: The glowworm i in the tribes using the formula (2) select a neighbor and move toward it, then using the formula (3) update the location of the glowworm i.
Step 7: Glowworm i use the formula (4) to update the value of the variable neighborhood range.
Step 8: Selecting the glowworm with the brightest light of each tribe at time t, then form the second level using them.
Step 9: Glowworm in the second level obey the formula of the basic GSO to select the glowworm with the brightest light.
Step 10: If reached the maximum number of iterations, executing the step (11); otherwise, executing the step (4).
Step 11: Output the results. The end.

Based tribes structure of Tribe-GSO algorithm flow char is as Figure 2.

4 Simulation Results

4.1 Experimental Platform

Environment for running programs of this experiment: processor: CPU T4400, main frequency: 2.20GHz, memory: 2.00GB, operating system: Windows 7.0, Mathematical software: Matlab 7.0.

4.2 Parameters Set

The glowworm swarm optimization algorithm based tribes algorithm used to solve flow shop scheduling problem.

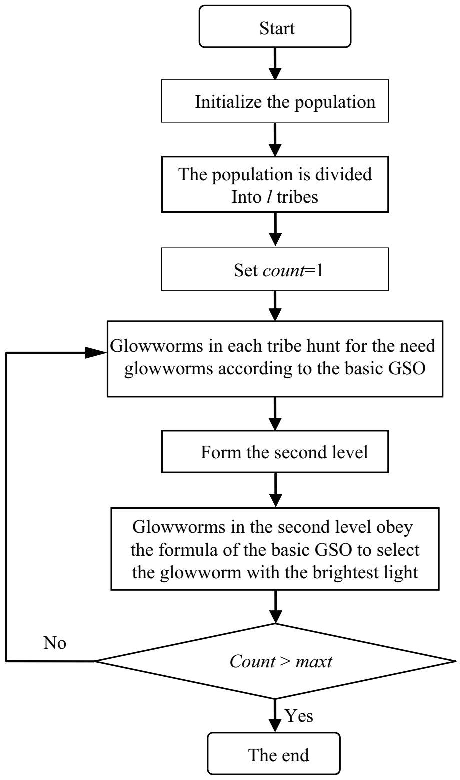

Fig. 2 Based tribe's structure of Tribe-GSO algorithm flow char

In this section, values of algorithm parameters that are kept fixed for all the experiments in Table 1.

Table 1 The values of algorithm parameters

ρ	β	γ	st	n_t	l_0
0.4	0.08	0.6	0.03	5	5

The flow shop scheduling problem can be described as follows: n tasks are $J_i(i = 1,2,...,n)$, respectively, every task J_i contains m processes $(J_{i1}, J_{i2}, ..., J_{im})$. m processes will be executed on the m machines using the same order in the pipeline. The event that the task J_i was executed in the machine M_j was known as an operation O_{ij}, the operation had a processing time t_{ij} and a waiting time W_{ij}. The goal of scheduling is to arrange the order processes, making the least amount of time to complete these tasks.

The objective functions of flow shop scheduling problem as follow: $T = \min(\sum_{i=1}^{n}\sum_{j=1}^{m} O_{ij})$. The operation that n tasks were executed on m machines must be satisfied some requirements:

[(1)]All tasks can be processed at time zero, and the task processing is not interrupted. One machine can only process a task and a task can only be processed by a machine at the same time. Every task required completing a process at each stage and each process of every task can be processed at any machine in the corresponding stage.

The time required for the workshop process in this experiment as Table 2.

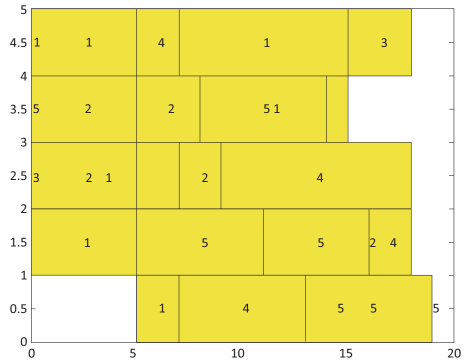

Fig. 3 Gantt chart of 5×5 instance (GA algorithm)

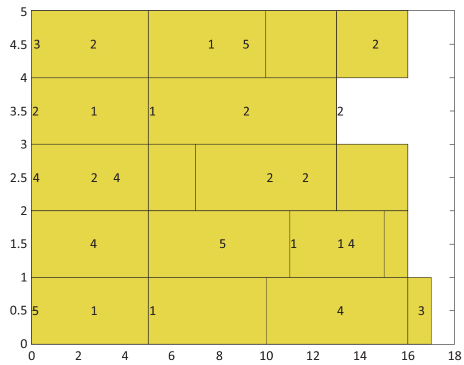

Fig. 4 Gantt chart of 5×5 instance (GSO algorithm)

Table 2 5×5 instance of processes processing time t_{ij}

t_{ij}	1	2	3	4	5
1	6	5	2	8	3
2	1	5	3	8	6
3	9	5	7	2	9
4	5	6	5	9	4
5	2	6	10	6	1

From the Figure 3 to Figure 5, we can see that for the 5×5 instance of processes using the traditional GA algorithm to solve the flow shop scheduling problem. the least time we used to get the best solution is 17 seconds; using the basic GSO algorithm to solve it, the minimum time is greater than 17 seconds; using TGSO algorithm which presented in this article to solve it, the minimum is 15 seconds. The experimental results show that TGSO algorithm is very effective for solving practical problems.

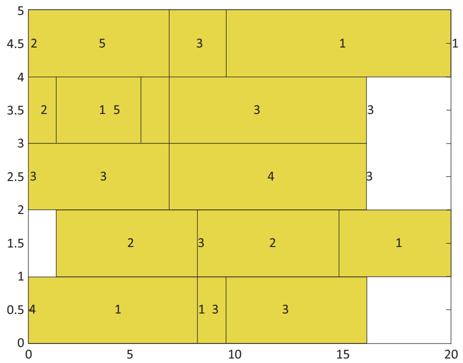

Fig. 5 Gantt chart of 5×5 instance (TGSO algorithm)

5 Conclusions

In this paper, firstly, the model of basic GSO algorithm has been described. Then, a novel glowworm swarm optimization algorithm based tribes is proposed and applied it to the flow shop scheduling problem. The experimental results show that the algorithm has a great improvement in the convergence and in terms of overall. Further work in this direction involves how to use the theory to prove the effectiveness of the proposed algorithm and what are the influences of parameters on the results of the algorithm.

Acknowledgement

This work was supported in part by NSFC under Grant No. 61165015; Key Project of Guangxi Science Foundation under Grant No. 2012GXNSFDA053028; Guangxi higher school significant scientific research project under Grant No. 201201ZD008 and funded by open research fund program of key lab of intelligent perception and image understanding of ministry of education of China under Grant No. IPIU012011001.

References

[1] K.N.Krishnand, D. Ghose. Glowworm Swarm Optimisation for Simultaneous Capture of Mutiple Local Optima of Multimodal Functions. Swarm Intell. **3**, 87-124 (2009).
[2] K.N.Krishnand, D. Ghose. Glowworm Swarm Optimisa-tion: A New Method for OptimizingMulti-modal Func-tions. Int. J. Computational Intellingence Studies. **1**, 93-11 (2009).
[3] K.N.Krishnand, D.Ghose. Glowworm Swarm Based Optimization Algorithm for Multimodal Functions with Collective Robotics Applications. Multiagent and Grid Sys-tems. **3**,209-222 (2006).
[4] Kuo-Tai Tseng. A Glowworm Algorithm for Solving Data Clustering Problems. Tatung University, 2008.
[5] Kai Chen, Tonghua Li, Tongcheng Cao.Tribe-PSO: A novel Global Optimization Algorithm and Its Application in Molecular Docking. Science direct.,248-25 (2006).
[6] Cai Liang-wei, Zhang Ji-hong, Li Xia. A Multi-Population Genetic Algorithm for Job Shop Scheduling Problem. Acta Electronica Sinica. **6**, 991-994 (2005).
[7] Chen En-hong, Liu Gui-quan, Cai Qing-sheng. A Genetic Algorithm Based Job-shop Scheduling Problem Solving Method .Journal of Software, **2**,139-143 (1998).
[8] Song Xiao-yu, Gao Yang, Mwng Qiu-hong. Study on Particle Swarm Algorithm for Job Shop Scheduling Problems. Systems Engineering and Electronics, **12**, 93-96 (2008).
[9] Wang Zhong-hua, Gao Mao-ting. Solve Job-shop Scheduling Problem Based on NPSO Algorithm. Computer Simulation, **4**,313-315 (2010).
[10] Cai Bin, Mao Fan, Fu Li. Hybrid Algorithm of Particle Swarm Optimization and Stimulated Annealing for Job-Shop Scheduling. Application Research of Computers.**3**, 856-859 (2010).
[11] Matlab Application. Beijing: Bublishing House of Electronics Industry, 2008.

Yongquan Zhou, Ph.D & Professor. His received the MS degree in Computer science from Lanzhou University in 1993, and the PhD degree in Computation Intelligence from the Xiandian University in 2006. He is currently a professor in Guangxi University for Nationalities. His research interests are in the areas of computation intelligence, neural networks, and intelligence in-formation processing.

Guo Zhou, Ph.D. was born in 1987. His research interests include Virtual reality and digital animation.

Yingju Wang, master, was born in 1987. Her research interests include computation intelligence, swarm intel-ligence algorithm and applies.

Guangwei Zhao, master, was born in 1985. His research interests include computation intelligence, swarm intel-ligence algorithm and applies.