Chapter 12
Group Search Optimizer Algorithm

Abstract In this chapter, we introduce a new optimization algorithm called group search optimizer (GrSO) which is inspired from the relationship of group foraging behaviours, i.e., producer-scrounger paradigm. We first describe the general knowledge of the producer-scrounger model in Sect. 12.1. Then, the fundamentals and performance of GrSO are introduced in Sect. 12.2. Finally, Sect. 12.3 summarises this chapter.

12.1 Introduction

The most easily recognized animals' behaviour is the foraging behaviour, i.e., searching for and exploiting food resources (Mills et al. 2010). Nowadays, several population-based algorithms are proposed based on the foraging theory, such as ant colony optimization (ACO) and particle swarm optimization (PSO). Recently, He et al. (2009) introduced a newly developed algorithm called group search optimizer (GrSO) algorithm which is inspired from the relationship of group foraging behaviours, i.e., producer-scrounger paradigm (Millor et al. 2006).

12.1.1 Producer-Scrounger Model

Generally speaking, the producer-scrounger (PS) model is a group-living foraging strategy in which the food will be founded by the discoverers (producers) and shared with others (scroungers) (Barnard and Sibly 1981; Brockmann and Barnard 1979). It assumed that individuals should specialize in either producing or scrounging at any one time (Parker 1984). In fact, it is the novel behaviour and adopted by many animals. To understand when and how such exploitative relationships will occur, several studies are made. For example Giraldeau and Lefebvre (1987) studied the PS model in pigeons, and (Biondolillo et al. 1997)

B. Xing and W.-J. Gao, *Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms*, Intelligent Systems Reference Library 62,
DOI: 10.1007/978-3-319-03404-1_12,

tested the PS model between the zebra finches. In addition Vickery et al. (1991) proposed a newly model in which an information sharing mechanism (i.e., a third strategist: producer-scrounger opportunist) is incorporated.

12.2 Group Search Optimizer Algorithm

12.2.1 Fundamentals of Group Search Optimizer Algorithm

Group search optimizer (GrSO) algorithm was originally proposed in He et al. (2006). The population of GrSO is called group and each one inside is called member. Based on the PS model, the main steps of GrSO can be described as follows (Chen et al. 2012; Shen et al. 2009):

- Initializing phase: In GrSO, ith member at the kth searching iteration has a position $\mathbf{X}_i^k \in R^n$, a head angel $\boldsymbol{\varphi}_i^k = \left(\boldsymbol{\varphi}_{i,1}^k, \ldots, \boldsymbol{\varphi}_{i,n}^k\right) \in R^{n-1}$, and a head direction $\mathbf{D}_i^k\left(\boldsymbol{\varphi}_i^k\right) = \left(\mathbf{d}_{i,1}^k, \ldots, \mathbf{d}_{i,n}^k\right) \in R^n$ in an $n-$dimensional search space. In general, the distance can be calculated from $\boldsymbol{\varphi}_i^k$ via a polar to Cartesian coordinate transformation via Eqs. 12.1–12.3, respectively (Chen et al. 2012):

$$\mathbf{d}_{i,1}^k = \prod_{p=1}^{n-1} \cos\left(\boldsymbol{\varphi}_{i,p}^k\right), \tag{12.1}$$

$$\mathbf{d}_{i,j}^k = \sin\left(\boldsymbol{\varphi}_{i,(j-1)}^k\right) \cdot \prod_{p=j}^{n-1} \cos\left(\boldsymbol{\varphi}_{i,p}^k\right), \quad \text{for } j = 2, \ldots, n-1, \tag{12.2}$$

$$\mathbf{d}_{i,n}^k = \sin\left(\boldsymbol{\varphi}_{i,(n-1)}^k\right). \tag{12.3}$$

- Producing phase: The searching process of producer $\mathbf{X}_p$ at the kth iteration samples three points randomly via Eq. 12.4 (He et al. 2009):

$$\begin{cases} \mathbf{X}_z = \mathbf{X}_p^k + r_1 \cdot l_{\max} \cdot \mathbf{D}_p^k\left(\boldsymbol{\varphi}^k\right) \\ \mathbf{X}_r = \mathbf{X}_p^k + r_1 \cdot l_{\max} \cdot \mathbf{D}_p^k\left(\boldsymbol{\varphi}^k + \mathbf{r}_2 \cdot \theta_{\max}/2\right), \\ \mathbf{X}_l = \mathbf{X}_p^k + r_1 \cdot l_{\max} \cdot \mathbf{D}_p^k\left(\boldsymbol{\varphi}^k - \mathbf{r}_2 \cdot \theta_{\max}/2\right) \end{cases} \tag{12.4}$$

  where $\mathbf{X}_p^k$ is the current position of pth individual in kth generation, $\mathbf{X}_z$, $\mathbf{X}_r$, and $\mathbf{X}_l$ are the positions which the pth individual found in zero degree, right and left direction of it, respectively, $r_1 \in R^1$ is a normally distributed random number with mean 0 and standard deviation 1, $\mathbf{r}_2 \in R^{n-1}$ is a uniformly distributed random sequence in the range $(0, 1)$, and $\theta_{\max}$ and $l_{\max}$ are the maximum pursuit angle and distance, respectively.

If the searching on three directions is ended, there are three states as follows (He et al. 2009):

When the new position has a better fitness value, the producer will move to the new point. The producer will keep its current position, however turn its head to a new angle via Eq. 12.5 (He et al. 2009):

$$\boldsymbol{\varphi}^{k+1} = \boldsymbol{\varphi}^{k} + \mathbf{r}_2 \cdot \alpha_{\max}, \tag{12.5}$$

where $\alpha_{\max} \in R^1$ is the maximum tuning angle.

When there is no better position can be found after α iterations, the producer will turn its head back to zero degree via Eq. 12.6 (He et al. 2009):

$$\boldsymbol{\varphi}^{k+\alpha} = \boldsymbol{\varphi}^{k}, \tag{12.6}$$

where $\alpha \in R^1$ is a pre-defined constant.

- Scrounging phase: After the determination of the producer, the scroungers will perform random walks by searching the opportunities to join the resources found by the producer via Eq. 12.7 (He et al. 2009):

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^{k} + \mathbf{r}_3 \cdot \left(\mathbf{X}_p^{k} + \mathbf{X}_i^{k}\right), \tag{12.7}$$

  where $\mathbf{r}_3$ is an uniform random sequence in the range $(0, 1)$, $\mathbf{X}_i^k$ and $\mathbf{X}_i^{k+1}$are the positions of ith scrounger in t and $t + 1$ iterations, respectively.
- Ranging phase: The inefficiency foragers will be selected as rangers that will perform a new searching process based on the random walks, i.e., generating a new random head angle (φ_i), choosing a random distance (l_i), and moving to the new position ($\mathbf{X}_i^{k+1}$), via Eqs. 12.8–12.10, respectively (He et al. 2009):

$$\boldsymbol{\varphi}^{k+1} = \boldsymbol{\varphi}_i^{k} + \mathbf{r}_2 \cdot \alpha_{\max}, \tag{12.8}$$

$$l_i = \alpha \cdot r_1 \cdot l_{\max}, \tag{12.9}$$

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^{k} + l_i \cdot \mathbf{D}_i^{k}\left(\boldsymbol{\varphi}^{k+1}\right), \tag{12.10}$$

  where $r_1 \in R^1$ is a normally distributed random number with mean 0 and standard deviation 1, $\mathbf{r}_2 \in R^{n-1}$ is a uniformly distributed random sequence in the range $(0, 1)$, $l_{\max}$ is the maximum pursuit distance, $\alpha_{\max} \in R^1$ is the maximum tuning angle.

Taking into account the key phases described above, the steps of implementing the GrSO algorithm can be summarized as follows (He et al. 2006, 2009):

- Step 1: Defining the optimization problem, and initializing the optimization parameters.
- Step 2: Repeat till stopping criteria met.
- Step 3: Choose a member as producer.

- Step 4: The producer performs producing.
- Step 5: Choose scroungers.
- Step 6: Scroungers perform scrounging.
- Step 7: Dispersed the rest members to perform ranging.
- Step 8: Evaluate members.
- Step 9: Check if maximum iteration is reached, go to Step 2 for new beginning, if a specified termination criteria is satisfied, stop and return the best solution.

12.2.2 Performance of GrSO

In order to show how the GrSO algorithm performs, the founders have conducted a set of studies to convince us. First, in He et al. (2006), four benchmark functions are studied. Second, in He et al. (2009), an intensive study based on a set of 23 benchmark functions are illustrated. For comparison purposes, several traditional computational intelligence (CI) methods are employed, namely genetic algorithm (GA), PSO, evolutionary programming (EP), fast EP (FEP), evolution strategies (ES), and fast ES (FES). Experimental results showed that GrSO outperforms others in solving multimodal functions, while performing a similar performance for unimodal functions in terms of accuracy and convergence rate.

12.3 Conclusions

Based on the producer-scrounger model, we introduced an interesting algorithm called GrSO in which three types of members are involved: producers (i.e., seeking food resources), scroungers (i.e., joining resources founded by the producer), and rangers (i.e., performing random walks from their current positions). For simplification, He et al. (2009) assumed that there is only one producer at each searching iteration and the remaining members are divided into scroungers and rangers, respectively. In addition, three mechanisms are employed to perform better searching strategies, i.e., environment scanning (i.e., vision) for producers, area copying for scroungers and random walks for rangers. Also, it is worth mentioning that the GrSO algorithm is capable of handling a variety of optimization problems, especially for the large scale optimization problems. The differences between GrSO and other algorithms (such as ACO, EA, and PSO) please refer to He et al. (2009) for more details. Although it is a newly introduced CI method, we have witnessed the following rapid spreading of GrSO:

First, several enhanced versions of GrSO can be found in the literature as outlined below:

- Fast GrSO (Zhan et al. 2011; Qin et al. 2009).
- GrSO with multiple producers (Guo et al. 2012).
- Hybrid GrSO and extreme learning machine (Silva et al. 2011a).

- Hybrid GrSO with metropolis rule (Fang et al. 2010).
- Improved GrSO (Xie et al. 2009; Shen et al. 2009; Silva et al. 2011b; Chen et al. 2012).
- Multiobjective GrSO (Wang et al. 2012).

Second, the GrSO algorithm has also been successfully applied to a variety of optimization problems as listed below:

- Artificial neural network training (He and Li 2008; Silva et al. 2011a, b).
- Mechanical design optimization (Shen et al. 2009).
- Power system optimization (Wu et al. 2008; Zhan et al. 2011; Kang et al. 2011, 2012; Guo et al. 2012; Liao et al. 2012).
- Truss structure design optimization (Liu et al. 2008; Xie et al. 2009).

Interested readers are referred to them as a starting point for a further exploration and exploitation of the GrSO algorithm.

References

Barnard, C. J., & Sibly, R. M. (1981). Producers and scroungers: a general model at its application to captive flocks of house sparrows. *Animal Behaviour, 29*, 543–550.

Biondolillo, K., Stamp, C., Woods, J., & Smith, R. (1997). Working and scrounging by zebra finches in an operant task. *Behavioural Processes, 39*, 263–269.

Brockmann, H. J., & Barnard, C. J. (1979). Kleptoparasitism in birds. *Animal Behaviour, 27*, 487–514.

Chen, D., Wang, J., Zou, F., Hou, W., & Zhao, C. (2012). An improved group search optimizer with operation of quantum-behaved swarm and its application. *Applied Soft Computing, 12*, 712–725.

Fang, J., Cui, Z., Cai, X., & Zeng, J. A. (2010, July 17–19). *Hybrid group search optimizer with metropolis rule. The 2010 International Conference on Modelling, Identification and Control (ICMIC)* pp. (556–561). Japan: IEEE.

Giraldeau, L.-A., & Lefebvre, L. (1987). Scrounging prevents cultural transmission of food-finding behaviour in pigeons. *Animal Behaviour, 35*, 387–394.

Guo, C. X., Zhan, J. P., & Wu, Q. H. (2012). Dynamic economic emission dispatch based on group search optimizer with multiple producers. *Electric Power Systems Research, 86*, 8–16.

He, S., & Li, X. (2008). *Application of a group search optimization based artificial neural network to machine condition monitoring. IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)* (pp. 1260–1266). IEEE.

He, S., Wu, Q. H., & Saunders, J. R. A. (2006, July 16–21). *Novel group search optimizer inspired by animal behavioural ecology. IEEE Congress on Evolutionary Computation (CEC) Sheraton Vancouver Wall Centre Hotel, Vancouver, BC*, (pp. 1272–1278). Canada: IEEE.

He, S., Wu, Q. H., & Saunders, J. R. (2009). Group search optimizer: an optimization algorithm inspired by animal searching behavior. *IEEE Transactions on Evolutionary Computation, 13*, 973–990.

Kang, Q., Lan, T., Yan, Y., An, J., & Wang, L. (2011). *Swarm-based optimal power flow considering generator fault in distribution systems. IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 786–790). IEEE.

Kang, Q., Lan, T., Yan, Y., Wang, L., & Wu, Q. (2012). Group search optimizer based optimal location and capacity of distributed generations. *Neurocomputing, 78*, 55–63.

Liao, H., Chen, H., Wu, Q., Bazargan, M., & Ji, Z. (2012). Group search optimizer for power system economic dispatch. In: Y. Tan, Y. Shi, & Z. Ji. (Eds.), ICSI 2012, Part I, LNCS 7331, (pp. 253–260). Berlin: Springer.

Liu, F., Xu, X.-T., Li, L.-J., & Wu, Q. H. (2008). *The group search optimizer and its application on truss structure design. Fourth International Conference on Natural Computation (ICNC)* (pp. 688–692). IEEE.

Millor, J., Amé, J. M., Halloy, J., & Deneubourg, J. L. (2006). Individual discrimination capability and collective decision-making. *Journal of Theoretical Biology, 239*, 313–323.

Mills, D. S., Marchant-Forde, J. N., McGreevy, P. D., Morton, D. B., Nicol, C. J., Phillips, C. J. C., et al. (Eds.). (2010). *The encyclopedia of applied animal behaviour and welfare*. UK: CAB International. ISBN 978-0-85199-724-7.

Parker, G. A. (1984). Evolutionarily stable strategies. In: J. R. Krebs, & N. B. Davies, (Eds.), *Behavioural ecology: an evolutionary approach.* Oxford: Blackwell Scientific.

Qin, G., Liu, F., & Li, L. (2009). *A quick group search optimizer with passive congregation and its convergence analysis. International Conference on Computational Intelligence and Security (CIS)* (pp. 249–253). IEEE.

Shen, H., Zhu, Y., Niu, B., & Wu, Q. H. (2009). An improved group search optimizer for mechanical design optimization problems. *Progress in Natural Science, 19*, 91–97.

Silva, D. N. G., Pacifico, L. D. S., & Ludermir, T. B. (2011a). *An evolutionary extreme learning machine based on group search optimization IEEE Congress on Evolutionary Computation (CEC)* (pp. 574–580). IEEE.

Silva, D. N. G., Pacifico, L. D. S., & Ludermir, T. B. (2011b). *Improved group search optimizer based on cooperation among groups for feedforward networks training with weight decay. IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 2133–2138). IEEE.

Vickery, W. L., Giraldeau, L.-A., Templeton, J. J., Kramer, D. L., & Chapman, C. A. (1991). Producer, scroungers, and group foraging. *The American Naturalist, 137*, 847–863.

Wang, L., Zhong, X., & Liu, M. (2012). A novel group search optimizer for multi-objective optimization. *Expert Systems with Applications, 39*, 2939–2946.

Wu, Q. H., Lu, Z., Li, M. S., & Ji, T. Y. (2008). *Optimal placement of FACTS devices by a group search optimizer with multiple producer. IEEE World Congress on Computational Intelligence (WCCI)* (pp. 1033–1039). IEEE.

Xie, H., Liu, F., & Li, L. (2009). *A topology optimization for truss based on improved group search optimizer. International Conference on Computational Intelligence and Security (CIS)* (pp. 244–148). IEEE.

Zhan, J. P., Yin, Y. J., Guo, C. X., & Wu, Q. H. (2011). *Integrated maintenance scheduling of generators and transmission lines based on fast group searching optimizer. IEEE Power and Energy Society General Meeting* (pp. 1–6). IEEE.