Software Impacts 19 (2024) 100630

Contents lists available at ScienceDirect

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

Original software publication

A QOBL-SAO and its variant: An open source software for optimizing PV/wind/battery system and CEC2020 real world problems

Abdullahi Abubakar Mas'ud [a,*], Ahmed T. Salawudeen [b], Abubakar A. Umar [c], Yusuf A. Shaaban [d], Firdaus Muhammad-Sukki [e,f,*], Umar Musa [c], Saud J. Alshammari [a]

[a] Department of Electrical Engineering, Jubail Industrial College, Royal Commission for Jubail, Saudi Arabia
[b] Department of Electrical and Electronic Engineering, University of Jos, Nigeria
[c] Department of Electrical Engineering, Ahmadu Bello University, Samaru, Zaria, Nigeria
[d] Department of Electrical Engineering, University of Hafr Batin, Saudi Arabia
[e] Engineering and the Built Environment, Edinburgh Napier University, Merchiston Campus, Edinburgh, UK
[f] Solar Research Institute, School of Electrical Engineering, College of Engineering, Universiti Teknologi MARA, Selangor, Malaysia

ARTICLE INFO

Keywords:
Smell agent optimization
Levy flight smell agent optimization
Quasi oppositional smell agent optimization

ABSTRACT

The Quasi oppositional smell agent optimization (QOBL-SAO) and its levy flight variant (LFQOBL-SAO) are two cutting-edge software tools for optimizing PV/wind/battery power systems. They can also be used to solve real-world CEC2020 optimization problems and are as good as top-performing software such as IUDE, ϵ MAgES and the iLSHAD ϵ. The QOBL-SAO exploits the random mode's weakness and then adds a number to the initial population. The LFQOBL-SAO, on the other hand, improves the random mode's weakness in order to solve this problem. The LFQOBL-SAO improves performance and search space by using levy flight instead of random code.

Code metadata

Current code version	*V1.0*
Permanent link to code/repository used for this code version	https://github.com/SoftwareImpacts/SIMPAC-2023-514
Permanent link to reproducible capsule	https://codeocean.com/capsule/1743513/tree/v1
Legal code license	*GNU*
Code versioning system used	*git*
Software code languages, tools and services used	*Matlab*
Compilation requirements, operating environments and dependencies	
If available, link to developer documentation/manual	
Support email for questions	atsalawudeen@unijos.edu.ng

1. Introduction

It was recently discovered that hybrid renewable energy systems (HRES), including photovoltaic (PV)/wind/battery systems, are the most cost-effective and viable options for electrifying off-grid locations. [1–4]. The economic and technical planning of the PV/wind/battery system design are complex for a number of reasons. The renewable energy sources unpredictability and dependence on weather are two of such reasons [5–7]. To meet energy demand, these systems are often oversized or undersized [8–10]. An oversized system wastes energy and has a high operating cost. Conversely, a microgrid that is undersized will not be able to provide the necessary amounts of electricity to the loads. A robust energy management plan must be paired with appropriate sizing for a wind, battery and PV power system to yield maximum benefits [11]. As a result, the LFQOBL-SAO and QOBL-SAO are novel software's developed intended to effectively

DOI of original article: https://doi.org/10.1016/j.asoc.2023.111081.
The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding authors.
E-mail addresses: masud_a@rcjy.edu.sa (A.A. Mas'ud), f.muhammadsukki@napier.ac.uk (F. Muhammad-Sukki).

https://doi.org/10.1016/j.simpa.2024.100630
Received 13 January 2024; Received in revised form 13 February 2024; Accepted 17 February 2024

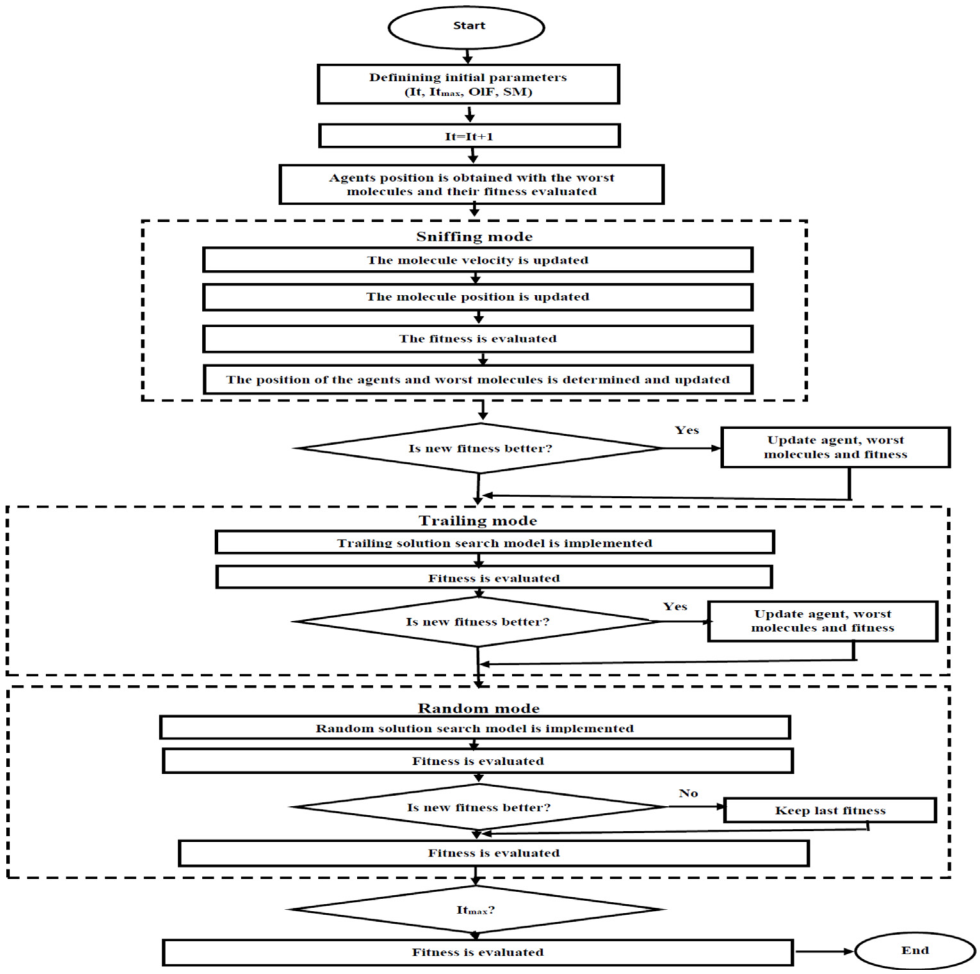

Fig. 1. 1 SAO software flowchart (*SM* = smell molecules, *FE* = number of function evaluations, Itr_{max} = maximum iteration and *Itr* = Iteration).

optimize HRES and have demonstrated faster convergence than the conventional SAO. They are as good as the best performing functions and particularly good at solving real-world CEC2020 optimization competition problems [12]. They can also reduce the net present cost (NPC) and the Lowest Cost of Energy (LCOE) of renewable energy projects.

2. Software description

The SAO is a member of the paradigm class of swarm intelligence [13]. The SAO optimization procedures were inspired by the notion that an agent can use chemosensory receptors in the olfactory organ to track a portion of a smell molecule [14]. This SAO has three distinct phases: trailing, random, and sniffing. The inspiration for the sniffing mode was derived from the hydrostatic gas theory, which posits that smell molecules disperse in the direction of the source. This evaporation is initiated by generating a random population of smell molecules. Evaporation results from the random generation of the original population of smell molecules in the trailing mode. All molecules generated in the previous mode have a chance of becoming smell agents in the trailing modes, depending on their position. In this particular scenario, while the agent is conducting a search in hyperspace, it is possible for the aggregation of smell molecules to rise above the designated location. To circumvent the issue of becoming trapped in local minima, the agent will ultimately transition to the random mode when the trailing mode fails to discover an optimal solution [13]. This study introduces two software applications, namely LFQOBL-SAO and QOBL-SAO, which are adaptations of the SAO software. Fig. 1 is the description of the conventional SAO.

2.1. QOBL-SAO and the LF-QOBL

The QOBL-SAO employs generation jumping and quasi-opposition-based initialization [15]. In this particular case, employing the antithetical points, also referred to as the antithetical population, produces more favorable initial conditions, even in the absence of prior knowledge regarding the solutions. The objective of the QOBL-SAO is to produce a population of olfactory agents that is the antithesis of the original population, with the intention of effectively broadening the scope of the search. As demonstrated in Algorithm 1, a function utilizing the QOBL-SAO pseudocode was developed in order to implement the QOBL-SAO. Among the three characteristics of SAO, the random mode exhibits a comparatively lower level of conceptual comprehension. The incorporation of a randomly generated number within the initial population does not yield a substantial impact on the algorithm's rate of convergence. The introduction of the LFQOBL-SAO aims to enhance the inherent limitations of the random mode, thereby providing a solution to this issue. Both search space and performance are enhanced by the LFQOBL-SAO [16]. When implementing the LFQOBL-SAO, care must be taken because an oversized step may result in a better solution that outperforms the true solution; on the other hand, a small step size may cause the convergence rate to drop and the algorithm to perform poorly. The LFQOBL-SAO algorithm integrates the levy flight function as a substitute for the random code. Algorithm 2 introduces the LFQOBL-SAO algorithm.

Eqs. (2.1)–(2.3) are used to compute the respective parameters of the LFQOBL-SAO pseudocode.

Algorithm 1: Pseudocode for the QOBL-SAO

```
Inputs: L, U, preliminary population (x), N and d
Output:
for i = 1:M
  | for j = 1:d
  |   x^o_{i,j} = L_j + U_j − x_{i,j} % creating the reverse of the present population
  |   D_{i,j} = (L_j + U_j)/2
  |   if (x_{i,j} < D_{i,j}) % generating quasi opposite of x
  |     | x^{qo}_{i,j} = D_{i,j} + (x^o_{i,j} − D_{i,j}) × rand
  |   else
  |     | x^{qo}_{i,j} = D_{i,j} + (D_{i,j} − x^o_{i,j}) × rand
  |   end
  | end
End
```

[M= number of molecules, L and U are the settings for the initial population's minimum and maximum values and d= variables dimension]

Algorithm 2: Pseudocode of the LFQOBL-SAO

```
Input: min τ, β , f(x) and σ_h
Output:
Select the population x_i to modify the position.
Compute σ_h update (from Equation 2.1)
while (τ <∈) do
  | Determine the step size (from equation 2.2)
  | Obtain New Solution x'_i (from equation 2.3)
  | Compute f(x'_i)
  | if f(x_i) > f(x_i') then
  |   | x_i = x'_i
  | end if
end while
```

[τ= is the step size]

$$\sigma_h = \left\{ \frac{\sin\left(\frac{\pi\beta}{2}\right)\Gamma(1+\beta)}{2^{\left(\frac{\beta-1}{2}\right)}\beta\Gamma\left(\frac{1+\beta}{2}\right)} \right\} \tag{2.1}$$

$$\text{Step\_size } (\tau) = \text{s}(\tau) \times 0.01; \tag{2.2}$$

$$x_i'(\tau+1) = x_i'(\tau) + \text{Step\_size } (\tau) \times \text{U}(0,1) \tag{2.3}$$

3. Software impacts

First, the LFQOBL-SAO and QOBL-SAO software demonstrate a high level of effectiveness in optimizing systems that combine wind and battery, as well as the PV/battery and PV/wind/battery configurations. The LFQOBL-SAO and QOBL-SAO outperformed the SAO in terms of convergence time, LCOE, and total annualized cost [17]. The most cost-effective configuration for the HRES system is achieved by utilizing the LFQOBL-SAO and QOBL-SAO and strategies. Second, both the LFQOBL-SAO and QOBL-SAO algorithms have demonstrated their ability to effectively address real-world optimization challenges and yield outcomes that are on par with leading software solutions such as IUDE, ϵ MAgES, and the iLSHAD ϵ [17]. These include the process synthesis problem [18], tension/compression spring design [19], weight minimization of a speed reducer [20], the design of gear train [21] and the three-bar trust design problem [22].

In this study, the capabilities of SAO, QOBL-SAO, and LFQOBL-SAO to address challenging problems derived from CEC 2020 [12], have been harnessed, encompassing the process synthesis problem, tension/compression spring design, weight minimization of a speed reducer, the design of a gear train, and the three-bar trust design problem. Users can seamlessly apply this software to tackle these specific CEC 2020 benchmark functions by configuring the algorithm parameters, problem constraints, and objectives pertinent to each case.

The software facilitates a friendly interface, which allows researchers and practitioners to input their problem specifications easily. The implementation intelligently explores the solution space, iteratively optimizing the design variables to reach optimal or near-optimal solutions for the stated problems. The impact of the software on the analysis lies in its ability to efficiently and effectively converge towards optimal solutions. These offer valuable insights into the design and synthesis challenges outlined in CEC 2020. Its adaptability ensures broad pertinence, making it a valuable asset for researchers seeking to address diverse optimization problems within the specified domains.

The LFQOBL-SAO and QOBL-SAO software applications are designed for the purpose of facilitating applied research conducted by researchers and scientists in the field of developing optimization strategies for microgrids that incorporate renewable energy sources. These software tools are capable of effectively addressing intricate optimization problems.

4. Future work and limitations

Future research efforts will be directed towards enhancing and optimizing the QOBL-SAO and LFQOBL-SAO algorithms. This can be achieved by integrating supplementary optimization techniques, employing novel hybrid methodologies, or incorporating a number of deep learning algorithms. The objective is to augment the efficacy of these algorithms. Further investigation can be conducted to assess the scalability of the QOBL-SAO and LFQOBL-SAO and algorithms in the context of engineering optimization problems and larger-scale HRES. The convergence behavior of the system can be analyzed when considering the increase in system size and complexity. This analysis incorporates a range of factors, such as the existence of diverse load conditions, dynamic operating states and multiple renewable sources. Additional areas of future research encompass conducting a sensitivity analysis on the LFQOBL-SAO and QOBL-SAO models in order to assess their resilience in the face of variations in input parameters, optimization objectives and system configurations. This analysis has the potential to offer valuable insights into the performance of algorithms in various scenarios and facilitate the identification of key factors that influence their effectiveness. A potential constraint of the current study is the absence of verification for the QOBL-SAO LFQOBL-SAO software through the utilization of case studies derived from HRES and real-world data. Moreover, it is crucial to conduct a comprehensive examination of policy and economic factors to assess the policy implications and financial viability linked to the adoption of optimized HRES. To fully assess the scheme's practical implications, a number of factors, such as regulatory frameworks, market dynamics, government incentives, and tariff structures, should be taken into consideration. These factors may provide insightful managerial information for the scheme's successful execution.

CRediT authorship contribution statement

Abdullahi Abubakar Mas'ud: Formal analysis, Conceptualization. **Ahmed T. Salawudeen:** Software, Methodology. **Abubakar A. Umar:** Writing – original draft. **Yusuf A. Shaaban:** Writing – review & editing, Writing – original draft. **Firdaus Muhammad-Sukki:** Writing – review & editing, Writing – original draft, Methodology. **Umar Musa:** Writing – review & editing, Writing – original draft. **Saud J. Alshammari:** Writing – review & editing, Writing – original draft, Visualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] G. Zhang, C. Xiao, N. Razmjooy, Optimal operational strategy of hybrid PV/wind renewable energy system using homer: A case study, 2021, http://dx.doi.org/10.1080/01430750.2020.1861087.

[2] O. Ekren, B.Y. Ekren, Size optimization of a PV/wind hybrid energy conversion system with battery storage using simulated annealing, Appl. Energy 87 (2010) 592–598, http://dx.doi.org/10.1016/J.APENERGY.2009.05.022.

[3] R.K. Rajkumar, V.K. Ramachandaramurthy, B.L. Yong, D.B. Chia, Techno-economical optimization of hybrid pv/wind/battery system using neuro-fuzzy, Energy 36 (2011) 5148–5153, http://dx.doi.org/10.1016/J.ENERGY.2011.06.017.

[4] A.A. Mas'ud, A hybrid combination of EV charging stations with renewable energy sources for powering a residential community. A techno-economic perspective, Energy Storage (2023) e483, http://dx.doi.org/10.1002/EST2.483.

[5] J. Li, W. Wei, J. Xiang, A simple sizing algorithm for stand-alone PV/Wind/Battery hybrid microgrids, Energies 5 (2012) 5307–5323, http://dx.doi.org/10.3390/EN5125307.

[6] F. Al-Turjman, Z. Qadir, M. Abujubbeh, C. Batunlu, Feasibility analysis of solar photovoltaic-wind hybrid energy system for household applications, Comput. Electr. Eng. (2020) 86, http://dx.doi.org/10.1016/J.COMPELECENG.2020.106743.

[7] A. Shaqour, H. Farzaneh, Y. Yoshida, T. Hinokuma, Power control and simulation of a building integrated stand-alone hybrid PV-wind-battery system in Kasuga city, Jpn. Energy Rep. 6 (2020) 1528–1544, http://dx.doi.org/10.1016/J.EGYR.2020.06.003.

[8] M.S. Ansari, M.F. Jalil, Manaullah, S. Diwania, Techno-Economic Analysis of Diesel/Wind/PV/Battery Hybrid Energy System for Androth Island, in: Lect Notes Electr Eng, 2021, http://dx.doi.org/10.1007/978-981-33-4080-0_27, 723 LNEE:285–95.

[9] S.O. Frimpong, I.E. Agbehadji, A. Abayomi, R.C. Millham, E. Freeman, M.M. Ujakpa, Economic and reliability determination of sustainable renewable energy mix based on social spider prey optimization algorithm, vol. 8, Springer Berlin Heidelberg, 2023, http://dx.doi.org/10.1007/s00500-023-08231-8.

[10] A. Bhimaraju, G.B. Ganesh, A. Mahesh, Optimal sizing of PV/Wind/Battery stand-alone hybrid renewable energy systems using TLBO algorithm, 2023, pp. 65–73, http://dx.doi.org/10.1007/978-981-19-9285-8_7.

[11] O. Nadjemi, T. Nacer, A. Hamidat, H. Salhi, Optimal hybrid PV/wind energy system sizing: Application of Cuckoo search algorithm for Algerian dairy farms, Renew. Sustain. Energy Rev. 70 (2017) 1352–1365, http://dx.doi.org/10.1016/J.RSER.2016.12.038.

[12] A. Kumar, G. Wu, M.Z. Ali, R. Mallipeddi, P.N. Suganthan, S. Das, A test-suite of non-convex constrained optimization problems from the real-world and some baseline results, Swarm Evol. Comput. (2020) 56, http://dx.doi.org/10.1016/J.SWEVO.2020.100693.

[13] A.T. Salawudeen, M.B. Mu'azu, Y.A. Sha'aban, A.E. Adedokun, A novel smell agent optimization (SAO): An extensive CEC study and engineering application, Knowl.-Based Syst. 232 (2021) 107486, http://dx.doi.org/10.1016/J.KNOSYS.2021.107486.

[14] A.T. Salawudeen, A.A. Olaniyan, G.A. Olarinoye, T.H. Sikiru, Formulation and optimization of overcurrent relay coordination in distribution networks using metaheuristic algorithms, Commun. Comput. Inf. Sci. 1350 (2021) 389–402, http://dx.doi.org/10.1007/978-3-030-69143-1_30.

[15] S. Nama, A.K. Saha, An ensemble symbiosis organisms search algorithm and its application to real world problems, Decis. Sci. Lett. 7 (2018) 103–118, http://dx.doi.org/10.5267/J.DSL.2017.6.006.

[16] K. Amalina Mohd Rosli, Z. Mustaffa, Y. Yusof, M. Farhan Mohamad Mohsin, Dengue outbreak prediction using an improved salp swarm algorithm, IOP Conf. Ser. Mater. Sci. Eng. (2020) 769, http://dx.doi.org/10.1088/1757-899X/769/1/012031.

[17] A.A. Mas'ud, A.T. Salawudeen, A.A. Umar, A.S. Aziz, Y.A. Shaaban, F. Muhammad-Sukki, et al., A quasi oppositional smell agent optimization and its levy flight variant: A PV/Wind/battery system optimization application, Appl. Soft Comput. 147 (2023) 110813, http://dx.doi.org/10.1016/J.ASOC.2023.110813.

[18] G.R. Kocis, I.E. Grossmann, A modelling and decomposition strategy for the minlp optimization of process flowsheets, Comput. Chem. Eng. 13 (1989) 797–819, http://dx.doi.org/10.1016/0098-1354(89)85053-7.

[19] A.D. Belegundu, J.S. Arora, A study of mathematical programming methods for structural optimization, part I: Theory, Int. J. Numer. Methods Eng. 21 (1985) 1583–1599, http://dx.doi.org/10.1002/NME.1620210904.

[20] S.H. Chew, Q. Zheng, Integral global optimization : theory, implementation, and applications, 1988, p. 179.

[21] E. Sandgren, Nonlinear integer and discrete programming for topological decision making in engineering design, J. Mech. Des. 112 (1990) 118–122, http://dx.doi.org/10.1115/1.2912568.

[22] Optimization in pre-contract ship design, 2022, https://trid.trb.org/view/14541. (Accessed 1 December 2022).