Expert Systems With Applications 119 (2019) 184–209

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Review

Symbiotic Organisms Search Algorithm: theory, recent advances and applications

Absalom E. Ezugwu[a,*], Doddy Prayogo[b]

[a] School of Computer Science, University of KwaZulu-Natal, King Edward Road, Pietermaritzburg Campus, Pietermaritzburg, KwaZulu-Natal 3201, South Africa

[b] Department of Civil Engineering, Petra Christian University, Jalan Siwalankerto 121-131, Surabaya, Indonesia

A R T I C L E I N F O

Article history:
Received 27 May 2018
Revised 31 August 2018
Accepted 26 October 2018
Available online 28 October 2018

Keywords:
Symbiotic organisms search algorithm
Swarm intelligence
Metaheuristic algorithms
Optimization

A B S T R A C T

The symbiotic organisms search algorithm is a very promising recent metaheuristic algorithm. It has received a plethora of attention from all areas of numerical optimization research, as well as engineering design practices. it has since undergone several modifications, either in the form of hybridization or as some other improved variants of the original algorithm. However, despite all the remarkable achievements and rapidly expanding body of literature regarding the symbiotic organisms search algorithm within its short appearance in the field of swarm intelligence optimization techniques, there has been no collective and comprehensive study on the success of the various implementations of this algorithm. As a way forward, this paper provides an overview of the research conducted on symbiotic organisms search algorithms from inception to the time of writing, in the form of details of various application scenarios with variants and hybrid implementations, and suggestions for future research directions.

1. Introduction

Several research contributions in the field of metaheuristics have produced numerous optimization methods (Weise, 2011; Yang, 2010c; Yang, 2013; Yang and He, 2013), which are more powerful than the traditional gradient-based (Haupt, 1995) or hessian matrix based methods (Tahk, Park, Woo, & Kim, 2009). While these contributions can currently be considered optimum (Nowak and Cirpka (2004)), most real-world problems are highly unpredictable and are often accompanied with numerous uncertainties, making the search for optimal solutions even more complex. Therefore, it may be said that while optimality is not the only focus of optimization, it also includes robustness, which is a key factor when dealing with most scientific and engineering problems. As an alternative to the classical optimization methods, scientists have succeeded in developing and using the nature inspired metaheuristic algorithms to circumvent the limitations of the traditional optimization techniques (Blum & Roli, 2003; Weise, 2011; Yang, 2010c). Metaheuristic algorithms are well known for their speed, robustness and near-optimal solution accuracy. Moreover, it is very common to use metaheuristic algorithms to efficiently and effectively solve most of the complex optimization problems that are difficult to solve by using the gradient descent algorithms (Clerc & Kennedy, 2002; Das, Mukhopadhyay, Roy, Abraham, & Panigrahi, 2011; Yang & Deb, 2010).

The term 'metaheuristic' which connotes 'higher level' is often seen as an iterative generation process that integrates different concepts for *exploring* and *exploiting* the search space to guide a subordinate heuristic 'lower level', with learning strategies used to structure information to find efficiently near-optimal solutions (Osman & Laporte, 1996; Yang, 2014). In general, while searching for a global optimum solution, the efficiency of the search process of most metaheuristic algorithms, to some extent, greatly depends on their ability to strike a balance between exploration and exploitation. According to Tashkova, Šilc, Atanasova, and Džeroski (2012) and Fister, Fister, Yang, and Brest (2013) the exploration process guides each metaheuristic in discovering the diverse solutions within the search space. In other words, the exploration process assists in enabling the metaheuristic to attain the new searching region of the search space. It helps to prevent the premature convergence into local optimum; however, it will slow the convergence rate. The exploitation process enables the metaheuristic to focus the search process within the neighborhoods of the current best solutions. It allows convergence rapidly to the optimum solution in the region, with a consequence of being trapped in local optimum prematurely. (Interested readers may

* Corresponding author.
E-mail addresses: Ezugwua@ukzn.ac.za (A.E. Ezugwu), prayogo@petra.ac.id (D. Prayogo).

https://doi.org/10.1016/j.eswa.2018.10.045
0957-4174/

consult Črepinšek, Mernik, and Liu (2011) and Črepinšek, Liu, and Mernik (2013) for more information about studies on exploration and exploitation in evolutionary algorithms).

It is noteworthy to mention here that despite the proliferations in the number of new metaheuristic algorithms, nearly all of them share the same common characteristics (Boussaïd, Lepagnot, & Siarry, 2013; Ishibuchi et al. 2003; Cheng & Prayogo, 2014). Some of the shared characteristics include the following: they all draw their inspirations from nature, they have parameters that need to be fine-tuned to make them adaptable for the problem at hand, and they do not require substantial gradient information. However, each metaheuristic possesses some unique advantages in terms of speed, accuracy, robustness and overall performances in different problem spaces. In addition, studies have shown that there is no single algorithm that is good enough to solve all the optimization problems with their unique problem features (Wolpert & Macready, 1997). Therefore, there is a need for a continuous search for the development of new metaheuristic algorithms to handle different specific optimization problems that emerge.

Some of the notable examples of metaheuristic algorithms that are widely used to solve various optimization problems are the following: Variable Neighborhood Search (VNS) (Mladenović & Hansen, 1997), Tabu Search (TS) (Glover, 1989), Simulated Annealing (SA) (Kirkpatrick, Gelatt, & Vecchi, 1983), Particle Swarm Optimization (PSO) (Eberhart & Kennedy, 1995), Ant Colony Optimization (ACO) (Dorigo & Di Caro, 1999), Differential Evolution (DE) (Storn & Price, 1997), Cuckoo Search (CS) algorithm (Yang & Deb, 2009), Fire Fly Algorithm (FA) (Yang, 2010b), Flower Pollination Algorithm (FPA) (Yang, 2012), Invasive Weed Optimization (IWO) algorithm (Mehrabian & Lucas, 2006), Bat Algorithm (BA) (Yang, 2010a), Fruit Fly Optimization algorithm (FOA) (Pan, 2012), Artificial Bee Colony (ABC) algorithm (Teodorovic & Dell'Orco, 2005), Bacterial Foraging (BF) algorithm (Passino, 2002), Krill Herd (KH) algorithm (Gandomi & Alavi, 2012), Bees Algorithm (BeeA) (Pham et al., 2011), Particle Bee Algorithm (PBA) (Cheng & Lien, 2012), Cat Swarm Optimization (CSO) algorithm (Chu, Tsai, & Pan, 2006), Glowwarm Swarm Optimization (GSO) (Krishnanand & Ghose, 2009) and Symbiotic Organisms Search (SOS) algorithm (Cheng & Prayogo, 2014). (Interested readers may also consult the work of Fister et al. (2013) for more details on some of the classifications of the nature inspired algorithms).

Among the notable and newly developed metaheuristic algorithms, the SOS algorithm has received significantly wider attention from the optimization research community and other related domains, as a result of the algorithm's implementation simplicity and stability, that results from its parameter-less nature. In addition, the algorithm has equally witnessed a drastic increase in its area of application to several optimization domains as compared to other alternative algorithms or competitors, which include GA, ACO, PSO, and DE. Therefore, this paper presents a comprehensive review of the standard SOS algorithm, its basic concepts and structures, variants and hybrid implementations for handling constrained, unconstrained, single objective, multi-objective, large scale global optimization problems as well as practical oriented real world optimization problems. The main contribution of this paper consists of: -

- an exhaustive study of the classical SOS algorithm;
- a systematic review of up to date SOS variants and hybrid algorithms;
- an identification and presentation of the various application areas of SOS algorithm in solving both complex and non-linear scientific and engineering design problems;
- suggestions for further development and possible areas of improvement; and
- recommendations for novel application areas for the SOS algorithm.

The paper is structured as follows: The introduction is followed by Section 2 presenting background information including conceptual features and structures of the classical SOS algorithm. Section 3 presents a systematic review of studies on SOS algorithms covering variants and hybrid implementations of the standard algorithm, while in Section 4 the various applications of the SOS in different domains are discussed. Section 5 offers a discussion about the performance validation of the standard SOS algorithm. In Section 6, a discussion about further development and possible improvement on the standard SOS algorithm is presented. Then Section 7 covers critical remarks as well as future perspectives about the symbiotic organisms search algorithm. Lastly, a conclusion is given in Section 8.

2. Symbiotic Organisms Search Algorithm

The symbiotic organisms search (SOS) algorithm was first introduced by Cheng and Prayogo (2014). The standard algorithm is simple and powerful, and employs the perspective of a population-based search strategy by guiding a population of candidate solutions to search for promising optimum region iteratively until a global optimum solution to a given objective function is found. However, it was originally designed for dealing with numerical optimization problems in a continuous solution space, even though it has undergone several transformations that have made it more robust and adaptive to other problem spaces. SOS is inspired by the relationship among different organisms which live together in one ecosystem and continuously strive and compete for survival or to grow together. This phenomenon is referred to as 'symbiosis' in biology.

2.1. Symbiosis relationships

Symbiosis is a phenomenon used to describe the kind of adaptive relationships or interactions that often exist between any two distinctive organisms or species that cohabit in the natural ecosystem. This form of cohabitation between two different organisms is further subdivided into obligate and facultative types of relationships (Cheng & Prayogo, 2014). The obligate symbiosis defines the type of relationship between two different species of organisms that are inclusively dependent on each other for their survival, while the facultative variety of symbiosis defines a relationship method between two different species of organisms that decide to cohabit in a mutually beneficial but nonessential relationship (Cheng & Prayogo, 2014).

The three fundamental symbiotic relationships types found among organisms include mutualism, commensalism and parasitism. These three modes of interactive interrelationships are described as the relationships where two living organisms benefit reciprocally, is known as mutualism. One common example of mutualism is the interaction between the Oxpecker birds and Rhinoceros. The Oxpecker lives on the rhino, sustaining itself by eating the bugs and parasites on the animal and the Rhinoceros get pest control. Commensalism symbiosis is the relationship between two organisms where one of the organisms obtains all the benefits from the other organism while the other is not affected by the immediate interaction. An example of the commensalism is the interaction between cattle egrets (birds that live near grazing cattle) that eat insects that have been disturbed by cattle foraging for food. However, if the benefits obtained by one organism from another is causing the detriment of the other organism, the relationship is referred as parasitism symbiosis. An example in this regard is mosquitoes that rely on human blood to produce their

Fig. 1. Three phases of symbiosis among organisms in the ecosystems: from left - commensalism (Oxpeckers on the back of a Rhinoceros), mutualism (Cattle egrets and Cattle), and parasitism (Mosquito feeding on Human blood).

eggs. This relationship is considered parasitism because the human (host) is affected negatively and the mosquito (parasite) benefits from the relationship. Fig. 1 illustrates the three types of symbiotic relationships that inspired the development of the symbiotic organisms search algorithm.

2.2. Symbiotic organisms search algorithm encoding

The SOS algorithm encoding was modeled based on the three fundamental relationships structures described in Section 2.1, namely mutualism, commensalism and parasitism. These natural phenomena illustrated in Fig. 1 can be associated with the specific problem objective function to be optimized or solved. As such, the SOS algorithm is encoded using the three aforementioned relationship processes. The formulation of the standard SOS is outlined as shown in Algorithm listing 1.

The SOS algorithm engages through the iterative steps highlighted in Algorithm 1, starting first with the ecosystem initialization, which is usually expressed as $X=\{X_1,X_2,\ldots, X_{ecosize}\}$. More so, as discussed in (Ezugwu, Adeleke, & Viriri, 2018), the population size of the organisms can be set within the range of a moderate population size of 25, an average population size of 50, and a large population size of 100. Subsequently, the algorithm generates new organisms position by computing and comparing their respective objective function values (Ezugwu et al., 2018), so that the organism with the best objective value is selected as X_{best}. The algorithmic process is repeated by updating the current best solution until the organism with the global best solution is discovered. The algorithm execution is terminated when the maximum number of fitness function evaluation is reached. Otherwise, the algorithm continues to evaluate by exploring and exploiting other new possible solution search spaces. It is noteworthy to mention here that the stopping condition is quite an important factor in determining the final result of the entire optimization process. For example, reducing the length of the algorithm execution time, might not give solutions that are close to the targeted global optimum and prolonging the execution time might as well increase the computational cost of the algorithm (Ezugwu et al., 2018). Detailed descriptions of the SOS algorithm phases are presented next in Sections 2.2.1–2.2.3.

2.2.1. Mutualism phase

In the mutualism phase, for each organism X_i, an organism X_j is randomly selected from the ecosystem to interact with X_i ($where\ X_i \neq X_j$) on the bases of establishing a relationship that is mutual. However, the association between X_i and X_j is to increase the mutual survival rate of the two organisms within the ecosystem. The new candidate solutions X_{inew} and X_{jnew} are generated using Eq. (1) and (2), respectively:

$$X_{inew} = X_i + rand(0,1) \times (X_{best} - X_{mutual} \times BF_1) \quad (1)$$

$$X_{jnew} = X_j + rand(0,1) \times (X_{best} - X_{mutual} \times BF_2) \quad (2)$$

where X_{mutual} is represented by using the expression given in Eq. (3).

$$X_{mutual} = \frac{X_i + X_j}{2} \quad (3)$$

$$\mathrm{BF}_1 = (1 + \mathrm{round}(\mathrm{rand}(0,1))), \ |rand \in [0,1] \quad (4)$$

$$BF_2 = (1 + round(rand(0,1))), \ |rand \in [0,1] \quad (5)$$

The $rand(0,1)$ function is a vector of uniformly distributed random numbers between 0 and 1. The organism with the best objective or fitness function value in terms of the degree of adaptation in the ecosystem, represented by X_{best}, while X_{mutual} signifies a mutualistic characteristic exhibited between the two organisms to increase their survival advantage Ezugwu & Adewumi, 2017b). The values of the benefit factors BF_1 and BF_2 are determined randomly using Eqs (4) and ((5). These factors represent the level of benefit from the interaction for each organism. It should be noted that an update is only made, if the new computed fitness function value denoted by $f(X_{inew})$ and $f(X_{jnew})$, is better than the previous fitness functions, $f(X_i)$ and $f(X_j)$, (Ezugwu & Adewumi, 2017b). Thus, the two equations, which are equation *1* and *2*, can further be transformed as follows:

$$X_{inew} = X_i + rand(0,1) \times (X_{best} - X_{mutual} \times BF_1), \quad if\ f(X_{inew}) > f(X_i) \quad (6)$$

$$X_{jnew} = X_j + rand(0,1) \times (X_{best} - X_{mutual} \times BF_2), \quad if\ f(X_{jnew}) > f(X_j) \quad (7)$$

Fig. 2 illustrates the visualization pattern of the SOS mutualism phase in the problem search space.

2.2.2. Commensalism phase

In the commensalism phase, an organism X_j is selected randomly from the ecosystem to interact with the second organism X_i. However, the organism X_i strives to increase its benefits from the association with X_j, while X_j neither benefits nor suffers from such relationship. For this type of relationship, the organism X_i is placed at an advantageous position over X_j; however, the organism X_j is not harmed in the process (Ezugwu, Adewumi, & Frîncu, 2017). The new solution that emerges as a result of this symbiotic relationship is updated using the following expression as shown in *equation 8* (Ezugwu & Adewumi, 2017b).

$$X_{inew} = X_i + rand(-1,1) \times (X_{best} - X_j), \ if\ f(X_{inew}) > f(X_i) \quad (8)$$

The $rand(1,1)$ function is a vector of uniformly distributed random numbers between -1 and 1. The new organism X_{inew} replaces the organism X_i if its fitness value is better. The expression $(X_{best} - X_j)$ denotes in this case the beneficial advantages provided by the organism X_j to assist the organism X_i increase its survival advantage among the population in the ecosystem to the highest

Algorithm 1 Standard SOS pseudocode.

Input: ecosize: population size; UB: upper bound of the search space
D: dimensions of problem; LB: lower bound of the search space
maxFE: maximum function evaluation; F(X): objective function
Output: X_{best} which is the final best solution of the population

Generate an initial population $X=\{X_1, X_2, \ldots, X_{ecosize}\}$, and evaluate its fitness, assign $FE \leftarrow 0$;
Identify the best solution of the initial population X_{best}
while FE < maxFE
for i = 1 to ecosize **do**
/* **Mutualism Phase** */
Choose randomly index j $\in \{1, 2, \ldots, ecosize\}$, which j $\neq$ i
$BF_1 = (1 + round(rand(0, 1))$
$BF_2 = (1 + round(rand(0, 1))$
$X_{mutual} = (\frac{X_i + X_j}{2})$
for k = 1 to D **do**
$X_{inew} = X_i + rand(0, 1)^*(X_{best} - BF_1{}^*X_{mutual})$
$X_{jnew} = X_j + rand(0, 1)^*(X_{best} - BF_2{}^*X_{mutual})$
end for
if $F(X_{inew}) < F(X_i)$
$X_i = X_{inew}$
end if
if $F(X_{jnew}) < F(X_j)$
$X_j = X_{jnew}$
end if
FE = FE + 2 /* Increase the number of function evaluation counter */
Commensalism Phase
Choose randomly index j $\in \{1, 2, \ldots, ecosize\}$, which j $\neq$ i;
for k = 1 to D **do**
$X_{inew} = X_i + rand(-1, 1)^*(X_{best} - X_j)$
end for
if $F(X_{inew}) < F(X_i)$
$X_i = X_{inew}$
end if
FE = FE + 1 /*Increase the number of function evaluation counter
Parasitism Phase
Choose randomly index j $\in \{1, 2, \ldots, ecosize\}$, which j $\neq$ i
for k = 1 to D **do**
if rand(0, 1) < rand(0, 1)
$X_{parasite} = X_i$
else
$X_{parasite} = rand(0, 1)^*(UB[k] - LB) + LB$
end if
end for
if $F(X_{parasite}) < F(X_j)$
$X_j = X_{parasite}$
end if
FE = FE + 1 /* Increase the number of function evaluation counter/
Update the best solution of the current population X_{best}
end for
end while

degree in X_{best} (the most current or updated organism). Fig. 3 illustrates the visualization pattern of the SOS commensalism phase in the problem search space.

2.2.3. Parasitism phase

The parasitism phase involves an association between two organisms, for which one of the organism derives all the benefit by harming the partner organism. An example of parasitism is the association that exists amongst three organisms: the Plasmodium parasite, Anopheles mosquito and the human host. In this type of relationship, the human host is harmed, the Anopheles mosquito, which is the parasite carrier is left unharmed, while the Plasmodium parasite thrives and reproduces inside the human body (Ezugwu et al., 2017). Therefore, by mimicking the aforementioned parasitic behaviors, an organism X_i is assigned a role similar to that of the Anopheles mosquito through the creation of an artificial vector $X_{parasite}$ in the solution search space, by fine-tuning the randomly selected dimension of organism X_i (Ezugwu & Adewumi, 2017b). Subsequently, another organism X_j is selected randomly from the ecosystem and serves as a host to $X_{parasite}$. Then, $X_{parasite}$ will try to replace X_j in the ecosystem. However, if $X_{parasite}$ turns out to have a better fitness value than X_j, then X_j is replaced by $X_{parasite}$. Otherwise, X_j develops an immunity from $X_{parasite}$, which will invariably cease to exist in the ecosystem (Ezugwu et al., 2017). The artificial parasite vector is computed as follows:

$$X_{parasite} = rand(0,1) * (UB - LB) + LB \tag{9}$$

where LB(lower bound) and UB(upper bound) are the respective boundary limits of the problem to be evaluated. In Fig. 4, an illustration of the visualization pattern for the SOS mutualism phase in the problem search space is presented.

It is noteworthy to mention here that most of the improvements done on the standard SOS algorithm were carried out by modifying either the mutualism or commensalism phase or both. Only in some very rare cases is another new phase introduced to the existing three phases. In Section 3, some of the recent improvements and hybridization implementations of SOS algorithms that exist in the literature are discussed in detail.

2.3. Characteristics of symbiotic organisms search algorithm

Several characteristics and advantages of the SOS algorithm are highlighted in comparison with other related population-based al-

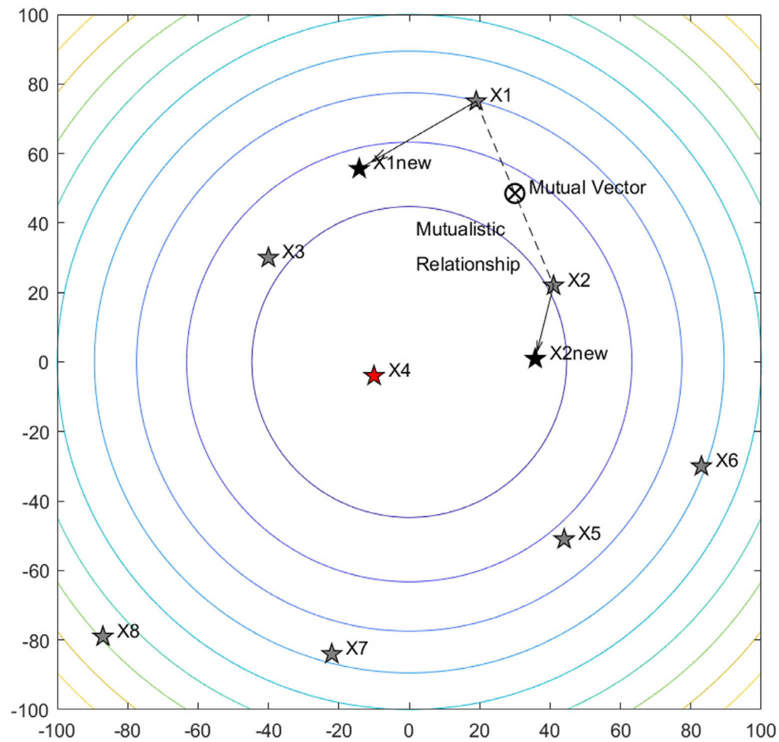

Fig. 2. Visualization of mutualism in the search space.

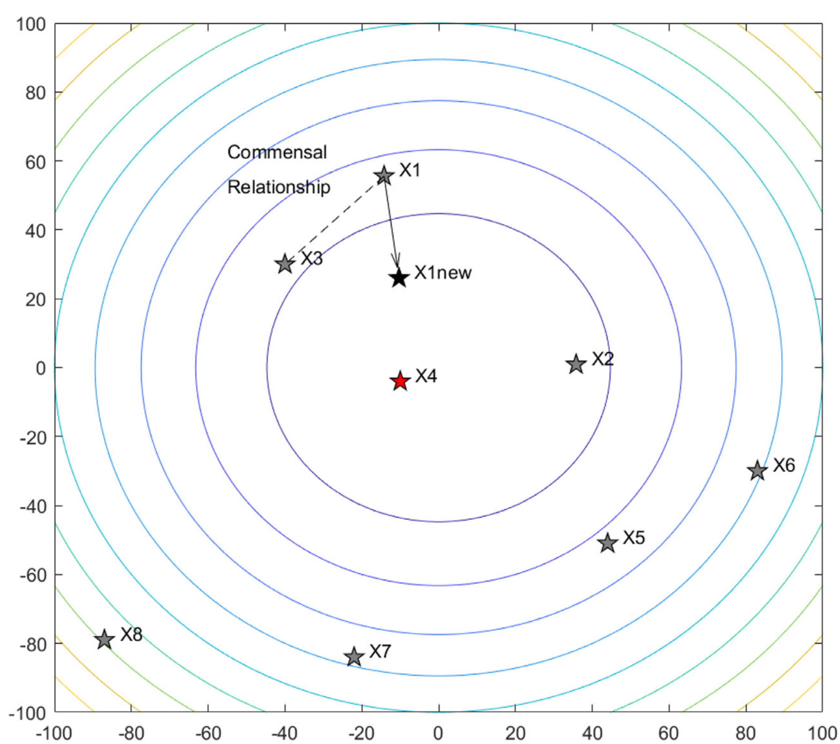

Fig. 3. Visualization of commensalism in the search space.

gorithms such as GA, DE, PSO, BA, and CS. Based on the work of Cheng and Prayogo (2014), these characteristics and advantages are summarized as follows:

- SOS, being a population based algorithm, shares similar characteristics with other related population based approaches.
- To find an optimum solution, SOS uses an iterative technique which is performed on a group of candidates' solutions.
- SOS adaptability greatly depends on the level of modification in the three interaction strategies, namely, commensalism, mutualism and parasitism. However, while the mutualism strategy might be unique to the SOS, the commensalism strategy is a common interaction feature that is also shared by other related algorithms such as the PSO, DE and CS, although there might exist some slight differences in application. For example, SOS

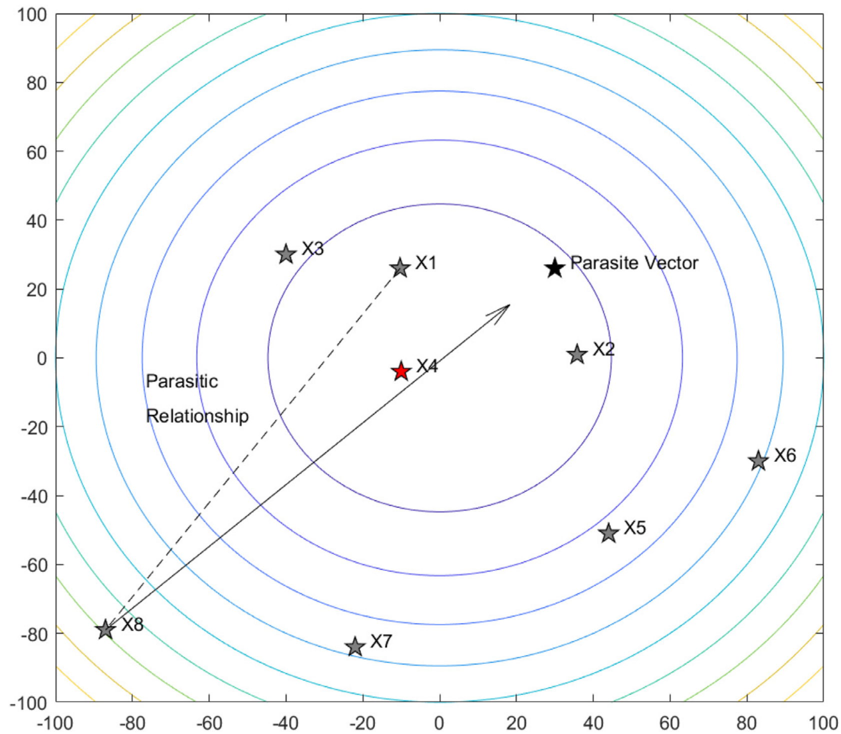

Fig. 4. Parasitism between Anopheles mosquito and human.

uses the best solution as the reference point to exploit promising regions that are closer to the best solution during the commensalism update phase. Another unique strategy employed by the SOS is the parasitism phase, which is a mutation operator that introduces perturbation to the ecosystem to maintain diversity and prevent premature convergence.

- At the end of each update phase of the SOS algorithm, the new organisms X_{inew} and X_{jnew} replace the original organisms X_i and X_i if and only if, they give better objective values and a greedy selection strategy is used by SOS. Some population based algorithms also use this selection mechanism such as the DE algorithm.

2.4. Advantages of symbiotic organisms search algorithm

Having considered some of the most special features and qualities the SOS algorithm, the focus now falls on presenting the specific reasons why SOS should be chosen as a target algorithm for solving various hard real-world optimization problems. The reasons why SOS should be chosen as a target algorithm include the following:

- SOS is a parameter free algorithm excluding the general parameters that are commonly found among the population-based algorithms including population size, maximum number of iteration and problem dimension. Therefore, it does not require any parameter fine tuning or adjustments.
- SOS has good exploitation capabilities with the processes of mutualism and commensalism. SOS uses best solution as a reference point that might help in exploiting the neighborhood solutions of the current best solution.
- SOS has exploration capability by incorporating cloning and mutation in the parasitism phase.
- SOS has the capability to eliminate inferior solutions in the parasitism phase. Very few metaheuristic algorithms possess all four properties listed above.

2.5. Comparison with other algorithms

In Table 1, an enumerative comparison between SOS and other popular metaheuristics algorithms is presented. Most metaheuristic algorithms share some common characteristics in terms of their method of generating initial solutions. These include a population-based, versus a single point search or trajectory; memory usage against memory-less methods; single neighborhood versus multiple neighborhood structures; and dynamic, against a static objective function (Beheshti & Shamsuddin, 2013; Birattari, Paquete, Stützle, & Varrentrapp, 2001; Vaessens, Aarts, & Lenstra, 1998). However, for a more equitable comparison between SOS and other similar algorithms, the comparison presented here is limited to the most important characteristics associated with most popular metaheuristics in terms of their method of generating initial solution “first row”, population-based search or trajectory “second row, memory utilization “third row”, multiple neighborhood structures “fourth row”, parameter-tuning “fifth row” and dynamic objective function “sixth row”. Mathematical notations have used to denote if a particular feature is present, partially present, or is not present in each of the algorithms.

3. Studies on Symbiotic Organisms Search Algorithms

In this Section a systematic review of the various improvement efforts made on the standard SOS and its hybridization with other state-of-the-art problem-solving approaches, is presented. Recent studies from numerous publications on the applications of the SOS algorithm showed that few variants and several hybrid implementations of the SOS algorithm exist (Do and Lee, 2017; Nama, Saha, & Ghosh, 2017; Panda & Pani, 2018).

3.1. Variants and modified symbiotic organisms search algorithms

The standard SOS was initially designed for solving numerical optimization problems. Just as in the other metaheuristics before SOS, the performance of the algorithm was tested using similar

Table 1
Comparison of SOS with selected popular metaheuristic algorithms.

	Popular metaheuristic algorithms												
Features	SOS	GA	DE	PSO	ACO	CS	FA	FPA	BA	ABC	VNS	SA	TS
Random search	√	√	√	√	√	√	√	√	√	√	√	∃	∃
Population based search	√	∄	∄	√	√	√	√	√	√	√	√	∃	∃
Memory	√	∃	√	√	√	√	√	√	√	√	∃	∄	∃
Multiple neighborhood	√	∃	∄	√	∄	√	√	√	√	√	√	∄	∄
Parameter tuning	∄	√	√	√	√	√	√	√	√	√	∄	∃	∄
Dynamic objective function	√	√	√	√	√	√	√	√	√	√	∃	∄	∃

√ implies that the compared characteristic is present.
∃ implies that the compared characteristic is partially present.
∄ implies that the compared characteristic is not present.

standard numerical optimization benchmark functions. However, in order to make SOS suitable for problems with different problem characteristics or structures, some versions of the SOS algorithm were developed. These versions include the discrete, adaptive, improved or modified, and multi-objective SOS algorithms. A brief discussion of each aforementioned version citing examples from the literature body of knowledge is presented.

3.1.1. Discrete symbiotic organisms search algorithm

Researchers have implemented several versions of the basic SOS algorithm, including the discrete SOS (DSOS) (Cheng, Prayogo, & Tran, 2015). The latter has been applied to solve several classes of optimization and real-world problems. Examples describing the successful implementation of the DSOS include the work of Cheng et al. (2015), Ezugwu and Adewumi (2017) and Sharma and Verma (2017), Talatahari (2015). The DSOS algorithm was first formulated by Cheng et al. (2015) and applied to solve scheduling problems of multiple projects. Resource leveling is used in project scheduling to reduce fluctuations in resource usage over the period of project implementation (Cheng et al. (2015). The new DSOS basically operates by transforming continuous solutions into discrete solutions. This is usually achieved by developing a function that is able to convert the real-value variables which the DSOS operates on, into integer values that are constrained in the feasible domain. In conducting this research, the following example was developed to illustrate how this conversion can be achieved by first generating an initial ecosystem population comprising of feasible solutions:

$$X = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_i \\ \vdots \\ X_{ecosize} \end{bmatrix} = \begin{bmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,D} \\ x_{2,1} & x_{2,2} & \dots & x_{2,D} \\ \vdots & \vdots & \vdots & \vdots \\ x_{i,1} & x_{i,2} & x_{i,j} & x_{i,D} \\ \vdots & \vdots & \vdots & \vdots \\ x_{ecosize,1} & x_{ecosize,2} & \dots & x_{ecosize,D} \end{bmatrix} \quad (10)$$

where in Eq (10), the variable $x_{i,j} \in [0, 1]$ is optimized by SOS during the search process and D is the number of decision variables in the problem under consideration. The decision variables for the problem under consideration were represented in the form of a vector and were originally given as real values, expressed as follows:

$$X_{i,j} = \left[x_{i,1},\ x_{i,2}, \dots,\ x_{i,j}, \dots,\ x_{i,D}\right] \quad (11)$$

The transformation of these variables from their real-value forms into integer values is achieved using the function given in Eq (12) as follows:

$$X_{i,j} = round\left\{LB(j) + x_{i,j} \times \ [UP(j) - LB(j)]\right\} \quad (12)$$

where $X_{i,j}$ is defined as the starting time of noncritical activity j, $UP(j)$ and $LB(j)$ defined as the early and late starting times for the activity j.

The results obtained by the DSOS algorithm were compared with the results of other existing methods namely, GA, PSO and DE. This comparison showed that the proposed DSOS performed better than the compared methods when used for the same purpose.

Similarly, in Ezugwu and Adewumi (2017), a discrete version of the SOS was proposed to find the best routes for several traveling salesman problems (TSPs). A DSOS algorithm is presented in this paper to solve the well-known symmetric TSP. In addition, some improvements in the basic structure of the standard SOS algorithm are proposed to make SOS suitable for solving the TSP efficiently. The results of the proposed DSOS for TSP were compared with other state-of-the-art algorithms and it showed that the DSOS outperforms the compared methods in terms of quality of solutions, convergence speed and execution time.

Another implementation of the discrete SOS is found in the work undertaken by Talatahari (2015) that uses the *Fix* function to transform each of the three basic SOS update phases into integer variables. The *Fix* function for example in Matlab, rounds each element of the real variable X off to the nearest integer towards zero. The mode of conversion for the SOS mutualism, commensalism, and parasitism phases adopted in this paper are as follows:

DSOS-mutualism phase:

$$X_{inew} = \ Fix\{X_i + r_1 \times (X_{best} - MV \times BF_1)\} \quad (13)$$

$$X_{jnew} = Fix\left\{X_j + r_2 \times (X_{best} - MV \times BF_2)\right\} \quad (14)$$

DSOS-commensalism phase:

$$X_{inew} = \ Fix\left\{X_i + (2r_1 - 1) \ \times \left(X_{best} - X_j\right)\right\} \quad (15)$$

DSOS-parasitism phase:

$$X_{ijnew} = \ Fix\left\{x_{jmin} + r_1 \times \left(x_{jmax} - x_{jmin}\right)\right\} \quad (16)$$

where r_1,r_1 $\in$ [0, 1], MV denotes the mutual vector and BF_1 and BF_2 denote benefit factors.

The main goal of compiling this paper was to find the optimum design of frame structures and grillage systems. The empirical analysis of the proposed DSOS algorithm was performed by using the DSOS to solve some structural engineering design problems. The results obtained were compared with those of the charged system search (CSS), improved harmony search (HS), imperialist competitive algorithm (ICA), and accelerated particle swarm optimization (APSO). The results of the comparisons show the validity in the superiority of the proposed DSOS over existing methods.

3.1.2. Adaptive symbiotic organisms search algorithm

In the work of Tejani, Savsani, and Patel (2016), an adaptive symbiotic organisms search (SOS) algorithm for structural design optimization problems is presented. In this work, three modified versions of the SOS algorithm are proposed by combining the existing SOS benefit factors (presented in Eqs (4) and (5) with their new proposed adaptive benefit factors ($ABF1$ and $ABF2$) in the standard SOS algorithm, to improve its performance efficiency. The goal of the adaptive SOS was to improve the algorithm's capability to strike a good balance between exploration and exploitation of the solution search spaces using the expression given in Eqs (17) and (18).

$$ABF1 = \frac{f(X_i)}{f(X_{best})},\ if\ f(X_{best}) \neq 0 \tag{17}$$

$$ABF2 = \frac{f\left(X_j\right)}{f(X_{best})},\ if\ f(X_{best}) \neq 0 \tag{18}$$

As mentioned earlier, the objective of introducing these two benefit factors, that is ABF1 and ABF2, into the SOS is to further strengthen its exploration capability. This exploration search strategy is necessary and is required more specifically, when the organism X_i or X_j is far from the best organism, whereas the good exploitation strategy employed by ABF1 or ABF2, is necessary when the organism X_i or X_j is near to the best organism within the solution search spaces. It is noteworthy to mention here, that as much as the exploration and exploitation search mechanisms set by either ABF1 or ABF2 are significant for improving the efficiency of the SOS, it is equally very important to maintain a balance between these two strategies for SOS to retain a more superior capability in finding a global optimum solution. It is noted that a pure exploration reduces the precision of the algorithm, whereas pure exploitation moves the algorithm to a local optimal solution. To check the feasibility and effectiveness of their methods, the algorithms were tested on different structural engineering problems. Their results showed that the adaptive SOS algorithm outperformed the standard SOS algorithm.

3.1.3. Improved or modified symbiotic organisms search algorithm

Another effort to improve the basic SOS algorithm is found in the work of Nama *et al.* (2016). In their study the authors present an improved version of the basic SOS algorithm referred to as the I-SOS algorithm, by introducing a random weighted reflective parameter and predation phase into the basic SOS algorithm in other to enhance the performance of the algorithm (Nama *et al.*, 2016). The addition of the fourth predation update phase beside the mutualism, commensalism and parasitism update phases common to the basic SOS, was inspired by the predation strategy often adopted by the same organisms that live in the ecosystem. In ecology, predation is said to be a biological interaction similar to the parasitism relationship, where a predator (an organism that is searching) feeds on its prey (the organism that is attacked) (Nama *et al.*, 2016). The similarity between the two mechanisms is that in both relationships, one organism is harmed and the other benefits. However, the difference lies in the fact that in predation one organism - the predator - would most likely kill and eat its prey, while in parasitism, not all parasites kill their hosts. The new solution for the predation update phase is generated using the expression given in Eq. (19).

$$Predation\_Vector = X_i + rand(0,1) \times \left(X_i^{max} - X_i^{min}\right) \tag{19}$$

where X_i^{min} and X_i^{max} represent the minimum and maximum numbers of the dimensions of the organism X_i. The fitness value of the organism, which is illustrated in the superiority of the predator or prey, is such that the worst organisms in the population are replaced by predation vector.

In addition to the formulation of the new I-SOS algorithm, a random weighted reflective parameter was also introduced in the basic SOS to formulate new sets of mutualism and commensalism update phases. The random weighted reflective vector RWRV was formulated as follows by using the expression given in Eq. (20):

$$RWRV = 0.5 + (1 + rand(1, D)) \tag{20}$$

This now transforms the SOS's mutualism and commensalism phases, respectively, into the following expression I-SOS update phases:

I-SOS mutualism phases

$$X_{inew} = X_i + RWRV \times (X_{best} - Mutual\_Vector \times BF_1) \tag{21}$$

$$X_{jnew} = X_j + RWRV \times (X_{best} - Mutual\_Vector \times BF_2) \tag{22}$$

I-SOS commensalism phase:

$$X_{inew} = X_i + RWRV \times \left(X_{best} - X_j\right) \tag{23}$$

This I-SOS algorithm was tested on some unconstrained mathematical benchmarked function and real world problems, such as the RWP.1 Gas transmission compressor design problem and RWP.2 Optimal capacity of gas production facilities. The experimental results of I-SOS when compared with other state-of-the-art algorithms showed that the proposed improved I-SOS algorithm is superior to all the other methods.

There are some other new improvements or variants of the SOS, which have been developed by different researchers (Al-Sharhan & Omran, 2018; Guha, Roy, & Banerjee, 2018; Saha & Mukherjee, 2018), but for reasons of space, this cannot be discussed here in detail. However, in Table 2, a concise summary of some selected variants of the SOS algorithms are presented according to the modifications, reference, SOS parameter, test benchmark function, objective description, benchmarked algorithms, journal /conference, publisher, and remarks.

3.2. Hybrid symbiotic organisms search algorithms

As described above, the basic SOS algorithm and its various modified versions have successfully been used to solve many continuous and discrete optimization problems. However, there are some exceptions of other optimization areas were these sets of algorithms did not perform very well or were not successful in finding the desired solutions. Therefore, in order to address this challenge, researchers are often compelled to combine multiple algorithms to find improved solutions to their problems. Studies have also shown that the hybrid algorithms are more likely to be more robust and produce better results than the classical approaches (Ezugwu et al., 2017; Oh, Lee, & Moon, 2004). The purpose of the hybridization method is to exploit the complimentary advantages and value-added information found in some algorithms. Some recent research efforts on the application of SOS to solve complex optimization and real-world problems, have shown that the classical version of the SOS algorithm still requires some level of improvement to achieve better performance (Al-Sharhan & Omran, 2018; Ezugwu et al., 2017; Saha & Mukherjee, 2018). The newly developed hybrid implementations of the SOS algorithm include the works of Abdullahi and Ngadi (2016), Nama et al. (2017), and Ezugwu et al. (2017). The achievements of some of these hybrid implementations are briefly discussed.

Abdullahi and Ngadi (2016) proposed a hybrid SA based symbiotic organisms search (SOSSA) with the aim of optimizing the process of task scheduling in cloud computing environments. In addition, the authors considered the improvement of the classical SOS algorithm based convergence rate and quality of solutions produced by the SOS. Abdullahi and Ngadi (2016) also incorporated into their work, a new objective function which takes into account the utilization level of virtual machines, makespan minimization and the degree of imbalance among virtual machines. The numerical results obtained as a result of their experiments showed

Table 2
Summary of literature review on variants and modified SOS algorithms.

SN	Modification	Reference	SOS parameter	Test Benchmark function	Objective description	Benchmarked algorithm	Journals /conference	Publisher	Remark
1	Standard SOS	Cheng and Prayogo (2014)	Ecosystem size = 50, maximum function evaluations = 500,000, run times=30	Twenty-six unconstrained mathematical problems and four structural engineering design problems	Numerical optimization and minimization of engineering design problem	GA, DE, PSO, BeeA and PBA	Computers and Structures	Elsevier	The main goal and contribution of this paper was to design and apply a new metaheuristic algorithm (SOS) to solve structural engineering optimization problems such as cantilever, I-beam vertical deflection, 15-bar planar truss structure, and 52-bar planar truss structure optimization problems.
2	Improved SOS	Nama et al. (2016)	ecosize = 50, maximum function evaluations = 500,000, run times=30	Global numerical optimization problems	Global numerical optimization problems	PSO, DE, SOS and DE variants	Decision Science Letters	Growing Science	The improved I-SOS algorithm was shown to have outperformed the basic SOS using similar parameters settings and with applications to some real world problems.
3	Adaptive SOS	Tejani et al. (2016)	Ecosystem size = 20, maximum function evaluations = 4000, run times=100	Six different planar and space trusses problems	Structural optimization with frequency constraints	OC, GA, FA, CBO, CS, NHGA, PSO, DPSO, NHPGA, CSS, enhanced CSS, HS, FA, CSS-BBBC and hybrid OC-GA	Journal of Computational Design and Engineering	Elsevier	The paper develops an adaptive SOS algorithm with the aim of minimizing truss weight by finding the optimal nodal positions and optimal elemental cross-sectional areas such that it satisfies multiple natural frequency constraints.
4	Enhanced SOS	Al-Sharhan and Omran (2018)	Ecosystem size = 25, 50, 75, 100, maximum function evaluations = 100,000, run times=30	26 benchmark problems and 6 engineering design optimization problems from the structural engineering field	Numerical optimization	SOS, DE, PSO and BSA	Neural Computing and Applications	Springer	This paper investigates the performance of SOS algorithms using a set of unbiased and characteristically different problems. The effect of ecosize, SOS' only parameter was equally investigated and the use of different random distributions also explored. Three variants of SOS' were proposed and their results compared with basic SOS.
5	Chaos-integrated SOS	Saha and Mukherjee (2018)	Ecosystem size = 50, maximum function evaluations = 500,000, run times=100	26 unconstrained benchmark problems	Optimization of siting and sizing problem of distributed generators in radial distribution system	SOS, PSO, GA, TLBO, GA-PSO and QOTLBO	Soft Computing	Springer	This article presents a chaotic SOS (CSOS) algorithm for improving the solution accuracy and convergence speed of the standard SOS algorithm. The proposed method was successfully implemented and tested on several unconstrained benchmarked functions and other real-world problems.

(*continued on next page*)

Table 2 (*continued*)

SN	Modification	Reference	SOS parameter	Test Benchmark function	Objective description	Benchmarked algorithm	Journals /conference	Publisher	Remark
6	Quasi-oppositional SOS	Guha et al. (2018)	Ecosystem size = 60, maximum number of iterations = 100	two-area interconnected power system with nonlinearity effect of governor dead band and generation rate constraint, four-area power system showing the consequence of load perturbation	obtain an optimum and effective solution for the load frequency control problem	hBFOA-PSO, GA, BFOA, PSO and TLBO	Swarm and Evolutionary Computation	Elsevier	The article presents an improved SOS algorithm. In this paper, basic SOS algorithm is integrated with the theory of quasi-oppositional based learning to find an optimal and effective solution for load frequency control problem of the power system.
7	Discrete SOS	Ezugwu and Adewumi (2017a)	Ecosystem size = 50, Maximum number of iterations = 1000, run times = 20	set of benchmarks of symmetric TSP instances selected from the TSPLIB library	finding a near optimal solution for the travelling salesman problem	SOS, ACS + 2-Opt, DPSO, DIWO and PSO-ACO-3-opt	Expert Systems With Applications	Elsevier	The discrete symbiotic organisms search algorithm has been shown to outperform the basic SOS for all the TSP benchmark problems tested. In addition, the optimal solutions obtained by the DSOS appear to be far better than the best solutions obtained by the compared state-of-the-art algorithms.
8	Orthogonal parallel SOS	Panda and Pani (2018)	Ecosystem size = 50, maximum number of iterations = 200, run times=20	Twelve benchmark nonlinear constrained problems and four engineering design problems	To solve the constrained optimization problems	OSOS MABC ALPSO PGA	Soft Computing	Springer	This paper proposed the design of an Orthogonal Parallel SOS (OPSOS) for solving constrained optimization problems. The results obtained by the proposed method are more superior to other compared methods.

Note: optimality criterion (OC), niche genetic hybrid algorithm (NGHA), charged system search (CSS), hybridized CSS and big bang-big crunch (CSS-BBBC), harmony search (HS) and firefly algorithm (FA), democratic PSO (DPSO), and colliding-bodies optimization (CBO), teaching–learning-based optimization (TLBO), quasi-oppositional teaching learning based optimization (QOTLBO), Original Symbiotic Organism Search (OSOS), augmented Lagrangian-based PSO (ALPSO), Penalty function-based GA (PGA), Modified artificial bee colony (MABC).

that the proposed hybrid method outperforms the classical SOS algorithm.

In a study conducted by Nama et al. (2017), a hybrid symbiotic organisms search algorithm (HSOS) and its application to solve some real-world problems, were discussed in detail with the proposed HSOS tested on varying benchmark functions. In the proposed HSOS method, the Simple Quadratic Interpolation (SQI) operator is combined with a standard SOS algorithm, thereby leveraging the exploration capability of the SQI and the exploitation potential of the SOS to increase the robustness of the proposed HSOS algorithm. In their implementation, the combination of the two approaches improved the searching capability of HSOS for attaining the global optimum (Nama et al., 2017). Thirteen well known benchmark functions namely, CEC2005 and CEC2010 were used to validate the performance of proposed method. The superiority of the HSOS over some existing state-of-the-art metaheuristic algorithms, namely DE, PSO and several variants of these two algorithms was also demonstrated.

The application of a hybrid SOS algorithm was suggested by Ezugwu et al. (2017), whereby a new hybrid algorithm that combines SOS with SA to solve the well-known traveling sales- man problems (TSPs) was proposed. The TSP is known to be NP-hard, and consists of a set of (n −1)! / 2 feasible solutions. The paper analysed the convergence behavior, average execution time and scalability of the hybrid algorithm to solve different problem scales of the TSP (Ezugwu et al., 2017). The authors evaluated the performance of the hybrid and standard SOS algorithms on different sets of TSP benchmarks obtained from TSPLIB (a library containing samples of TSP instances). The results of the empirical analysis showed that the hybrid SOS is able to find optimal solutions in some cases and near optimal solutions in most case with a low convergence rate.

Vincent, Redi, Yang, Ruskartina, and Santosa (2017) proposed two classes of solution representations (SR-1 and SR-2) for transforming the basic SOS algorithm into an applicable solution approach for CVRP. The hybrid process involves applying a local search strategy to improve the solution quality of the standard SOS algorithm. Their paper also proposed an additional improvement on the three updated interaction phases of the SOS (mutualism, commensalism, and parasitism,), by introducing two new update phases namely, competition and amensalism. In their work, amensalism was defined as an interaction whereby an organism inflicts harm upon another organism without receiving any costs or benefits (Vincent et al., 2017). An example was given of an instance where sheep or cattle trample on grass during grazing, the grass causes negligible harm to the hoof of the animal, while the grass suffers from being trampled upon by being crushed. In the case of competition, the interaction between the two organisms causes harm to each of the organisms. In a summary description of the paper's contribution, six version of SOSs were developed for solving CVRP. The performance of the hybrid SOSs was tested on classical CVRP benchmark instances. The experimental results showed that the proposed methods were able to produce good quality solutions in reasonable computational time. This is an indication that each of them is a highly competitive alternative algorithm for solving the capacitated vehicle routing problem. A concise summary of the hybrid implementations of the basic SOS with other algorithms or optimization methods as recorded in the literature is presented in Table 3.

3.3. Multi-objective symbiotic organisms search algorithms

The multi-objective optimization problems are often classified as multi-criteria decision making, which comprises of more than one objective functions to be optimized simultaneously. These classes of optimization methods are also known as multi-objective programming, vector optimization, multi-criteria optimization, multi-attribute optimization or Pareto optimization (Deb, 2001; Panda & Pani, 2016). Mathematically, a multi-objective problem is expressed as follows:

$$\begin{aligned} &\mathit{Optimize}\ \vec{f}(\vec{x}) \\ &\text{where} \\ &\vec{f}(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \ldots, f_i(\vec{x}), \ldots f_m(\vec{x})]^T \\ &s.t.\ x \in X, \end{aligned} \tag{24}$$

where $f(.)$ is the vector function to be optimized, the set X is the feasible set of vectors of decision variables. The integer $m \geq 2$ is the number of objectives and the function $f_i(.)$ Comprises of m objective functions such that

$$f_i : X \to R^m,\ \ \forall i \in [1, 2, \ldots, m] \tag{25}$$

In this case $\mathrm{R^m}$ is given here as the problem objective space.

There are a number of contributions of SOS algorithms to multi-objective optimization problems, which include the research input from Cheng and Prayogo (2015) proposing a multi-objective discrete symbiotic organisms search algorithm (DSOS) for optimizing multi-resource leveling in multiple projects scheduling to reduce fluctuation in resource usage over the period of project implementation. The multi-objective DSOS algorithm capability was tested in a case study which was adopted from the literature by Guo, Li, and Ye (2009). The experimental results from the work of Cheng and Prayogo (2015) were compared with other optimization methods namely, PSO, DE, and GA, and statistical tests was conducted to determine the robustness of the proposed methods. The statistical results showed that the multi-objective DSOS is an efficient multi-objective optimizer with capability of producing superior solutions that are better than the competing approaches.

In the work presented by Tran, Cheng, and Prayogo (2016), a novel multiple objective symbiotic organisms search approach was introduced to solve the multiple work shifts problems for a construction project. The main goal of the multi-objective SOS algorithm was to enhance the overall success of construction project by implementing a tradeoff to handle project schedules, while maintaining resource availability constraints. This algorithm was tested and analyzed using a numerical case study of a construction project to demonstrate its effectiveness for the time cost utilization labor tradeoff (TCUT) problem. The results were compared to other state-of-the-art algorithms such as the non-dominated sorting genetic algorithm II (NSGA-II), the multiple objective particle swarm optimization (MOPSO), the multiple objective differential evolution (MODE), and the multiple objective artificial bee colony (MOABC). The results presented in Tran et al. (2016) showed that the multi-objective SOS algorithm significantly outperformed these other methods.

Panda and Pani (2016) developed an improved symbiotic organisms search algorithm using adaptive as penalty function to handle equality and inequality constrains associated with multi-objective optimization problems. The study considered problems associated with the presence of local optima in complex problems with specific requirements to simultaneously optimize two or more functions that include certain constrains, which the conventional derivative based algorithms are not able to handle. Extensive simulation studies were conducted on twelve unconstrained and six constrained benchmarked multi-objective functions with an additional two constrained truss design engineering application problems. The results obtained with this method, were compared with those of the multi-objective colliding bodies optimization, multi-objective particle swarm optimization, non-dominated sorting genetic algorithm II, and two gradient based multi-objective algorithms, which include a multi-gradient explorer and multi-gradient pathfinder. According to the numerical results obtained, the au-

Table 3
Summary of a literature review on hybrid SOS algorithms.

SN	Reference	SOS parameter	Test Benchmark function	Objective function	Benchmarked algorithm	Journals /conference	Publishers	Remark
1	Abdullahi and Ngadi (2016)	Number of Organisms =100, number of Iterations = 1,000	NASA Ames iPSC/860 and HPC2N datasets	Makespan minimization, optimization of tasks scheduling and minimizing the degree of imbalance among virtual machines	SOS	PloS one	PLOS	The proposed hybrid algorithm (SOSSA) has been shown to outperform the existing standard SOS algorithm in terms of tasks scheduling in cloud computing environment.
2	Nama et al. (2017)	Population size =50, maximum function evalua-tions = 200,000, run times=25	CEC2005 and CEC2010 benchmarks	Enhance accuracy and convergence speed	DE, PSO	Memetic Computing	Springer	This paper improved the standard SOS algorithm by enhancing its accuracy and convergence rate.
3	Wu et al. (2018)	NA	Small-scale and large-scale knapsack problem instances	Optimization of feasible solution	NA	International Journal of Bio-Inspired Computation	Inderscience	In this article, a greedy strategy is employed by combining SOS and harmony search with a greedy strategy to repair the infeasible solution and optimize the feasible solution.
4	Ezugwu et al. (2017)b)	ecosize = 50, maximum iteration = 1000	40 TSPLIB	Finding the best TSP tour routes and convergence speed minimization	GA-PSO-ACO, MSA-IBS, ASA-GS, LBSA and IBA	Expert Systems with Applications	Elsevier	This study demonstrates the use of hybrid SOS-SA to solve TSP and the method was able to produce better solution and convergence in minimum time.
5	Vincent et al. (2017)	ecosize: 25, 50, 75, maximum number of iteration = 500, 1000, 1500, parasite force (pf): 0.7, 0.8, 0.9, run times=30	CVRP benchmark instances	Decide the routes for a set of vehicles to serve a set of demand points while minimizing the total routing cost	PSO	Applied Soft Computing	Elsevier	This article develops a framework that apply two solution representations namely SR-1 and SR-2, to transform the classical SOS into an applicable solution approach for CVRP and then combine the improved SOS with a local search strategy to improve the solution quality of SOS.
6	Wu et al. (2016)	ecosize = 30, maximum iteration = 500, run times=20	Eight different datasets selected from the UCI machine learning repository	Training feedforward neural networks	BBO, CS, GA, GSA, PSO and MVO,	Computational intelligence and neuroscience	Hindawi	The purpose of the paper is to present a satisfactory and efficient training algorithm for feedforward neural networks (FNN) using SOS. The SOS-FNN results performed better than other algorithms in terms convergence speed and accuracy.

Note: High-Performance Computing Center North (HPC2N), National Aeronautics and Space Administration (NASA), library containing samples of TSP instances (TSPLIB), Genetic Algorithm Particle Swarm Optimization Ant Colony Optimization (GA-PSO-ACO), Adaptive Simulated Annealing Algorithm with Greedy Search (ASA- GS), Multi-agent Simulated Annealing Algorithm with Instance-Based Sampling (MSA-IBS), List-Based Simulated Annealing (LBSA), and Improved Discrete Bat algorithm (IBA), capacitated vehicle routing problem (CVRP), University of California, Irvine (UCI).

Table 4
List of published work on multi-objective SOS algorithm.

SN	Problem	Reference	Journal /conference	Publisher
1	Multiple projects Scheduling problem	Cheng and Prayogo (2015)	Journal of Computing in Civil Engineering	American Society of Civil Engineers (ASCE)
2	Time–cost–labor utilization tradeoff problem	Tran et al. (2016)	Knowledge-Based Systems	Elsevier
3	Multi-objective constrained optimization problems	Panda and Pani (2016)	Applied Soft Computing	Elsevier
4	Electromagnetic Optimization	Ayala et al. (2017)	IEEE Transactions on Magnetics	IEEE

thors concluded that their algorithm outperformed all the other techniques (Panda & Pani, 2016).

In addition to the included list of multi-objective SOS algorithms compiled from research contained in the literature, Ayala, Klein, Mariani, and Coelho (2017) proposed a new multi-objective SOS algorithm based on non-dominance and crowding distance criteria. Ayala et al. (2017) also investigated a possible improvement strategy by incorporating the normal (Gaussian) probability distribution function into the basic SOS algorithm. In the test evaluation carried out for the two proposed methods in the paper, the brushless DC motor optimization problem was used to validate the effectiveness of these algorithms. A comparison with the NSGA-II was also conducted to further investigate the better performance of the introduced methods over existing algorithms in solving multi-objective electromagnetic optimization. A summary of journal articles on multi-objective SOS algorithm implementations are presented in Table 4.

4. Application of Symbiotic Organisms Search Algorithm

This section introduces the application of the basic, variants, and hybrid SOS algorithms to different research domains and their applications to address some specific real-world problems. Since its introduction as a metaheuristic algorithm in 2014, the SOS algorithm has received wider acceptance from the optimization and metaheuristic research communities. This is observed from the number current research outputs published in reputable journals and conference proceedings and the list keeps on extending every day (as supported by the list of references at the end of this paper). Therefore, it is pertinent to present a timely review of the growing application areas of the SOS algorithm to inspire researchers to further improve the performance of the SOS and as well as investigate other possible areas of application that could be exploited. In discussing the application of the SOS algorithm, this section is divided into two subsections to cover the discussions on the applications of SOS in engineering design problem and general optimization application problems.

4.1. Application to engineering design problems

The SOS algorithm was originally derived from the pursuit of researchers trying to find alternative and better solution approaches to the various gradient-based optimization methods used for solving different engineering design optimization problems. The primary and initial design objectives of the SOS algorithm were to solve numerical optimization and engineering design problems (Cheng & Prayogo, 2014). In this section, a list of engineering application areas of the SOS algorithm including its variants and hybrid implementations of the basic SOS are presented. Table 5 lists a number of construction engineering domains in which SOS has been successfully applied, while Table 6 shows the application of SOS to the areas of power or energy optimization engineering problems. Table 6 shows that the power engineering application has the largest list of SOS contributions based on the number of listed publications.

The wide recognition and acceptance of the SOS algorithm as an alternative metaheuristic algorithm with high capability and robustness, which is evident in numerous experimental evaluations done by addressing different engineering design problems and in some cases real-world problems too, specifically demonstrate the algorithm's superiority over other existing state-of-the-art metaheuristic algorithms. In addition, SOS as a newcomer in the domain of metaheuristics has been shown to be relevant in this area of research, with new articles emerging from day to day across all the different fields of engineering applications.

4.2. General application of symbiotic organisms search algorithm

This subsection presents a brief review of several applications of SOS algorithm in all major areas of research, based on existing published research. The research domains concerned were categorized based on the available articles used in the course of this study. The current review exercise showed that the categories of application areas that have employed the SOS algorithm and its variants are machine learning, distribution systems and cloud computing. Presented is a brief discussion of some application areas followed by a summary table listing relevant articles published in different journals.

In the work of Wu, Zhou, Luo, and Basset (2016) SOS was employed as training algorithm in supervised learning for the training of feedforward neural networks. In the training process, an SOS algorithm was employed to simultaneously determine the set of weights and biases required to minimize the overall error of one feedforward neural network and its corresponding accuracy, by training the network. Therefore, such effort is necessary to remove any imposed limitations often associated with the use of backpropagation algorithms and to subsequently improve convergence speed and accuracy when training feedforward neural networks. The effectiveness of the proposed method was tested on eight different standard datasets selected from the UCI machine learning repository (Wu et al., 2016). The results showed that the SOS-based training method has good convergence speed and accuracy. In addition, the obtained results from Wu et al. (2016) were compared with the results of six other metaheuristic algorithms, which include biogeography-based optimizer (BBO), cuckoo search (CS), GA, gravitational search algorithm (GSA), PSO, and multiverse optimizer (MVO) that were tested on the same problems. The above mentioned results showed that the SOS-based training for the feedforward neural networks was better in terms of converging speed.

Nanda and Jonwal (2017) developed new equalizers based on a nonlinear neural structure, referred to as a 'wavelet neural network' and subsequently employed the SOS metaheuristic algorithm to train its estimated weights. It turned out that the weakness of these equalizers is that during training, the mean square error is easily trapped into the local minimum and therefore, the estimated weights would fail to reach their optimum values. The existing training techniques were applied using the gradient based approach, which is known to have the aforementioned weakness.

Table 5
Construction engineering applications of SOS algorithm.

SN	Problem	Reference	Journal /conference	Publisher
1	Civil engineering optimization	Prayogo, Cheng, and Prayogo (2017)	Civil Engineering Dimension	Petra Christian University
2	Risk evaluation and maintenance strategies for bridge life cycle	Cheng et al. (2014)	Journal of the Chinese Institute of Civil and Hydraulic Engineering	The Chinese Institute of Civil and Hydraulic Engineering
3	Frame and grillage systems designs	Talatahari (2015)	Asian Journal of Civil Engineering	Springer
4	Operation of reservoir systems	Bozorg-Haddad et al. (2017)	Journal of Hydroinformatics	International Water Association Publishing (IWA)
5	Pin-jointed structures	Do and Lee (2017)	Applied Soft Computing	Elsevier

Table 6
Power/energy optimization with SOS algorithm.

SN	Problem	Reference	Journal /conference	Publisher
1	Congestion management in deregulated environment	Verma et al. (2017)	Journal of Experimental & Theoretical Artificial Intelligence	Taylor & Francis
2	Dynamic economic dispatch with valve-point effects	Sonmez et al. (2017)	Experimental & Theoretical Artificial Intelligence	Taylor & Francis
3	Short-term hydrothermal scheduling	Das and Bhattacharya (2016)	Ain Shams Engineering Journal	Elsevier
4	Optimal power flow of power system with FACTS devices	Prasad and Mukherjee (2016)	Engineering Science and Technology, an International Journal	Elsevier
5	Design of linear antenna arrays with low side lobes level	Dib (2016)	Progress In Electromagnetics Research B	Electromagnetics Academy
6	Optimization of Renewable Energy Sources	Kalkhambkar et al. (2017)	Journal of Electrical Systems	Engineering and Scientific Research Groups (ESR Groups)
7	SVC Installation in Voltage Control	Zamani et al. (2017)	Indonesian Journal of Electrical Engineering and Computer Science	Institute of Advanced Engineering and Science (IAES)
8	Optimal power flow problem based on valve-point effect and prohibited zones	Duman (2017)	Neural Computing and Applications	Springer
9	Economic/emission dispatch problem in power systems	Dosoglu et al. (2018)	Neural Computing and Applications	Springer
10	Minimizing Energy of Point Charges on a Sphere	Kanimozhi et al. (2016)	International Journal on Electrical Engineering and Informatics	The School of Electrical Engineering and Informatics, Institut Teknologi Bandung
11	Optimal Fuzzified Joint Reconfiguration and Capacitor Placement in Electric Distribution Systems	Sedighizadeh et al. (2017)	INAE Letters	Springer
12	Optimal Size and Siting of Distributed Generators in Distribution Systems	Nguyen et al. (2017)	International Journal of Energy Optimization and Engineering	IGI Global
13	Large scale economic dispatch problem with valve-point effects	Secui (2016).	Energy	Elsevier
14	Power optimization of three dimensional turbo code	Banerjee and Chattopadhyay (2017).	Wireless Personal Communications	Springer
15	Concurrent optimal design of TCSC and PSS	Alomoush (2017)	Turkish Journal of Electrical Engineering & Computer Sciences	Scientific and Technical research Council of Turkey - TUBITAK/Turkiye Bilimsel ve Teknik Arastirma Kurumu
16	Automatic generation control of interconnected power systems including wind farms	Hasanien and El-Fergany (2016)	IET Generation, Transmission & Distribution	Institution of Engineering and Technology
17	Optimal Distributed Generation and Capacitor Placement for Loss Minimization and Voltage Profile Improvement	Lalitha et al. (2016)	International Journal of Electrical Engineering	International Research Publication House
18	Optimal coordination of directional overcurrent relays in power systems	Saha, Datta, and Das (2016)	IET Generation, Transmission & Distribution	Institution of Engineering and Technology (IET)
19	Optimal Reactive Power Dispatch	Prasad and Mukherjee (2018)	IETE Journal of Research	Taylor & Francis
20	Load frequency control of multi-area power system	Guha et al. (2018)	Energy Systems	Springer
21	Design of planar concentric circular antenna arrays with reduced side lobe level	Dib (2017)	Neural Computing and Applications	Springer
22	Optimal placement and sizing of DGs in RDS	Saha and Mukherjee (2016)	IET Generation, Transmission & Distribution	IET

Table 7
Applications of SOS algorithm in Journal publications.

SN	Problem	Reference	Journal /conference	Application domain	Publisher
1	Training Feedforward Neural Networks	Wu et al. (2016)	Computational intelligence and neuroscience	Machine Learning	Hindawi
2	Capacitated Vehicle Routing Problem	Ruskartina et al. (2015)	International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering	Distribution networks	World Academy of Science, Engineering and Technology
3	Energy analysis	Sharma and Yerma (2016)	In Recent Trends in Electronics, Information & Communication Technology (RTEICT), IEEE International Conference	Scheduling	IEEE
4	Resource discovery in cloud computing environment	Ezugwu and Adewumi (2017b)	Future Generation Computer Systems	Scheduling	Elsevier
5	Task scheduling in cloud computing environment	Abdullahi et al. (2016)	Future Generation Computer Systems	Scheduling	Elsevier
6	Optimization of scientific workflow scheduling in cloud environment	Anwar and Deng (2017)	Sci.Int.(Lahore)	Scheduling	Publications International Lahore, Pakistan
7	WNN training for Robust nonlinear channel equalization	Nanda and Jonwal (2017)	Applied Soft Computing	Machine Learning	Elsevier
8	Global optimization of real world problems	Nama and Saha (2018)	Decision Science Letters	Unconstrained global optimization problem	Growing Science
9	Simultaneous Scheduling Of Jobs, AGVs And Tools Considering Tool Transfer Times In Multi Machine FMS	Reddy, Ramamurthy, and Rao (2016)	International Conference on Advanced Material Technologies (ICAMT)	Scheduling	IOP Publishing
10	Improved adaptive fuzzy back stepping control of a magnetic levitation system	Sadek et al. (2017)	Applied Soft Computing	Magnetic Levitation System	Elsevier

The performance of the SOS based training was demonstrated on equalization of two non-linear three taps channels and a linear twenty-three taps telephonic channel. The results obtained from the experimentation of Nanda and Jonwal (2017) were compared with other training metaheuristics such as the cat swarm optimization and clonal selection algorithm, PSO and least mean square algorithm. The authors conclude that the training done with the SOS algorithm provided superior results than the other algorithms (Nanda & Jonwal, 2017).

Ruskartina, Yu, Santosa, and Redi (2015) proposed the application of SOS algorithms to solve the capacitated vehicle routing problem (CVRP). To enable the standard SOS to handle the problem structure, a two solution representations approach was employed to transform the algorithm into an applicable solution approach for solving CVRPs. In addition, the algorithm's solution quality was also improved by combining SOS with a local search strategy. As part of the paper's main technical contributions, the authors developed six versions of the improved SOS algorithm specifically for solving CVRP (Vincent et al., 2017). Two sets of CVRP benchmark problems were evaluated in order to determine the performance of each of the proposed algorithms. The experimental results showed that the best of the six proposed methods outperformed the best-known solutions in terms of solution quality and convergence speed.

Ezugwu and Adewumi (2017b) applied the SOS algorithm to optimize the process of resource discovering in cloud computing environments. The paper tried to simultaneously address the issue of uncertainty that occurs between advertised resources in the cloud and users' resource requirement queries, and dynamic resource discovery in a cloud computing environment (Ezugwu & Adewumi, 2017b). Soft-set was combined with an SOS algorithm in the proposed method to handle the aforementioned problem. For this purpose, soft-set was applied to tackle uncertainty problems that arise in static systems, while SOS was employed to tackle dynamic relationships that occur in dynamic environments in search of optimal solutions among objects. The performance of the proposed method was demonstrated through an empirical simulation study and benchmarking against recent techniques in literature. The experimental results presented in Ezugwu and Adewumi (2017b) revealed that the employment of the proposed algorithm for resource discovery in a cloud environment, yielded satisfactory results over other existing methods.

Generally, the effective scheduling of jobs or tasks in a distributed computing environment such as in cloud computing is challenging, especially with numerous cloud jobs competing for limited resources with dynamic characteristics. Therefore, the job scheduling problem in this dynamic environment is said to be NP-Complete (Johnson, 1985). Abdullahi et al. (2016) employed the SOS algorithm for effective and optimal scheduling of tasks on cloud resources. The performance of the SOS algorithm in Abdullahi et al. (2016) was tested on some specified resources consisting of physical hosts, virtual machines and uniformly distributed jobs. The results showed that the SOS tasks scheduling algorithm outperformed other similar scheduling algorithms such as PSO.

There are other areas were SOS algorithm have been applied to solve problems, in which the algorithm was either used as an optimization algorithm or as a training algorithm. These areas are not discussed in detail in this paper. The other areas referred to include: antennas, biomedical sciences or engineering, communication networks, clustering and classification, combinatorial optimization, control, engineering design, distribution networks, electronics and electromagnetics, engines and motors, entertainment, faults, financial, fuzzy and neurofuzzy systems, graphics and visualization, image and video processing, neural networks, robotics, prediction and forecasting, power systems and plants, scheduling, security and military, sensor networks, and signal processing. Summaries with listings of some main areas of general application interests in this regard, are presented, respectively, in Table 7 for journal publications and in Table 8 for conference articles.

In a contribution to the body of knowledge in the form of a master's dissertation, an SOS algorithm was hybridized with a

Table 8
Applications of SOS algorithm in IEEE Conference Publications.

SN	Problem	Reference	Problem domain	Conference	Year
1	Synthesis of antenna arrays	Dib (2016)	Antenna	2016 IEEE International Symposium on Antennas and Propagation (APSURSI)	2016
2	Economic load dispatch problems	Guvenc et al. (2016)	Power systems and plants	2016 International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)	2016
3	Bid based economic load dispatch problem	Tiwari and Pandit (2016)	Power systems and plants	2016 IEEE International Conference on Engineering and Technology (ICETECH)	2016
4	Optimal scheduling of short-term hydrothermal generation		Scheduling	2016 4th International Istanbul Smart Grid Congress and Fair (ICSG)	2016
5	Real Power Loss minimization	Balachennaiah and Suryakalavathi (2015)	Power systems and plants	2015 Annual IEEE India Conference (INDICON)	2015
6	Energy-aware task scheduling in a cloud environment	Sharma and Verma (2017)	Scheduling	2017 4th International Conference on Signal Processing and Integrated Networks (SPIN)	2017
7	Optimization of three-dimensional turbo code	Banerjee and Chattopadhyay (2017)	Communication networks	2016 IEEE Annual India Conference (INDICON)	2016
8	Energy analysis	Sharma and Yerma (2016)	Power systems and plants	2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)	2016
9	Security-constrained economic dispatch problem	Rajathy et al. (2015)	Security	2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015]	2015
10	Selection and parameter optimization for underwater target classification	Sherin and Supriya (2015)	Clustering and classification	OCEANS 2016 MTS/IEEE Monterey	2016
11	Improved Identification of Hammerstein plant	Panda and Pani (2016)	Prediction and forecasting	2016 IEEE Region 10 Conference (TENCON)	2016
12	DG placement for loss minimization considering reverse power flow in the distribution systems	Lalitha et al. (2016)	Power systems and plants	2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)	2016
13	Electromagnetic optimization	Ayala et al. (2017)	Electronics and electromagnetics	2016 IEEE Conference on Electromagnetic Field Computation (CEFC)	2016
14	Task scheduling in cloud environment	Kumar et al. (2017)	Scheduling	2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)	2017
15	An improved regularized extreme learning machine based	Zhang et al. (2016)	Machine learning	2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)	2016
16	Economic dispatch with valve point effect	Naveen et al. (2016)	Financial	2016 International Conference on Energy Efficient Technologies for Sustainability (ICEETS)	2016

quantum neural network and used for predicting high performance concrete compressive strength (Aulady, 2014). In the hybridization process, the SOS was employed to optimize all the parameters within the quantum neural network. In doing so, the capability of the quantum neural network to efficiently solve most instances of high performance concrete compressive problems was improved. The researcher compared the performance of the hybrid algorithm with an artificial neural network and a quantum neural network, by using variants of high performance concrete compressive problem instances (Aulady, 2014). The results indicated that the proposed algorithm efficiently and effectively solved various complex problems and outperformed other methods based on the benchmarked problems tested.

As part of the original development of the SOS algorithm, Prayogo (2015) in his PhD thesis proposed the development of an innovative parameter-free symbiotic organisms search algorithm for solving construction-engineering problems. As mentioned above (see Section 2.5), when compared with other similar metaheuristics and population based algorithms, an SOS algorithm has the advantage of using less parameters, with only two parameters that include maximum evaluation numbers and population sizes. Prayogo (2015) tested the effectiveness of the proposed SOS algorithm on varying benchmarked problems that cover both mathematical functions and constructional engineering problems. The remarkable achievements of the SOS algorithm can therefore be attributed the research of Aulady (2014) and that of Prayogo (2015) contributing to the body of knowledge, which is notable in the algorithm's application to various research domains including real-world industrial problems. In addition, another scholarly work that featured SOS alongside other well-known state-of-the-art metaheuristic algorithms as part of his technical contribution to knowledge was the PhD thesis of Tejani (2017). Table 9 presents a comprehensive summary of the two academic and scholarly contributions of SOS algorithm.

An extended summary of most recent work on SOS is presented in Table 10, where publications related to the algorithm in terms of various areas of modification and hybridization as applied to different formulations of combinatorial optimization problems are studied.

4.3. SOS publication trends

This section presents a discussion of the trends in the publication of the application of SOS algorithms in the different domains highlighted above (see Section 4). The publication evaluation in this case is from the inception of the SOS algorithm

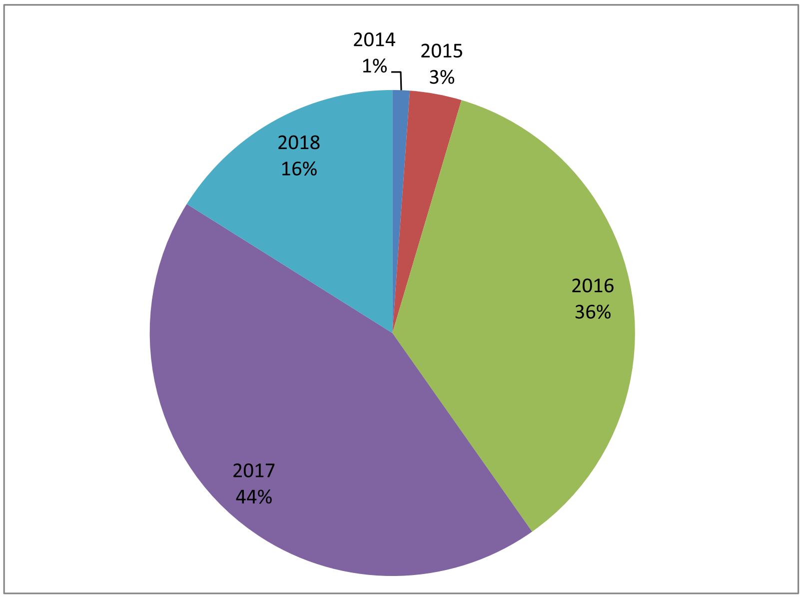

Fig. 5. Number of major SOS publication trends from Google Scholar and IEEE Explorer.

(2014) to the time of writing this paper (2018). The sources of all the publications identified were from the Google search engine, Google scholar, and IEEE Explore databases. In searching for publications about SOS algorithms, the authors identified the key search phrase "symbiotic organisms search algorithm" with restrictions to appearance, in article titles of around 184 relevant publications found, sorted by dates in Google's search database, 52 in Google scholar, and 25 in IEEE Explore database. According to the results of the descriptive statistics shown in Fig. 5, it is obvious that there is a dramatic rise in the acceptance and application of SOS algorithms, with the year 2017 (with a 36% publication rate) and the year 2017 (with a 44% publications rate) having the highest publication rates. The least publication count was recorded in the years 2014 and 2015, with 1% and 3% respectively. This is expected, because the algorithm was developed in 2014.

However, further analyses of these publications, reveals that the rate of using SOS is particularly astonishing, specifically for its application to different domain of science and engineering. For example, SOS has been widely used in computer science with specific interests in global optimization problems, power systems and plants. In addition, the number of publications reporting SOS applications has grown exponentially within the last three years, and seems to show no sign of slowing down at the present moment, which is a clear indication of SOS superiority and prospects over the other metaheuristic algorithms. The number of main publications using SOS from the literature is presented in Fig. 5, with sources from Google scholar and Scopus. The essential objective of this publication analyses is to furnish SOS practitioners, enthusiasts, and researchers with a clear up-to-date bird's eye view of what has been done so far in the area of SOS applications. Furthermore, the analysis provides researchers with firsthand information that illustrates the relevance of SOS algorithm to the development of science and engineering as a whole.

In Fig. 6, the major journal publishers in which SOS related literature appeared are illustrated using descriptive statistics based on a pie chart. This figure shows that SOS appeared in several reputable journal publishers like Elsevier with 26%, Springer 18%, and IEEE 20%. This is an indication that the algorithm has been widely accepted by the scientific and metaheuristic research communities as one of the most recent and popular optimization algorithms with a wide application interest and high performance profile. The others with 26% of SOS appearances include some Scopus and open access journals.

According to the descriptive statistics shown in Fig. 7, the major application areas in which the SOS algorithm was considered include the following: power systems, followed by combinatorial optimization, then construction engineering, and scheduling. These identified application areas are not exhaustive, as the search techniques applied only concentrated on two databases, which are Google and IEEE Explorer. However, every month, many new articles employing SOS algorithms within different computer science, engineering and industrial optimization areas of applications emerge, showing that the application of SOS in these domains is in high demand and rapidly expanding. The two subject areas where SOS receives the highest attention are in power systems and plants with 35% and combinatorial optimization with 24% contributions, respectively.

To further broaden the above application trend and subject areas of SOS algorithms, more specifically to cover highly reputable and impact factor publications on SOS, the number of published articles from 2014 up to 2018 was extracted from the Scopus database and subsequently analyzed. Fig. 8 shows the descriptive statistics of the publication trend.

In Fig. 9, the various subject areas in which SOS was used as the key metaheuristic or main application algorithm is presented. While the search results between the two major databases (Google scholar and Scopus) differ in the result analyses, it is obvious that they both show a similarity in the growth pattern for SOS in terms of publications and applications.

The yearly citation counts for SOS literature as found in the Web of Science Core Collection, which includes three flagship citation indexes, namely, the Science Citation Index Expanded (SCIE), the Social Sciences Citation Index (SSCI) and the Arts & Humanities Citation Index (AHCI), were extracted and evaluated. The citations count for SOS literature from 2014 to 2018 is illustrated in Fig. 10. The results show that the SOS literatures in 2018, received the highest citations count of 418, 415, and 413 in Web of Science, Scopus and Google scholar, respectively. This is followed by the citations in 2017, with citations counts of 341, 335, and 333 in the three citation indexes, while in 2016, the citation counts were 149, 144, and 144 in the three citation indexes. The least number of citations was recorded in 2014 with 1, 0, and 0 citations per cita-

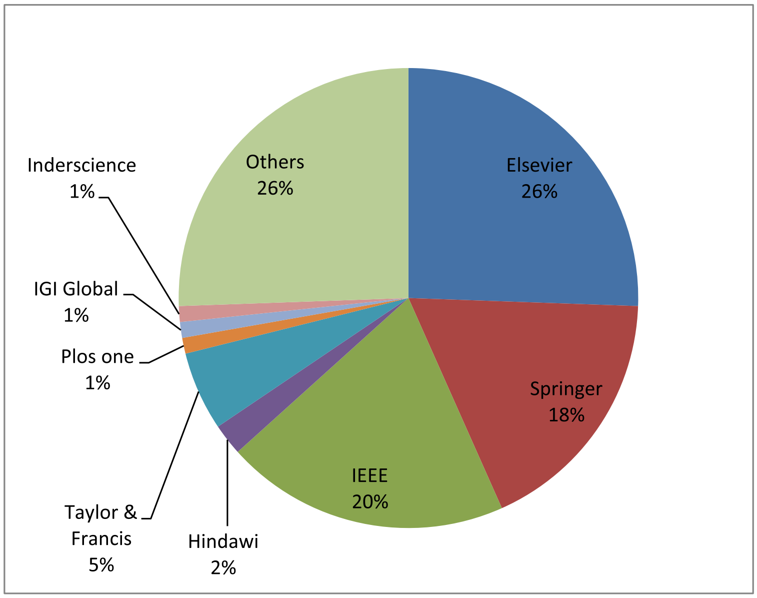

Fig. 6. SOS publications published by major journal publishers.

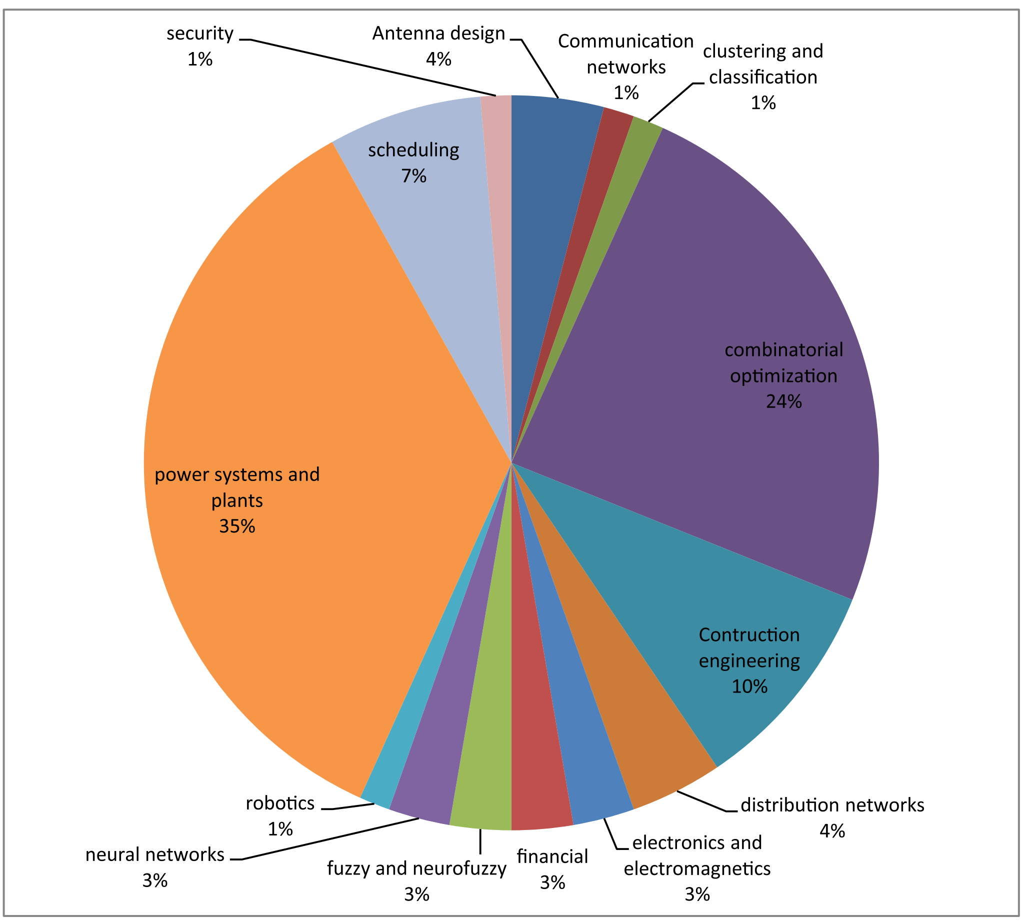

Fig. 7. Application areas of SOS algorithms from Google Scholar and IEEE Explorer.

tion index. However, the low citations count recorded for the articles published in 2014 can be attributed to the fact that it was the period in which the first article on SOS was published. It is noteworthy to mention here that the citations count presented in Fig. 10 does not describe the cumulative citation growth for the SOS articles, but rather it indicates the number of citations in major literature for a specific year. However, the cumulative number of citations in major literature or citation growth for the algorithm is illustrated in Fig. 11.

Therefore, to summarize the above SOS citations counts based on the sources of information received from the two major academic databases in which high impact literatures of SOS were pub-

Table 9
SOS scholarly contributions in thesis/dissertation.

SN	Method	Problem	Reference	Course	Institution
1	Hybrid Symbiotic Organisms Search-Quantum Neural Network	Prediction of High Performance Concrete Compressive Strength	Aulady (2014)	Construction Engineering/ Civil Engineering	National Taiwan University of Science and Technology
2	Innovative Parameter-Free SOS	Optimization of construction-engineering problems	Prayogo (2015)	Construction Engineering/ Civil Engineering	National Taiwan University of Science and Technology
3	Adaptive SOS	Investigation of advanced metaheuristic techniques for simultaneous size shape and topology optimization of truss structures	Tejani (2017)	Mechanical Engineering	Pandit Deendayal Petroleum University India

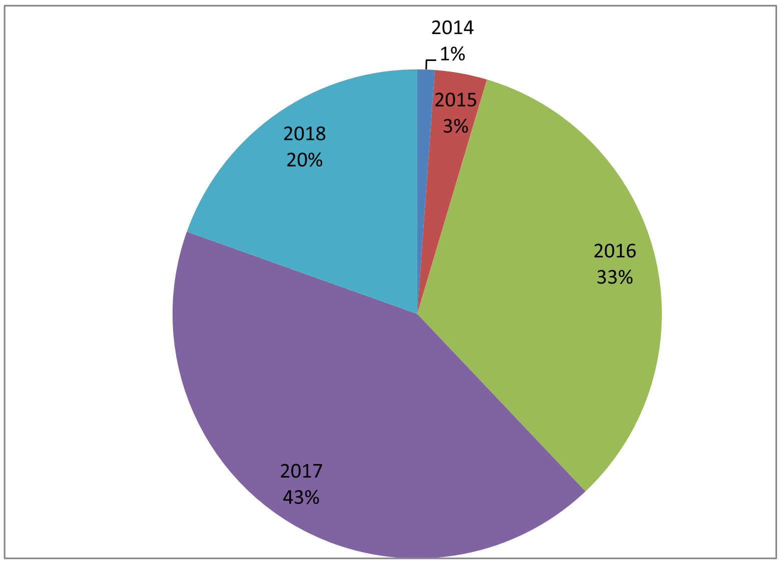

Fig. 8. Relevant SOS publication trends from Scopus.

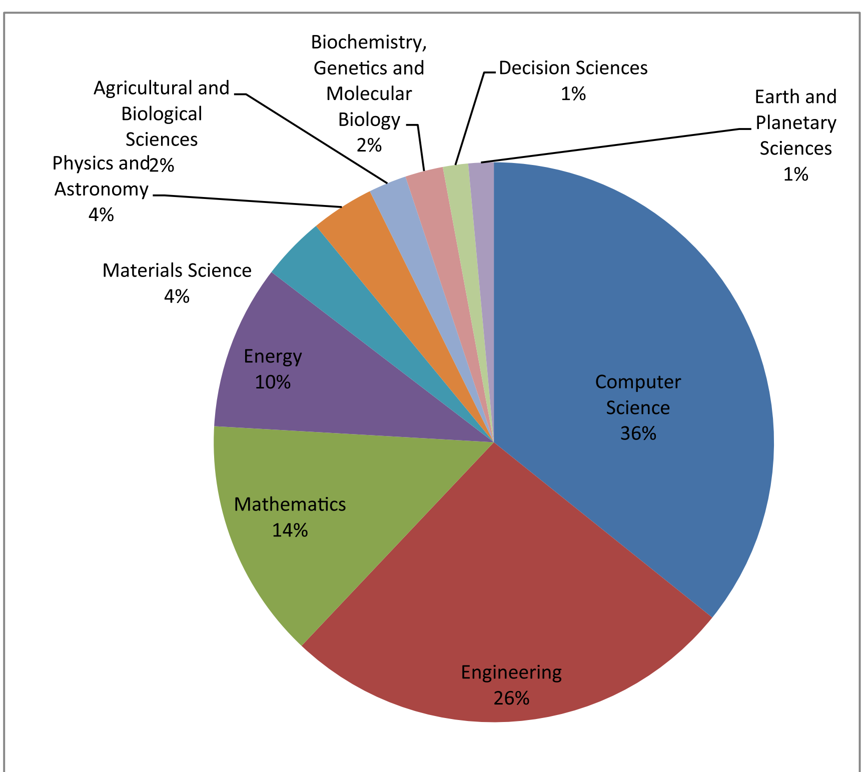

Fig. 9. Subject areas of SOS algorithm application from Scopus.

Table 10
Recent general applications of modified, improved and hybridized SOS algorithm.

SN	SOS variant	Problem	Problem domain	Reference
1	Improved	Truss optimization with natural frequency bounds	Construction engineering	Tejani et al. (2018)
2	Five methods: Discrete, Hybrid, cooperative hybrid, modified, modified.	Simultaneous optimization of feature subset and neighborhood size of KNN classification models	Machine learning	Liao and Kuo (2018)
3	Adaptive-SOS	Interval type-2 fuzzy-PID controller with derivative filter for automatic generation control of an interconnected power system	Power systems and plants	Nayak et al. (2018)
4	Multiple objective symbiotic	Time, cost, quality and work continuity tradeoff in repetitive projects	Construction engineering	Tran et al. (2018)
5	Modified	Optimal Number, Location, and Size of Distributed Generators in Distribution Systems	Power systems and plants	Nguyen and Vo (2018)
6	Hybrid	Optimal Operation of Directional Overcurrent Relays	Power systems and plants	Sulaiman et al. (2018)
7	Modified	PID controller for AGC of multi-area power system	Power systems and plants	Nayak et al. (2018)
8	Hybrid	Feasibility-based rules for constrained engineering design optimization	Construction engineering	Prayogo and Cheng (2017)
9	Modified	Load frequency control with TCSC	Power systems and plants	Guha et al. (2018b)
10	Hybrid	Efficient design of PID controller for automatic voltage regulator	Power systems and plants	Çelik and Öztürk (2018)
11	Modified/improved	Prediction of permanent deformation in asphalt pavements	Construction engineering	Cheng et al. (2018)
12	Quasi-Reflected	Short-Term Hydrothermal Scheduling Problem Considering Multi-fuel Cost Characteristics of Thermal Generator	Power systems and plants	Das et al. (2018)
13	Modified/improved	Load frequency control of multi-area power system	Power systems and plants	Guha et al. (2018a)
14	Modified	Investigation of static transmission expansion planning	Power systems and plants	Verma and Mukherjee (2018)
15	Modified	Optimal design of analog active filters	Power systems and plants	Dib and El-Asir (2018)
16	Improved	Unrelated parallel machines scheduling with sequence-dependent setup times.	Manufacturing optimization	Ezugwu et al. (2018)
17	Modified	Global optimization.	Continuous optimization	Miao et al. (2018)
18	Modified	Optimization model for construction project resource leveling	Construction engineering	Prayogo et al. (2018)
19	Modified	Portfolio Optimization in selected Tehran stock exchange companies	Portfolio optimization	Prayogo et al. (2018)

Table 11
The detailed of benchmark functions to evaluate standard SOS algorithm.

SN	Function	Search space	Dimension	minimum value	SOS(minimum value)
1	Beale	[−4.5, 4.5]	2	0	0
2	Easom	[−100, 100]	2	−1	−1
3	Matyas	[−10, 10]	2	0	0
4	Bohachevsky1	[−100, 100]	2	0	0
5	Booth	[−10, 10]	2	0	0.03382
6	Michalewicz2	[0, π]	2	−1.8013	−1.8013
7	Schaffer	[−100, 100]	2	0	0
8	Six Hump Camel Back	−5, 5]	2	−1.03163	−1.03163
9	Boachevsky2	[−100, 100]	2	0	0
10	Boachevsky3	[−100, 100]	2	0	0
11	Shubert	[−10, 10]	2	−186.73	−186.73
12	Colville	[−10, 10]	4	0	0
13	Michalewicz5	[0, π]	5	−4.6877	−4.6877
14	Zakharov	[−5, 10]	30	0	0
15	Michalewicz10	[0, π]	30	−9.6602	−9.65982
16	Step	[−5.12, 5.12]	30	0	0
17	Sphere	[−100, 100]	30	0	0
18	SumSquares	[100, 100]	30	0	0
19	Quartic	[−1.28, 1.28]	30	0	9.13E−05
20	Schwefel 2.22	[−10, 10]	30	0	0
21	Schwefel 1.2	[−100, 100]	30	0	0
22	Rosenbrock	[−30, 30]	30	0	1.04E−07
23	Dixon-Price	[10, 10]	30	0	0
24	Rastrigin	[−5.12, 5.12]	30	0	0
25	Griewank	[−600, 600]	30	0	0
26	Ackley	[32, 32]	30	0	0

Table 12
State-of-the-art metaheuristics that were compared with SOS.

	CEC2014 Benchmark function		Truss optimization problems	
SN	Method	Reference	Method	Reference
1	Invasive Weed Optimization	Mehrabian and Lucas (2006)	Niche genetic hybrid algorithm (NGHA).	Wang et al. (2004)
2	GSA	Rashedi et (2009)	Charged system search (CSS)	Kaveh and Talatahari (2010); Kaveh and Zolghadr (2011)
3	Hunting Search (HuS)	Oftadeh et al. (2010)	Enhanced CSS	Kaveh and Zolghadr (2011)
4	Bat Algorithm (BA)	Yang (2010a)	Harmony search (HS)	Miguel and Miguel (2012)
5	Water Wave Optimization (WWO)	Zheng (2015)	Firefly algorithm (FA)	Miguel and Miguel (2012)
6	Biogeography- Based Optimization (BBO)	Simon (2008)	Big bang-big crunch (CSS-BBBC)	Kaveh and Zolghadr (2012)
7			Optimality criterion (OC)	Wang et al. (2004)
8			Hybrid OC-GA	Zuo et al. (2014)
9			Colliding-bodies optimization (CBO) and 2D-CBO	Kaveh and Mahdavi (2014)
10			Particle swarm optimization (PSO)	Gomes (2011)
11			Democratic PSO (DPSO)	Kaveh and Zolghadr (2014)
12			Modified sub-population teaching-learning-based optimization (MS-TLBO)	Tejani et al. (2016)
13			Multi-Class TLBO (MC-TLBO)	Farshchin et al. (2016)
14			Tug of war optimization (TWO),	Kaveh and Zolghadr (2017)
15			Vibrating particles system (VPS)	Kaveh and Ilchi Ghazaan (2017)

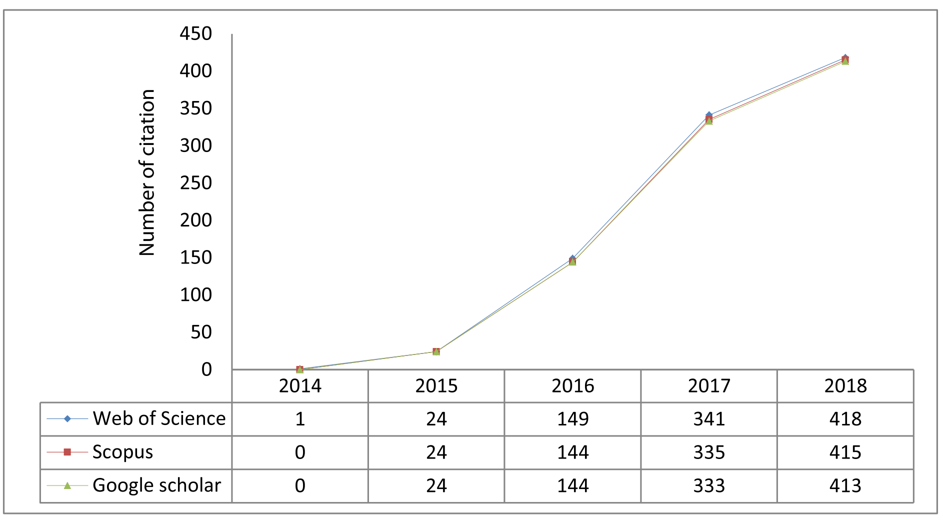

Fig. 10. Number of citations in major literature per year from Google scholar, Scopus, and Web of Science.

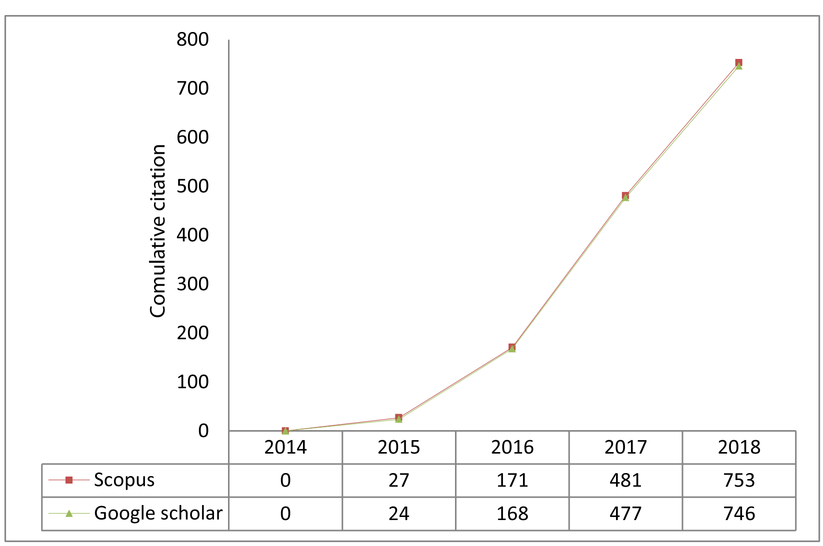

Fig. 11. Cumulative citation growth of SOS algorithm in major literature.

lished, more specifically for Web of Science and Scopus referencing, it is noted that the cumulative citations counts spanning from 2014 to 2018 are 753 and 746 respectively. However, the citation information provided here is not exhaustive, compared to the fact that the algorithm impact is exponentially growing as is evident in the total publication count for 2017 alone, which is more than 42% higher than the number of publications recorded in 2014, 40% higher than the number of publications recorded in 2015 and 10% higher than the number of publication recorded in 2016. To further support the above claim of SOS exponential growth, its impact on the research communities, and wide readability, the cumulative citation growth of SOS in major literature is presented in Fig. 11 below.

5. Performance Validation of Standard SOS

It is clear that the field of combinatorial optimization has witnessed an explosion of the so called new or novel metaheuristic algorithms, most of them based on a metaphor of some swarm intelligence, biological systems, and physical and chemical systems. Not all of these algorithms are, however, as efficient as proclaimed by their inventors. A few such as SOS have proved to be very efficient and thus have become popular tools for solving complex optimization and real-world problems. In this section, the various performance measurements criteria that have been employed by different researchers to validate the superiority of both the basic, improved and hybrid SOS algorithm against other related state-of-the-art metaheuristic algorithms, such as GA, DE, BA, PBA, PSO, and CS are compared and contrasted.

As required in the domain of metaheuristics for any new algorithm, a set of well-known mathematical benchmark functions and some selected real-world problems are often used to investigate the numerical optimization problem solving capabilities of such an algorithm. The required function evaluation number for reaching optimal solution and run-time complexity of the algorithms is usually evaluated in line with other existing techniques. Furthermore, the effects of population size on the performance of the algorithms are sometimes studied. However, the inventor of the basic SOS algorithm failed to conduct a comprehensive statistical analyses to validate the concluding remarks regarding the overall performance superiority of the proposed SOS over other competing state-of-the-art approaches. Reported in Table 11, are the details of the mathematical benchmark function test problems used to evaluate the standard SOS algorithm. In the comparative results of SOS with other methods, namely, GA, DE, PSO, BA, and PBA, SOS had an outstanding performance by being able to find twenty-two (22) global minimum functions out of the twenty-six (26) functions evaluated. This is followed by the PBA with 20 functions, BeeA and DE with 18, and the PSO, GA which is the least performed algorithm, with only 9 global minimum counts.

In the study conducted by Tejani, Savsani, Patel, and Mirjalili (2018), for which the exploitative behavior of the SOS was further improved to form a superior SOS with better solution accuracy, the feasibility and effectiveness of the improved SOS was validated on six (6) truss optimization problems and thirty (30) mathematical benchmark functions generated from the CEC2014 test suite. In that study, the computational results show that the improved SOS algorithm was more reliable and efficient than the compared state-of-the-art metaheuristics for all the test problem conducted. Table 12 presents the list of all the metaheuristic algorithms that were compared with the improved SOSas demonstrated in Tejani et al. (2018).

Other standard benchmark test problems in which the SOS algorithm was validated, include real world problems that cover various aspects of engineering design problems, such as cantilever beam, minimized I-beam vertical deflection, 15-bar planar truss structure, 52-bar planar truss structure, gear train, compression spring, pressure vessel, Lennard-jones, frequency modulation sound parameter identification, CEC05 and CECO5 functions (Al-Sharhan & Omran, 2018; Cheng & Prayogo, 2014).

6. Discussion

Part of the main contributions of this current review was to identify other novel and potential application areas that can be considered significant to the development of the SOS algorithm. More so, to also allow researchers to identify other aspects of the algorithms requiring further improvements. From the above analyses, it is clear that the application of the SOS algorithm and its variants are only limited to solving a few industrial based optimization problems, with little or no citation in other crucial areas such as the biomedical, entertainment, finances, image processing, oil and gas exploration, big data analytics, data mining, software engineering, quantum computing, and sensor networks.

In order to maintain the current exponential growth in the SOS algorithm utilization by different researchers for solving varying theoretical, industrial and real-world problems, a commensurate effort regarding the level of improvement in the algorithm's atomic components should be made. For example, the SOS algorithm uses the current best solution (X_{best}) found among the organism's population in the mutualism and commensalism phases to determine the quality of solution of the current problem. Also, since the X_{best} organism is used to model the highest degree of adaptation, it is equally seen as the main objective or the primary goal of each individual organism. Therefore, improving this single component can invariably enhance the search efficiency of the SOS in the different solution search spaces.

As proposed in the work of Al-Sharhan and Omran (2018), the network structure or topology in which the best organisms in the ecosystem are modeled, can also affect the algorithm's convergence rate, depending on the type of topology that was adopted for the implementation of the X_{best} solution search in the aforementioned two phases. For example, the authors emphasized that using a fully connected network structure or a star topology will efficiently accelerate the propagation of finding the so called best solution among the defined organism's population (Al-Sharhan & Omran, 2018). This approach might, however, also result in premature convergence. To avert this problem, Al-Sharhan and Omran (2018)proposed the use of a ring topology, which slows down the algorithm's convergence rate, because based on this network model, the X_{best} organism has to propagate through several neighborhoods before affecting all the individual organisms in the population. The advantage of this approach is that it decreases the chances of premature convergence by allowing the individual organisms to explore more areas in the available search spaces. This improvement is useful for handling multimodal functions (Mason & Howley, 2016; Peer, van den Bergh, & Engelbrecht, 2003).

The implication of the above finding is that the type of social network topology selected for the implementation of the two SOS interaction phases, mutualism and commensalism, has a direct impact on the behavior of the individual organism's population as a whole and much more on the propagation of the best organism too. Therefore, some of the most frequently used social network topologies such as the ring topology for the local best organisms, star topology for the global best organism, and the 3-D Von Neumann topology (Fig. 12) can be explored and used for further improvement of the basic SOS algorithm:

In most of the hybrid implementations involving SOS and other existing optimization techniques, SOS has always been used as a global optimizer, while the other algorithms are used as local search methods. Other improvement options that can possibly be considered in terms of augmenting SOS with some major meta-

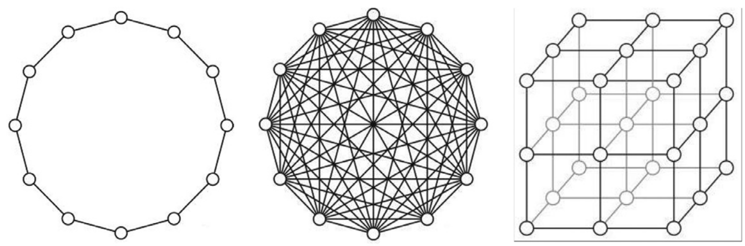

Fig. 12. Commonly used particle topologies (Bastos-Filho, Caraciolo, Miranda, & Carvalho, 2008).

heuristics algorithms for better performance and robustness can be explored. For example, it will be interesting to see some proposed implementation that considers the combination of SOS with the following metaheuristics: DE, ACO, PSO, and CS. Furthermore, other variants of local search strategies can also be investigated for possible hybridization with the SOS for improved accuracy and convergence speed.

The SOS algorithm uses the concept of greedy selection in its approach for modifying or updating newer organisms referred to as X_{inew} and X_{jnew}, which subsequently replace the original organisms X_i and X_j. However, this process works in such a way that it always selects the best step among the immediate choices, without looking ahead. The possible improvement in this part would be to improve the SOS algorithm's exploration and exploitation capability to find a better optimal solution with regards to the problem at hand. Exploration simply means that the SOS algorithm has the capability of investigating new and unknown areas in the solution search space, while exploitation means that the new method is able to make use of the knowledge found at the points previously visited to help find better points (Črepinšek et al., 2013). However, caution should be maintained regarding the level or depth to which these two search strategies should be applied. The reason being that too much exploration may slow down the algorithm's convergence rate, while on the other hand, too much exploitation may induce premature convergence. Therefore, maintaining a balance between the two methods can be an interesting improvement to achieve where possible.

7. Suggestions for further and future research

In the last two decades many nature inspired optimization algorithms like the SOS, a more recent algorithm, have been proposed with the intention of finding acceptable quality solutions to varying difficult optimization problems, where mathematical gradient-based techniques cannot be used. However, most of these algorithms are not really novel as claimed by the inventors, but it is simply a new metaphor for existing algorithms. As a result of this proliferation in the number of new algorithms, the number of surveys and literature reviews on these optimization algorithms has likewise increased. The main focus of all these surveys and the literature reviews of these optimization algorithms are on developing adequate and suitable benchmarks for the selection of best choice algorithms. However, most of the existing surveys have very limited empirical analysis and comparisons of these algorithms, which in most cases lead to poor judgments, decisions and choices of selecting best alternative algorithms. Therefore, a critical review is necessary for the identification and selection of the best algorithms based on the algorithm's problem orientations, control parameters, simplicity of implementation and robustness. In line with these identified points, the following suggestions are made for further and future research:

- An extensive empirical analysis of the SOS algorithm still needs to be conducted, with the objectives to -
  - conduct a sensitivity analysis of the SOS' only control parameters (population size and maximum number of function evaluation) and other hybrid parameters combined, with the outcome of determining optimal control parameter values for different optimization problem characteristics;
  - determine which of these parameter configurations leads SOS to perform better with respect to accuracy, computational efficiency, robustness, and reliability for different problem classes; and
  - determine the effect of the different atomic components on performance, and on the exploration-exploitation trade-off of the SOS algorithm.
- Conducting a study to determine if the SOS algorithm, which is a stochastic search algorithm, actually performs better than random search, and to determine how much is it better than random search.
- Undertaking a study by using Markov chains and complexity analysis theory to thoroughly analyze the convergence behaviors of the basic SOS and all its variants algorithms.

8. Conclusion

This paper presents a timely review of the SOS algorithm and its applications in various research domains. The study covers all research contributions available to the authors, concerning SOS and its variants, which were obtained by identifying and analyzing around 87 major SOS journal articles indexed in Web of Science alone. More so, about 187 journal and conference papers indexed in Google Scholar, Scopus, and IEEE Explorer databases as at the time this paper was drafted were equally analyzed. However, despite the fact that SOS is new in the domain of metaheuristics, results from the review process conducted in the current study, clearly show that the algorithm has gained a wider scope of application either in its standard form or hybrid implementation with other algorithms. The dramatic growth of this new algorithm is demonstrated in the exponential increase in the number of published research articles that keeps emerging monthly on the subject of SOS. Therefore, in order to keep pace with the current trend in application and growth of the algorithm, some necessary improvements were suggested that would further increase the overall performance of the algorithms. Similarly, other novel application domains were like-wise highlighted for interested researchers to explore.

The popularity and attractiveness of SOS to researchers are not surprising, because of the algorithm's unique characteristics, which make it different from other existing metaheuristic algorithms. For example, SOS is virtually parameter free and therefore does not require any parameter fine-tuning like in the case of other competing algorithms such as GA, ACO, PSO and CS. This enhances the performance stability of SOS. Another interesting aspect of the algorithm is its simplicity of implementation and adaptability, with not more than two parameters, the algorithm structural flow and encoding can easily be retraced, restructured and adapted to solve different complex problems in different environments. Furthermore, the three phases of the algorithm, namely mutualism, commensalism, and parasitism are also very simple to code and operate using only simple mathematical operations to implement.

Acknowledgement

The authors would like to thank the editor-in-chief, the associate editor, and reviewers for their constructive feedback during the review process. We deeply appreciate the reviewers' comments for making this paper more complete.

References

Abdullahi, M., & Ngadi, M. A. (2016). Hybrid symbiotic organisms search optimization algorithm for scheduling of tasks on cloud computing environment. *PloS one, 11*(6), e0158229 29 pages.

Abdullahi, M., Ngadi, M. A., & Abdulhamid, S. M. (2016). Symbiotic Organism Search optimization based task scheduling in cloud computing environment. *Future Generation Computer Systems, 56*, 640–650.

Alomoush, M. (2017). Concurrent optimal design of TCSC and PSS using symbiotic organisms search algorithm. *Turkish Journal of Electrical Engineering & Computer Sciences, 25*(5), 3904–3919.

Al-Sharhan, S., & Omran, M. G. (2018). An enhanced symbiosis organisms search algorithm: An empirical study. *Neural Computing and Applications, 29*(11), 1025–1043.

Anwar, N., & Deng, H. (2017). Optimization of scientific workflow scheduling in cloud environment through a hybrid symbiotic organism search algorithm. *Sci.Int.(Lahore), 29*(3), 499–502 2017.

Aulady, M. (2014). *A hybrid symbiotic organisms search-quantum neural network for predicting high performance concrete compressive strength* Master Thesis. Department of Construction Engineering National Taiwan University of Science and Technology.

Ayala, H. V. H., Klein, C. E., Mariani, V. C., & Coelho, L. D. S. (2017). Multiobjective symbiotic search algorithm approaches for electromagnetic optimization. *IEEE Transactions on Magnetics, 53*(6), 1–4.

Balachennaiah, P., & Suryakalavathi, M. (2015). Real Power Loss minimization using symbiotic organisms search algorithm. In *India Conference (INDICON), 2015 Annual IEEE* (pp. 1–6). IEEE.

Banerjee, S., & Chattopadhyay, S. (2017). Power optimization of three dimensional turbo code using a novel modified symbiotic organism search (MSOS) algorithm. *Wireless Personal Communications, 92*(3), 941–968.

Bastos-Filho, C. J., Caraciolo, M. P., Miranda, P. B., & Carvalho, D. F. (2008). Multi-ring particle swarm optimization. In *Neural Networks, 2008. SBRN'08. 10th Brazilian Symposium on* (pp. 111–116). IEEE.

Beheshti, Z., & Shamsuddin, S. M. H. (2013). A review of population-based meta-heuristic algorithms. *International Journal of Advanced Soft Computing Applications, 5*(1), 1–35.

Birattari, M., Paquete, L., Stüztle, T., & Varrentrapp, K. (2001). Classification of Metaheuristics and Design of Experiments for the Analysis of Components. Technical report, Darmstadt University of Technology, Germany.

Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM computing surveys (CSUR), 35*(3), 268–308.

Boussaïd, I., Lepagnot, J., & Siarry, P. (2013). A survey on optimization metaheuristics. *Information Sciences, 237*, 82–117.

Bozorg-Haddad, O., Azarnivand, A., Hosseini-Moghari, S. M., & Loáiciga, H. A. (2017). Optimal operation of reservoir systems with the symbiotic organisms search (SOS) algorithm. *Journal of Hydroinformatics, 19*(4), 507–521 jh2017085.

Çelik, E., & Öztürk, N. (2018). A hybrid symbiotic organisms search and simulated annealing technique applied to efficient design of PID controller for automatic voltage regulator. *Soft Computing*, 1–14. doi:10.1007/s00500-018-3432-2.

Cheng, M. Y., & Lien, L. C. (2012). Hybrid Artificial Intelligence–Based PBA for Benchmark Functions and Facility Layout Design Optimization. *Journal of Computing in Civil Engineering, 26*(5), 612–624.

Cheng, M. Y., & Prayogo, D. (2014). Symbiotic organisms search: A new metaheuristic optimization algorithm. *Computers & Structures, 139*, 98–112.

Cheng, M. Y., Prayogo, D., & Tran, D. H. (2015). Optimizing multiple-resources leveling in multiple projects using discrete symbiotic organisms search. *Journal of Computing in Civil Engineering, 30*(3), 04015036 9 pages.

Cheng, M. Y., Prayogo, D., & Wu, Y. W. (2018). Prediction of permanent deformation in asphalt pavements using a novel symbiotic organisms search–least squares support vector regression. *Neural Computing and Applications*, 1–13. doi:10.1007/s00521-018-3426-0.

Cheng, M.-Y., Chiu, C.-K., Chiu, Y.-F., Wu, Y.-W., Syu, Z.-L., Prayogo, D., & Lin, C.-H. (2014). SOS optimization model for bridge life cycle risk evaluation and maintenance strategies. *Journal of the Chinese Institute of Civil and Hydraulic Engineering, 26*(4), 293–308.

Chu, S. C., Tsai, P. W., & Pan, J. S. (2006). Cat swarm optimization. In *Pacific Rim International Conference on Artificial Intelligence* (pp. 854–858). Berlin, Heidelberg: Springer.

Clerc, M., & Kennedy, J. (2002). The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE transactions on Evolutionary Computation, 6*(1), 58–73.

Črepinšek, M., Liu, S. H., & Mernik, M. (2013). Exploration and exploitation in evolutionary algorithms: A survey. *ACM Computing Surveys, 45*(3), 1–33 Article 35.

Črepinšek, M., Mernik, M., & Liu, S. H. (2011). Analysis of exploration and exploitation in evolutionary algorithms by ancestry trees. *International Journal of Innovative Computing and Applications, 3*(1), 11–19.

Das, S., & Bhattacharya, A. (2016). Symbiotic organisms search algorithm for short-term hydrothermal scheduling. *Ain Shams Engineering Journal*. doi:10.1016/j.asej.2016.04.002.

Das, S., Bhattacharya, A., & Chakraborty, A. K. (2018). Solution of Short-Term Hydrothermal Scheduling Problem Using Quasi-Reflected Symbiotic Organisms Search Algorithm Considering Multi-fuel Cost Characteristics of Thermal Generator. *Arabian Journal for Science and Engineering, 43*(6), 2931–2960.

Das, S., Mukhopadhyay, A., Roy, A., Abraham, A., & Panigrahi, B. K. (2011). Exploratory power of the harmony search algorithm: Analysis and improvements for global numerical optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 41*(1), 89–106.

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*: Vol. 16. New York, NY: John Wiley & Sons Inc..

Dib, N. (2017). Design of planar concentric circular antenna arrays with reduced side lobe level using symbiotic organisms search. *Neural Computing and Applications*, 1–10. doi:10.1007/s00521-017-2971-2.

Dib, N. I. (2016). Design of linear antenna arrays with low side lobes level using symbiotic organisms search. *Progress In Electromagnetics Research B, 68*, 55–71.

Dib, N., & El-Asir, B. (2018). Optimal design of analog active filters using symbiotic organisms search. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, 31*(5), 1–15 e2323.

Do, D. T., & Lee, J. (2017). A modified symbiotic organisms search (mSOS) algorithm for optimization of pin-jointed structures. *Applied Soft Computing, 61*, 683–699.

Dorigo, M., & Di Caro, G. (1999). Ant colony optimization: A new meta-heuristic. In *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on: 2* (pp. 1470–1477). IEEE.

Dosoglu, M. K., Guvenc, U., Duman, S., Sonmez, Y., & Kahraman, H. T. (2018). Symbiotic organisms search optimization algorithm for economic/emission dispatch problem in power systems. *Neural Computing and Applications, 29*(3), 721–737.

Duman, S. (2017). Symbiotic organisms search algorithm for optimal power flow problem based on valve-point effect and prohibited zones. *Neural Comput & Applic, 28*, 3571. https://doi.org/10.1007/s00521-016-2265-0.

Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. In *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on* (pp. 39–43). IEEE.

Ezugwu, A. E. S., & Adewumi, A. O. (2017a). Discrete Symbiotic Organisms Search Algorithm for Travelling Salesman Problem. *Expert Systems With Applications, 87*(2017), 70–78.

Ezugwu, A. E. S., Adewumi, A. O., & Frîncu, M. E. (2017). Simulated annealing based symbiotic organisms search optimization algorithm for traveling salesman problem. *Expert Systems with Applications, 77*, 189–210.

Ezugwu, A. E., & Adewumi, A. O. (2017b). Soft sets based symbiotic organisms search algorithm for resource discovery in cloud computing environment. *Future Generation Computer Systems, 76*(2017), 33–50.

Ezugwu, A. E., Adeleke, O. J., & Viriri, S. (2018). Symbiotic organisms search algorithm for the unrelated parallel machines scheduling with sequence-dependent setup times. *PloS one, 13*(7), 1–23 e0200030. doi:10.1371/journal.pone.0200030.

Farshchin, M., Camp, C. V., & Maniat, M. (2016). Multi-class teaching–learning-based optimization for truss design with frequency constraints. *Engineering Structures, 106*, 355–369.

Fister, I., Fister, I., Jr, Yang, X. S., & Brest, J. (2013). A comprehensive review of firefly algorithms. *Swarm and Evolutionary Computation, 13*, 34–46.

Gandomi, A. H., & Alavi, A. H. (2012). Krill herd: A new bio-inspired optimization algorithm. *Communications in Nonlinear Science and Numerical Simulation, 17*(12), 4831–4845.

Glover, F. (1989). Tabu search—part I. *ORSA Journal on computing, 1*(3), 190–206.

Gomes, H. M. (2011). Truss optimization with dynamic constraints using a particle swarm algorithm. *Expert Systems with Applications, 38*(1), 957–968.

Guha, D., Roy, P. K., & Banerjee, S. (2018). Symbiotic organism search algorithm applied to load frequency control of multi-area power system. *Energy Systems, 9*(2), 439–468.

Guha, D., Roy, P. K., & Banerjee, S. (2018a). Symbiotic organism search algorithm applied to load frequency control of multi-area power system. *Energy Systems, 9*(2), 439–468.

Guha, D., Roy, P. K., & Banerjee, S. (2018b). Symbiotic organism search based load frequency control with TCSC. In *2018 4th International Conference on Recent Advances in Information Technology (RAIT)* (pp. 1–6). IEEE.

Guo, Y., Li, N., & Ye, T. (2009). Multiple resources leveling in multiple projects scheduling problem using particle swarm optimization. In *Natural Computation, 2009. ICNC'09. Fifth International Conference on: 3* (pp. 260–264). IEEE.

Guvenc, U., Duman, S., Dosoglu, M. K., Kahraman, H. T., Sonmez, Y., & Yılmaz, C. (2016). Application of Symbiotic Organisms Search Algorithm to solve various economic load dispatch problems. In *INnovations in Intelligent SysTems and Applications (INISTA), 2016 International Symposium on* (pp. 1–7). IEEE.

Hasanien, H. M., & El-Fergany, A. A. (2016). Symbiotic organisms search algorithm for automatic generation control of interconnected power systems including wind farms. *IET Generation, Transmission & Distribution, 11*(7), 1692–1700.

Haupt, R. (1995). Comparison between genetic and gradient-based optimization algorithms for solving electromagnetics problems. *IEEE Transactions on Magnetics, 31*(3), 1932–1935.

Ishibuchi, H., Yoshida, T., & Murata, T. (2003). Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling. *IEEE transactions on evolutionary computation, 7*(2), 204–223.
Johnson, D. S. (1985). The NP-completeness column: An ongoing guide. *Journal of Algorithms, 6*(3), 434–451.
Kalkhambkar, V., Rawat, B., Kumar, R., & Bhakar, R. (2017). Optimal Allocation of Renewable Energy Sources for Energy Loss Minimization. *Journal of Electrical Systems, 113-1*(2017), 115–130.
Kanimozhi, G., Rajathy, R., & Kumar, H. (2016). Minimizing Energy of Point Charges on a Sphere using Symbiotic Organisms Search Algorithm. *International Journal on Electrical Engineering and Informatics, 8*(1), 1–29.
Kaveh, A., & Ghazaan, M. I. (2017). Vibrating particles system algorithm for truss optimization with multiple natural frequency constraints. *Acta Mechanica, 228*(1), 307–322.
Kaveh, A., & Mahdavi, V. R. (2014). Colliding-bodies optimization for truss optimization with multiple frequency constraints. *Journal of Computing in Civil Engineering, 29*(5) 04014078. doi:10.1061/(ASCE)CP.1943-5487.0000402.
Kaveh, A., & Talatahari, S. (2010). A novel heuristic optimization method: Charged system search. *Acta Mechanica, 213*(3-4), 267–289.
Kaveh, A., & Zolghadr, A. (2011). Shape and size optimization of truss structures with frequency constraints using enhanced charged system search algorithm, Asian J. *Civil Eng, 12*(4), 487–509.
Kaveh, A., & Zolghadr, A. (2012). Truss optimization with natural frequency constraints using a hybridized CSS-BBBC algorithm with trap recognition capability. *Computers & Structures, 102*, 14–27.
Kaveh, A., & Zolghadr, A. (2014). Democratic PSO for truss layout and size optimization with frequency constraints. *Computers & Structures, 130*, 10–21.
Kaveh, A., & Zolghadr, A. (2017). Truss shape and size optimization with frequency constraints using tug of war optimization. *Asian J Civil Eng, 7*(2), 311–333.
Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *science, 220*(4598), 671–680.
Krishnanand, K. N., & Ghose, D. (2009). Glowworm swarm optimisation: A new method for optimising multi-modal functions. *International Journal of Computational Intelligence Studies, 1*(1), 93–119.
Kumar, K. P., Kousalya, K., & Vishnuppriya, S. (2017). DSOS with local search for task scheduling in cloud environment. In *Advanced Computing and Communication Systems (ICACCS), 2017 4th International Conference on* (pp. 1–4). IEEE.
Lalitha, M. P., Babu, P. S., & Adivesh, B. (2016). Optimal Distributed Generation and Capacitor Placement for Loss Minimization and Voltage Profile Improvement using Symbiotic Organisms Search Algorithm. *International Journal of Electrical Engineering, 9*(3), 249–261.
Liao, T. W., & Kuo, R. J. (2018). Five discrete symbiotic organisms search algorithms for simultaneous optimization of feature subset and neighborhood size of KNN classification models. *Applied Soft Computing, 64*, 581–595.
Mason, K., & Howley, E. (2016). Exploring avoidance strategies and neighbourhood topologies in particle swarm optimisation. *International Journal of Swarm Intelligence, 2*(2-4), 188–207.
Mehrabian, A. R., & Lucas, C. (2006). A novel numerical optimization algorithm inspired from weed colonization. *Ecological informatics, 1*(4), 355–366.
Miao, F., Zhou, Y., & Luo, Q. (2018). Complex-valued encoding symbiotic organisms search algorithm for global optimization. *Knowl Inf Syst*, 1–40. doi:10.1007/s10115-018-1158-1.
Miguel, L. F. F., & Miguel, L. F. F. (2012). Shape and size optimization of truss structures considering dynamic constraints through modern metaheuristic algorithms. *Expert Systems with Applications, 39*(10), 9458–9467.
Mladenović, N., & Hansen, P. (1997). Variable neighborhood search. *Computers & operations research, 24*(11), 1097–1100.
Nama, S., & Saha, A. (2018). An ensemble symbiosis organisms search algorithm and its application to real world problems. *Decision Science Letters, 7*(2), 103–118.
Nama, S., Saha, A. K., & Ghosh, S. (2017). A hybrid symbiosis organisms search algorithm and its application to real world problems. *Memetic Computing, 9*(3), 261–280.
Nama, S., Saha, A., & Ghosh, S. (2016). Improved symbiotic organisms search algorithm for solving unconstrained function optimization. *Decision Science Letters, 5*(3), 361–380.
Nanda, S. J., & Jonwal, N. (2017). Robust nonlinear channel equalization using WNN trained by symbiotic organism search algorithm. *Applied Soft Computing, 57*, 197–209.
Naveen, P., Chandel, A. K., Vedik, B., & Topwal, K. (2016). Economic dispatch with valve point effect using symbiotic oragnisms search algorithm. In *In Energy Efficient Technologies for Sustainability (ICEETS), 2016 International Conference on (pp. 430-435)*. IEEE.
Nayak, J. R., Shaw, B., & Sahu, B. K. (2018). Application of adaptive-SOS (ASOS) algorithm based interval type-2 fuzzy-PID controller with derivative filter for automatic generation control of an interconnected power system. *Engineering Science and Technology, an International Journal, 21*(3), 465–485.
Nguyen, T. P., & Vo, D. N. (2018). Optimal Number, Location, and Size of Distributed Generators in Distribution Systems by Symbiotic Organism Search Based Method. *Advances in Electrical and Electronic Engineering, 15*(5), 724–735.
Nguyen, T. P., Dieu, V. N., & Vasant, P. (2017). Symbiotic Organism Search Algorithm for Optimal Size and Siting of Distributed Generators in Distribution Systems. *International Journal of Energy Optimization and Engineering (IJEOE), 6*(3), 1–28.
Nowak, W., & Cirpka, O. A. (2004). A modified Levenberg–Marquardt algorithm for quasi-linear geostatistical inversing. *Advances in water resources, 27*(7), 737–750.
Oftadeh, R., Mahjoob, M. J., & Shariatpanahi, M. (2010). A novel meta-heuristic optimization algorithm inspired by group hunting of animals: Hunting search. *Computers & Mathematics with Applications, 60*(7), 2087–2098.
Oh, I. S., Lee, J. S., & Moon, B. R. (2004). Hybrid genetic algorithms for feature selection. *IEEE Transactions on pattern analysis and machine intelligence, 26*(11), 1424–1437.
Osman, I. H., & Laporte, G. (1996). Metaheuristics: A bibliography. *Ann Oper Res, 1996*, 511–623.
Pan, W. T. (2012). A new fruit fly optimization algorithm: Taking the financial distress model as an example. *Knowledge-Based Systems, 26*, 69–74.
Panda, A., & Pani, S. (2016). A Symbiotic Organisms Search algorithm with adaptive penalty function to solve multi-objective constrained optimization problems. *Applied Soft Computing, 46*, 344–360.
Panda, A., & Pani, S. (2018). An orthogonal parallel symbiotic organism search algorithm embodied with augmented Lagrange multiplier for solving constrained optimization problems. *Soft Computing, 22*(8), 2429–2447.
Passino, K. M. (2002). Biomimicry of bacterial foraging for distributed optimization and control. *IEEE control systems, 22*(3), 52–67.
Peer, E. S., van den Bergh, F., & Engelbrecht, A. P. (2003). Using neighbourhoods with the guaranteed convergence PSO. In *Swarm Intelligence Symposium, 2003. SIS'03. Proceedings of the 2003 IEEE* (pp. 235–242). IEEE.
Pham, D. T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S., & Zaidi, M. (2011). The bees algorithm-A novel tool for complex optimisation. *Intelligent Production Machines and Systems-2nd I* PROMS Virtual International Conference (3-14 July 2006)* sn.
Prasad, D., & Mukherjee, V. (2016). A novel symbiotic organisms search algorithm for optimal power flow of power system with FACTS devices. *Engineering Science and Technology, an International Journal, 19*(1), 79–89.
Prasad, D., & Mukherjee, V. (2018). Solution of optimal reactive power dispatch by symbiotic organism search algorithm incorporating FACTS devices. *IETE Journal of Research, 64*(1), 149–160.
Prayogo, D. (2015). *An Innovative Parameter-Free Symbiotic Organisms Search (SOS) for Solving Construction-Engineering Problems* PhD Thesis. Department of Construction Engineering National Taiwan University of Science and Technology.
Prayogo, D., & Cheng, M. Y. (2017). Symbiotic organisms search with the feasibility-based rules for constrained engineering design optimization. In *2017 International Conference on Advanced Mechatronics, Intelligent Manufacture, and Industrial Automation (ICAMIMIA)* (pp. 13–18). IEEE.
Prayogo, D., Cheng, M. Y., & Prayogo, H. (2017). *A novel implementation of nature-inspired optimization for civil engineering: A comparative study of symbiotic organisms search*. Doctoral dissertation, Petra Christian University.
Prayogo, D., Cheng, M. Y., Wong, F. T., Tjandra, D., & Tran, D. H. (2018). Optimization model for construction project resource leveling using a novel modified symbiotic organisms search. *Asian J Civ Eng, 19*, 625. doi:10.1007/s42107-018-0048-x.
Rajathy, R., Taraswinee, B., & Suganya, S. (2015). A novel method of using symbiotic organism search algorithm in solving security-constrained economic dispatch. In *Circuit, Power and Computing Technologies (ICCPCT), 2015 International Conference on* (pp. 1–8). IEEE.
Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2009). GSA: a gravitational search algorithm. *Information sciences, 179*(13), 2232–2248.
Reddy, N. S., Ramamurthy, D. V., & Rao, K. P. (2016). Simultaneous Scheduling Of Jobs, AGVs And Tools Considering Tool Transfer Times In Multi Machine FMS By SOS Algorithm. In *International Conference on Advanced Material Technologies (ICAMT) (Vol. 2016, No. 27th). IOP Conf. Series: Materials Science and Engineering 225 (2017) 012294* (pp. 1–10). doi:10.1088/1757-899X/225/1/012294.
Ruskartina, E., Yu, V. F., Santosa, B., & Redi, A. A. N. P. (2015). Symbiotic Organism Search (SOS) for Solving the Capacitated Vehicle Routing Problem. *International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering, 101*, 857–861.
Sadek, U., Sarjaš, A., Chowdhury, A., & Svečko, R. (2017). Improved adaptive fuzzy backstepping control of a magnetic levitation system based on Symbiotic Organism Search. *Applied Soft Computing, 56*, 19–33.
Saha, D., Datta, A., & Das, P. (2016). Optimal coordination of directional overcurrent relays in power systems using Symbiotic Organism Search Optimisation technique. *IET Generation, Transmission & Distribution, 10*(11), 2681–2688.
Saha, S., & Mukherjee, V. (2016). Optimal placement and sizing of DGs in RDS using chaos embedded SOS algorithm. *IET Generation, Transmission & Distribution, 10*(14), 3671–3680.
Saha, S., & Mukherjee, V. (2018). A novel chaos-integrated symbiotic organisms search algorithm for global optimization. *Soft Comput, 22*, 3797. doi:10.1007/s00500-017-2597-4.
Secui, D. C. (2016). A modified Symbiotic Organisms Search algorithm for large scale economic dispatch problem with valve-point effects. *Energy, 113*, 366–384.
Sedighizadeh, M., Esmaili, M., & Eisapour-Moarref, A. (2017). Hybrid Symbiotic Organisms Search for Optimal Fuzzified Joint Reconfiguration and Capacitor Placement in Electric Distribution Systems. *INAE Letters, 2*(3), 107–121.
Sharma, M., & Verma, A. (2017). February. Energy-aware discrete symbiotic organism search optimization algorithm for task scheduling in a cloud environment. In *Signal Processing and Integrated Networks (SPIN), 2017 4th International Conference on* (pp. 513–518). IEEE.
Sharma, M., & Yerma, A. (2016, May). Energy analysis of symbiotic organisms search optimization based task scheduling algorithm. In *Recent Trends in Electronics, Information & Communication Technology (RTEICT), IEEE International Conference on* (pp. 1421–1424). IEEE.
Sherin, B. M., & Supriya, M. H. (2015). Selection and parameter optimization of SVM kernel function for underwater target classification. In *In Underwater Technology (UT), 2015 IEEE (pp. 1-5)*. IEEE.

Simon, D. (2008). Biogeography-based optimization. *IEEE transactions on evolutionary computation, 12*(6), 702–713.
Sonmez, Y., Kahraman, H. T., Dosoglu, M. K., Guvenc, U., & Duman, S. (2017). Symbiotic organisms search algorithm for dynamic economic dispatch with valve–point effects. *Journal of Experimental & Theoretical Artificial Intelligence, 29*(3), 495–515.
Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization, 11*(4), 341–359.
Sulaiman, M., Ahmad, A., Khan, A., & Muhammad, S. (2018). Hybridized Symbiotic Organism Search Algorithm for the Optimal Operation of Directional Overcurrent Relays. *Complexity* 11 pages, Article ID 4605769. doi:10.1155/2018/4605769.
Tahk, M. J., Park, M. S., Woo, H. W., & Kim, H. J. (2009). Hessian approximation algorithms for hybrid optimization methods. *Engineering Optimization, 41*(7), 609–633.
Talatahari, S. (2015). Symbiotic Organisms Search for Optimum Design of Frame and Grillage Systems. *Asian Journal of Civil Engineering (BHRC), 17*(3), 299–313.
Tashkova, K., Šilc, J., Atanasova, N., & Džeroski, S. (2012). Parameter estimation in a nonlinear dynamic model of an aquatic ecosystem with meta-heuristic optimization. *Ecological Modelling, 226*, 36–61.
Tejani, G. G. (2017). *Investigation of advanced metaheuristic techniques for simultaneous size shape and topology optimization of truss structures*. PhD Thesis, Pandit Deendayal Petroleum University. India.
Tejani, G. G., Savsani, V. J., & Patel, V. K. (2016). Adaptive symbiotic organisms search (SOS) algorithm for structural design optimization. *Journal of Computational Design and Engineering, 3*(3), 226–249.
Tejani, G. G., Savsani, V. J., Patel, V. K., & Mirjalili, S. (2018). Truss optimization with natural frequency bounds using improved symbiotic organisms search. *Knowledge-Based Systems, 143*, 162–178.
Teodorovic, D., & Dell'Orco, M. (2005). Bee colony optimization–a cooperative learning approach to complex transportation problems. *Advanced OR and AI methods in transportation*, 51–60. https://pdfs.semanticscholar.org/584e/fff67876e262ec174bd3f7dc46bc883f9500.pdf
Tiwari, A., & Pandit, M. (2016). Bid based economic load dispatch using symbiotic organisms search algorithm. In *Engineering and Technology (ICETECH), 2016 IEEE International Conference on* (pp. 1073–1078). IEEE.
Tran, D. H., Cheng, M. Y., & Prayogo, D. (2016). A novel Multiple Objective Symbiotic Organisms Search (MOSOS) for time–cost–labor utilization tradeoff problem. *Knowledge-Based Systems, 94*, 132–145.
Tran, D. H., Luong-Duc, L., Duong, M. T., Le, T. N., & Pham, A. D. (2018). Opposition multiple objective symbiotic organisms search (OMOSOS) for time, cost, quality and work continuity tradeoff in repetitive projects. *Journal of Computational Design and Engineering, 5*(2), 160–172.
Vaessens, R. J., Aarts, E. H., & Lenstra, J. K. (1998). A local search template. *Computers & Operations Research, 25*(11), 969–979.
Verma, S., & Mukherjee, V. (2018). Investigation of static transmission expansion planning using the symbiotic organisms search algorithm. *Engineering Optimization, 50*(9), 1544–1560.
Verma, S., Saha, S., & Mukherjee, V. (2017). A novel symbiotic organisms search algorithm for congestion management in deregulated environment. *Journal of Experimental & Theoretical Artificial Intelligence, 29*(1), 59–79.
Vincent, F. Y., Redi, A. P., Yang, C. L., Ruskartina, E., & Santosa, B. (2017). Symbiotic organisms search and two solution representations for solving the capacitated vehicle routing problem. *Applied Soft Computing, 52*, 657–672.
Wang, D., Zhang, W. H., & Jiang, J. S. (2004). Truss optimization on shape and sizing with frequency constraints. *AIAA journal, 42*(3), 622–630.
Weise, T. (2011). *Global optimization algorithms-theory and application* (third edition) Online E-Book http://www.it-weise.de/projects/bookNew.pdf .
Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE transactions on evolutionary computation, 1*(1), 67–82.
Wu, H., Zhou, Y., & Luo, Q. (2018). Hybrid symbiotic organisms search algorithm for solving 0-1 knapsack problem. *International Journal of Bio-Inspired Computation, 12*(1), 23–53.
Wu, H., Zhou, Y., Luo, Q., & Basset, M. A. (2016). Training Feedforward Neural Networks Using Symbiotic Organisms Search Algorithm. *Computational intelligence and neuroscience* Article ID 9063065, 14 pages. doi:10.1155/2016/9063065.
Yang, X. S. (2010a). A new metaheuristic bat-inspired algorithm. Nature inspired cooperative strategies for optimization (NICSO 2010), 65–74.
Yang, X. S. (2010b). Firefly algorithm, stochastic test functions and design optimisation. *International Journal of Bio-Inspired Computation, 2*(2), 78–84.
Yang, X. S. (2010c). *Engineering optimization: An introduction with metaheuristic applications*. Hoboken new jersey: John Wiley & Sons.
Yang, X. S. (2012). Flower pollination algorithm for global optimization. In *UCNC* (pp. 240–249).
Yang, X. S. (2014). Cuckoo search and firefly algorithm: Overview and analysis. In *Cuckoo search and firefly algorithm* (pp. 1–26). Springer, Cham.
Yang, X. S. (2013). Bat algorithm and cuckoo search: A tutorial. In *Artificial Intelligence, Evolutionary Computing and Metaheuristics* (pp. 421–434). Berlin, Heidelberg: Springer.
Yang, X. S., & He, X. (2013). Bat algorithm: Literature review and applications. *International Journal of Bio-Inspired Computation, 5*(3), 141–149.
Yang, X. S., & Deb, S. (2009). Cuckoo search via Lévy flights. In *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on* (pp. 210–214). IEEE.
Yang, X. S., & Deb, S. (2010). Engineering optimisation by cuckoo search. *International Journal of Mathematical Modelling and Numerical Optimisation, 1*(4), 330–343.
Zamani, M. K. M., Musirin, I., & Suliman, S. I. (2017). Symbiotic Organisms Search Technique for SVC Installation in Voltage Control. *Indonesian Journal of Electrical Engineering and Computer Science, 6*(2), 318–329.
Zhang, B., Sun, L., Yuan, H., Lv, J., & Ma, Z. (2016). An improved regularized extreme learning machine based on symbiotic organisms search. In *Industrial Electronics and Applications (ICIEA), 2016 IEEE 11th Conference on* (pp. 1645–1648). IEEE.
Zheng, Y. J. (2015). Water wave optimization: A new nature-inspired metaheuristic. *Computers & Operations Research, 55*, 1–11.
Zuo, W., Bai, J., & Li, B. (2014). A hybrid OC–GA approach for fast and global truss optimization with frequency constraints. *Applied Soft Computing, 14*, 528–535.