Chapter 13
Supply Chain Network Design Using an Enhanced Hybrid Swarm-Based Optimization Algorithm

Baris Yuce
Cardiff University, UK

Ernesto Mastrocinque
University of London, UK

ABSTRACT

Supply chain network design is one of the most important strategic issues in operations management. The main objective in designing a supply chain is to keep the cost as low as possible. However, the modelling of a supply chain requires more than single-objective such as lead-time minimization, service level maximization, and environmental impact maximization among others. Usually these objectives may cause conflicts such as increasing the service level usually causes a growth in costs. Therefore, the aim should be to find trade-off solutions to satisfy the conflicting objectives. The aim of this chapter is to propose a new method based on a hybrid version of the Bees Algorithm with Slope Angle Computation and Hill Climbing Algorithm to solve a multi-objective supply chain network design problem. A real case from the literature has been selected and solved in order to show the potentiality of the proposed method in solving a large scale combinatorial problem.

INTRODUCTION

The supply chain (SC) is a complex system (Mastrocinque et al., 2014) aiming to move products or services from suppliers to customers and involves people, technologies, information, materials among others. At the strategic level, supply chain network design is a crucial decision affecting the future success of the business (Lambiase et al., 2013). A supply network can have different configuration depending on the criteria used during the design stage.

DOI: 10.4018/978-1-7998-0945-6.ch013

Optimization is a powerful tool employed by decision-makers to calculate the optimal value of decision variables in order to minimize/maximize certain objective functions. It has been widely used for solving production related problems (Vasant, 2010; Vasant & Barsoum, 2009; Vasant & Basoum, 2010). The most common supply chain optimization model is based on the evaluation of a single objective function, usually an economic index such as total cost of the supply chain or profit. However, reality can be more complex than reduce the modelling of a supply network to a single-objective In fact other objectives might be considered when it comes to design the configuration of a supply chain such as lead-time minimization, environmental impact minimization, service level maximization, among others (Voudouris, 1996). Typically, these objectives may be in contrast between them such as increasing the service level usually causes a growth in costs. Therefore, the aim of a multi-objective supply network design should be to find trade-off solutions in order to satisfy the conflicting objectives.

In multi-objective optimization there is no single optimum solution, but there are different solutions forming the so-called Pareto set. Pareto solutions are a set of trade-offs between different objectives and they are non-dominated solutions which means there is no other solution which would improve an objective without causing a worsening in at least one of the other objectives (Deb, 2001).

In literature, several methods have been proposed to solve supply chain design problems to get the Pareto optimal solutions. Especially, evolutionary and meta-heuristics algorithms such as genetic algorithms (Altiparmak et al., 2006), particle swarm optimization algorithm (Mahnam et al., 2009), ant colony algorithm (Moncayo-Martinez & Zhang, 2011), the Bees Algorithm (Mastrocinque et al., 2013) among others, have proven to be a valid tool in order to solve the multi-objective supply chain network design.

The Bees Algorithm has proven to be a powerful method for multi-objective supply network design (Yuce et al., 2015). Recently an enhanced version of the Bees Algorithm (BA) using Slope Angle Computation and Hill Climbing Algorithm (SACHCA) has been proposed (Yuce et al., 2015a) and successfully applied for single-objective problems such as continuous type functions and single-objective job shop scheduling.

In this chapter the authors apply the SACHCA-based Bees Algorithm to a multi-objective optimization problem such as supply chain design in order to test its performance on a large scale combinatorial problem and give to the decision maker an alternative tool. In order to explore the behavior of the SACHCA-based Bees Algorithm, it has been used to solve a real-world problem such as the notebook supply network design (Graves & Willems, 2005) considering two conflicting objectives such as minimizing the total cost and the total lead time of the network.

This chapter is organized as follows: the background section revises the literature on the multi-objective supply chain optimization and the Bees Algorithm. In section 1, the enhanced version of the Bees Algorithm with Slope Angle Computation and Hill Climbing Algorithm is presented. In section 3 the notebook supply chain case study from the literature is explained. In section 4 the optimization results are presented and discussed. Finally future research direction and conclusion are given.

BACKGROUND

In literature, several multi-objective supply chain design models have been proposed. In (Altiparmak et al., 2006) a facility location problem of a four echelons supply chain (suppliers, plants, distribution centers-DCs and customers) have been proposed, where the objectives are to minimize the total cost, maximize customer services and the capacity utilization balance for DCs using a genetic algorithm based approach. In Che and Chiang (2010) a supply chain design is based on the supplier selection, product assembly and distribution system using a modified Pareto genetic algorithm to minimize the total cost, the delivery time and maximize the quality. In (Liao et al., 2011) a multi-objective location–inventory problem has been investigated using a multi-objective evolutionary algorithm based on the non-dominated sorting genetic algorithm II (NSGAII) in order to minimize total costs, maximize the volume fill rate and the responsiveness level . In Yeh and Chuang (2011) a planning model for green partner selection involving four objectives such as cost, time, product quality and green appraisal score, has been developed; this model employed two types of genetic algorithms to solve the multi-objectives problem by means of the weighted sum approach to find the set of Pareto optimal solutions. In Pishvaee and Razm (2012) authors have developed a multi-objective fuzzy mathematical programming model for a forward/reverse supply chain minimizing the total cost and the environmental impact. A multi-product, multi-stage, and multi-period scheduling model is proposed in (Chen & Lee, 2004) to deal with multiple goals for a multi-echelon supply chain network with uncertain market demands and product prices; a two-phase fuzzy decision-making method is presented to maximize the participants' expected profits, average safe inventory levels, average customer service levels and robustness of selected objectives to demand uncertainties. A bi-objective optimization approach to designing and planning of supply chain is proposed in (Pinto-Varela et al., 2011) in order to maximize the annual profit and minimize the environmental impact. In (Sazvar et al., 2014) a bi-objective stochastic programming model for a centralized green supply chain with deteriorating products have been proposed in order to minimize total supply chain costs and environmental impacts at the same time. In (Sarrafha et al., 2015) a supply chain design model minimizing the total SC costs as well as minimizing the average tardiness of product to DCs have been proposed and solved with a multi-objective biogeography based optimization (MOBBO).In (Pasandideh et al., 2015) a bi-objective optimization of a multi-product multi-period three-echelon supply chain network consisting of manufacturing plants, distribution centers each with uncertain services, and customer nodes is proposed. The two objectives are minimization of the total cost while maximizing the average number of products dispatched to customers.

Other models have also been proposed based on swarm inspired algorithm to solve the supply chain design problem. In (Mahnam et al., 2009) an inventory model for an assembly supply chain network with fuzzy demand has been proposed considering total cost and fill rate and solved using a hybridization of multi-objective particle swarm optimization and simulation optimization. In (Che, 2012) an optimization mathematical model integrating cost and time criteria has been solved using a modified particle swarm optimization method (MEDPSO) for solving a multi-echelon unbalanced supply chain planning problem. The results indicated that the MEDPSO method can obtain a better quality solution compared to classical GA and PSO. Furthermore, another swarm based optimization model based on the Ant Colony optimization method has been developed in order to configure a supply network with different options for each stage minimizing the total cost and the total lead time of the supply chain (Moncayo-Martinez & Zhang, 2011). The same problem has been solved using the Bees Algorithm and its enhanced version based on adaptive neighborhood search and abandonment strategy (Yuce et al., 2014).

The Bees Algorithm has been used successfully to solve many problems including mechanical design, job shop scheduling, supply chain optimization, robot path planning, chemical engineering problems, assembly problems and several other applications (Ang et al., 2013a; Ang et al., 2013b; Castellani et al., 2012; Pham et. al., 2008; Pham et. al., 2007a; Pham et. al., 2007b; Pham et. al., 2007c; Pham et. al., 2007d; Pham et. al., 2007e and Pham et. al., 2007f; Mastrocinque et. al., 2013; ; Lien and Cheng, 2014; Yuce et. al., 2014). Despite these several successful applications of the Bees Algorithm, a careful examination shows that the algorithm needs to be more robust and sensitive to a particular problem. Several attempts have been made in the past to improve the Bees Algorithm using methods such as early neighborhood search and efficiency based recruitment (Pham et al., 2012), adaptive neighborhood search and site abandonment strategy (Yuce et. al., 2013), combination of adaptive enlargement and reduction of the search neighborhood (Azfanizam et al., 2014) developing new local search strategies (Ahmad, 2012), tuning the algorithms parameters (Otri, 2011), introducing new parameters (Pham and Koc, 2011; Pham and Ghanbarzadeh, 2007), combining with other optimization algorithms (Packianather et al., 2014; Sholedolu, 2009,) novel initialization based on the patch concept and Levy flight distribution (Hussein et al., 2014).

Recently another hybrid version of the Bees Algorithm (BA) using Slope Angle Computation and Hill Climbing Algorithm (SACHCA) has been proposed (Yuce et al., 2015a). The aim of the SACHCA is to strengthen the neighborhood search. The SACHCA-base BA has been successfully applied for single-objective problems such as continuous type functions and single-objective job shop scheduling.

Since there is no algorithm which can find the best solution for all types of optimization problems according to the no-free lunch theorem (Wolpert & Macready, 1997) the aim of this chapter is to use the SACHCA-based BA to solve a bi-objective supply chain design problem minimizing the total cost and total lead time of a well-known problem such as the notebook supply network (Graves & Willems, 2005).

THE BEES ALGORITHM AND SLOPE ANGLE COMPUTATION AND HILL CLIMBING ALGORITHM

The Bees Algorithm

The Bees Algorithm (BA) is a swarm based algorithm proposed by (Pham et. al. 2005; 2006a). Swarm based algorithms emulate the behavior of animals in swarm such as Ant Colonies, Bee Colonies, Bird Flocking, Animal Herding, Fish Schooling among others (Beni, 1989).

The BA has been inspired from the foraging behavior of the honey bees. A swarm of honey bees consists of a queen bee and thousands of worker bees which are allocated to do different works in a colony such as cleaning their nest, foraging and constructing comb (Seeley, 1995). When a scout bee finds a food source, it returns to the hive and performs special ritual which is a communicational movement, known as the waggle dance in order to inform the colony (Seeley, 1995, Gould and Gould, 1988). The waggle dance contains information about direction, distance and quality of the flower patch found by the bee (Talbi, 2009). After the waggle dance, a number of bees is allocated for that particular food source. Promising patches should have more bees than the others. The recruited bees evaluate the related patch and look around of the patch as well as share this information with their peers in the hive. The same process happens for all patches emulating local search. After the recruitment stage the scout bees will continue searching for other promising patches in a random way emulating the global search and whenever they find another patch this information will be given to the colony randomly (Von Frisch, 1955). The Bees Algorithm involves several parameters given in Table 1.

The steps of the BA are: placing the “n” scout bees on the search space; in step 2 fitness values of the visited patches are evaluated. Step 3 regards the selection of the best patches with respect to their fitness value where these selected best patches will be split into two groups containing more scout bees to the elite patches ‘e’, and less scout bees to the non-elite best patches ’m-e’ in step 4. Step 5 is the

Table 1. Bees Algorithm parameters

Parameter	Symbol
Number of scout bees in the selected patches n	*n*
Number of best patches out of the selected patches	*m*
Number of elite patches out of the selected best patches	*e*
Number of recruited bees in the elite patches	*nep*
Number of recruited bees in the non-elite best patches	*nsp*
The size of neighborhood for each patch	*ngh*
Number of iterations MaxIter	*MaxIter*
Difference between fitness values of consecutive iterations	*Error*

neighborhood search in the patches given in steps 3 and 4. In step 6, the remainder bees, generated in the initial population, will be recruited for the random search to find better random solutions. In step 7 the random patches' fitness values will be evaluated and this process will continue until one of the stopping criteria is met.

The procedure described above equips the algorithm to combine exploitative neighborhood search with explorative global search enabling effective location of the globally optimal solution to a problem (Yuce, 2012). Due to the powerful search capability of the algorithm, the basic BA was successfully utilized to solve different types of optimization problems such as job scheduling (Pham et. al., 2007a), forming manufacturing cells (Pham et. al., 2007b), data clustering (Pham et. al., 2007c), tuning a fuzzy logic controller (Pham et. al., 2007d), training neural networks (Pham et. al., 2006b), finding the multiple feasible solutions to preliminary design problems (Pham et. al., 2007e) and design of mechanical components (Pham et. al., 2007f). The improvements made to the basic BA with slope angle computation and hill climbing algorithm is given in the following section.

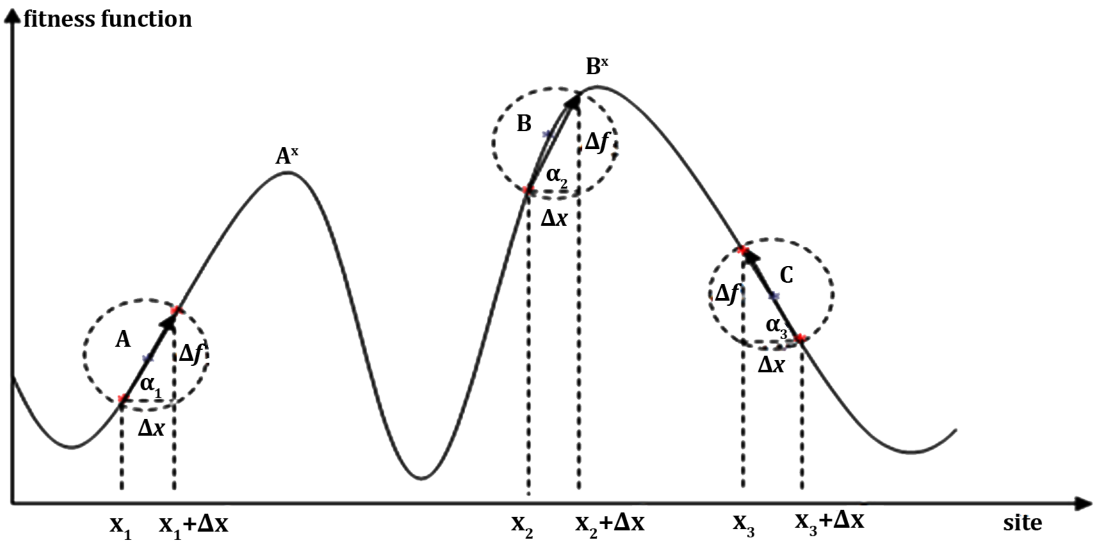
The Bees Algorithm with Slope Angle Computation and Hill Climbing Algorithm

The enhancement to the basic Bees Algorithm (BA) based on the Slope Angle Computation and Hill Climbing Algorithm (SACHCA) was recently proposed by (Yuce et al., 2015) in order to enhance the blind local search process (Yuce, 2012). The Hill Climbing Algorithm (HCA) is an iterative single element-based local search algorithm, also known as Gradient Ascent / Descent algorithm. Although the local minimum of an optimization problem can be found by the HCA the global optimum is not guaranteed (Grosan and Abraham, 2011).

The SACHCA-based improved BA deals with location process of the best sites. The inclination of the current sites is evaluated by slope angle computation. In general, the promising location will be far from the current position if the slope angle is close to 90º, while the current location will be close to a local optimum point if the slope angle is close to 0º. The direction of the local optimum can be determined according to the slope angle orientation.

The slope angle is computed using the first order numerical derivation. The numerical derivation of each site is calculated from its neighborhood. The two end points in the vicinity of the neighborhood are used to compute the numerical derivation. The central difference method is utilized for numerical derivation (see Equation 1).

Figure 1. Slope angle for three selected sites (adapted from Yuce et al., 2015)



$$Slope\,angle = f'\left(x\right) = \frac{f\left(x+\dfrac{\Delta x}{2}\right) - f\left(x-\dfrac{\Delta x}{2}\right)}{\Delta x} \tag{1}$$

If the slope angle is very steep, then the promised location will be more likely to be far from the selected site but when the slope angle is close to zero, the promised location is very close to the selected site, as shown in Figure 1.

Figure 1 shows three selected sites. Around site A, the angle direction is towards site A^x, which is a local optimum. The direction of sites B and C are towards the site B^x, which is another local optimum. At the end of the searching process, all the local optima will be sorted and the biggest will be selected as the global optimum which in Figure 1 is site B^x .The local search process was accomplished with the use of the Hill Climbing Algorithm, according with Equation 2.

$$x\left(i+1\right) = x\left(i\right) + h\nabla f \tag{2}$$

where i is the iteration number, $x(i)$ is the current position, $x(i+1)$ is the next position, h is the incremental size, ∇f is the gradient of the current position. The pseudo-code of the algorithm is the following (Yuce et al., 2015):

Generate the initial population size as n, set the best patch size as m, set the elite patch size as e, set the number of forager bees around of elite as *nep*, set the number of forager bees around of non-elite best patches as *nsp*, set the step size for HC algorithm as h, set the angle limit as *angle_limit*, set the "number of the waiting time for HC" algorithm as *HC_time_limit*, set the neighborhood size as *ngh*, set the maximum iteration number as *MaxIter*, and set the error limit as *Error*.

i= 0, *time*=0;
slope_angle(1:*m*)=0;
Generate initial population;
Evaluate Fitness Value of initial population;
Sort the initial population based on the fitness result.
While $i \leq MaxIter$ or $FitnessValue_i - FitnessValue_{i-1} \geq Error$
    i= i+l;
Select the elite patches and non-elite best patches for neighborhood search.
Recruit the forager bees to the elite patches and non-elite best patches.
Evaluate the fitness value of each patch; Sort the results based on their fitness.
For k=1:m
Calculate ..
While *slope_angle*(*k*)>*angle_limit* and *time*≤*HC_time_limit*

$$x\left(i+1,k\right)=x\left(i,k\right)+h\nabla f\left(x\left(i,k\right)\right)$$

Evaluate Fitness value for each position
End
Record all the local optimum sites found and sort them (end of the neighbourhood search).
Allocate the rest of the bees for global search to the non-best locations;
Evaluate the fitness value of non-best patches;
Sort the fitness values and positions;
Run the algorithm until termination criterions are met.
End

SUPPLY NETWORK DESIGN CASE STUDY AND METHODOLOGY

The notebook supply chain network design case analyzed in this chapter is based on the Graves and Willems (2005) case, readapted by Moncayo-Martinez and Recio (2014). The supply chain in this case, has been represented as a Bill of Materials (BOM) where each node could represent a supplying manufacturing, assembly, delivery and so on. Each of this stage could be performed by different options. In fact a part could be supplied by more than a supplier, an assembly operation could be performed in different plants, a manufacturing operations could use different technologies and a different transportation modes can be used chosen to reach the customers. Moreover, each of this option for each stage, has a different cost and needs different time to be performed. Usually, cost and time are conflicting objective, because if an option has a higher cost than another, usually it need less time to be performed. For instance, if I want to be delivered a part or raw material in a shorter amount of time, I might have to pay more.

The notebook supply network is composed by several stages going from the raw materials supply, to assembly of finished products and delivery to the markets. The structure of the network is shown in Figure 2. The network encompasses 17 stages and each of them can be performed by different options n. A cost and a lead time are associated with each option (Table 2). cost and lead time associated to each stage is are shown in Table 1. The number of possible different configuration of the supply network is given by $\prod_{s=1}^{S} n_s$ which is equal to 1990656 in the proposed case.

The aim of the optimization is to find the optimal configuration of the notebook supply chain minimizing the total cost and the total lead time of the network given by Equations 3 and 4 respectively, where d_i represents the demand of products for each stage, Θ the period of time considered, C_{sn} is the cost of option n for stage s, T_{sn} the lead time of option n for stage s and finally x_{sn} is the binary decision variable equal to 1 if option n for the stages s is selected, 0 otherwise. Only one option for each stage can be selected, according to the constraint in Equation 5. In the case analyzed, Θ=360 days, d_{15}=200, d_{16}=125, d_{17}=75 per day.

$$TCost = \Theta \sum_{s=1}^{S} \left(d_s \sum_{n=1}^{N_s} C_{sn} x_{sn} \right) \tag{3}$$

$$TLeadtime = \max_{s=15,16,17} LT_s \tag{4}$$

with

$$LT_s = \sum_{n=1}^{N_s} T_{sn} x_{sn} + \max_{k=input\ nodes\ to\ stage\ s} \left(LT_k \right)$$

$$\sum_{n=1}^{N_s} x_{sn} = 1 \quad for\ s \in S \tag{5}$$

Figure 2. Notebook supply chain structure

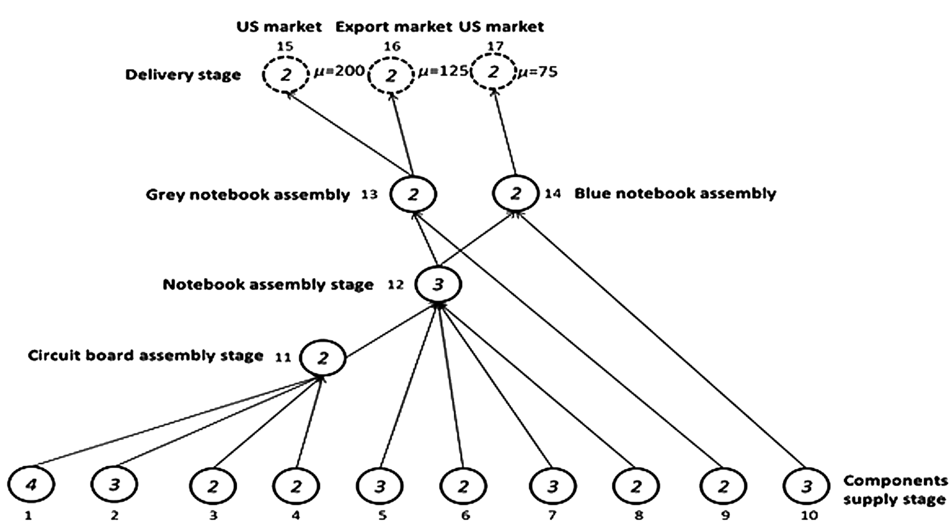


Table 2. Cost and lead time values for each stage option in the supply chain

Stage s	*Option n*	*C*	*T*
1	1 2 3 4	130.00 133.25 134.91 136.59	40 20 10 0
2	1 2 3	200.00 202.50 205.03	20 10 0
3	1 2	155.00 156.93	10 0
4	1 2	204.00 200.00	10 0
5	1 2 3	300.00 325.00 350.00	60 10 5
6	1 2	225.00 240.00	70 30
7	1 2 3	40.00 45.00 55.00	60 20 10
8	1 2	200.00 205.00	30 10
9	1 2	5.00 5.50	40 15
10	1 2 3	5.00 5.50 7.50	40 15 5
11	1 2	120.00 150.00	20 5
12	1 2 3	120.00 132.00 140.00	5 2 1
13	1 2	20.00 30.00	2 1
14	1 2	20.00 30.00	2 1
15	1 2	12.00 20.00	5 1
16	1 2	15.00 30.00	15 2
17	1 2	12.00 20.00	5 1

The basic and the SACHCA-based Bees Algorithm have been applied to solve the bi-objective notebook supply chain design. Four different combinations of the BA parameters (Table 3) are tested in order to obtain the Pareto front line. All the experiments have been carried out with the number of iterations fixed to 1000, ngh=2, sc=1, h=1, angle limit=0.5, HC time limit=1000. In all tests the algorithm is run 100 times and results are updated according to previous Pareto solutions after each run in order

Table 3. Bees Algorithm parameters values for the four combinations tested

Parameters					
Combination	*n*	*m*	*e*	*nep*	*nsp*
1	10	5	1	3	2
2	25	10	2	6	4
3	50	15	5	12	8
4	100	30	10	24	16

to increase robustness. The weighted sum approach has been used to generate the Pareto solutions. The bi-objective problem has been reduced to a single objective problem aiming to find the solution that minimize with the fitness function expressed by Equation 6.

$$fitness = w_1 normTCost + w_2 normTLeadtime \tag{6}$$

where normTCost and normTLeadtime are the normalized solutions and w_1 and w_2 are the weights where the summation is equal to 1. For this case a w_1=w_2=0.5.

In order to evaluate the Pareto frontline, two evaluation criteria (Zitzler et al., 2000; Tan et al., 2002) as coverage of two sets and spacing are considered. The coverage of two sets (Mallipeddi et al., 2010): let $X', X' \subseteq X$ be two sets of decision vectors. The function C maps the ordered pairs $\left(X', X''\right)$ to the interval [0, 1] as follows:

$$C\left(X', X''\right) > \frac{\left|\left\{a'' \in X''; a' \in X' : a' \le a''\right\}\right|}{\left|X''\right|}.$$

The value $C\left(X', X''\right) = 1$ means that all solutions in X'' are dominated by or equal to solutions in X'. The opposite, $C\left(X', X''\right) = 0$, represents the situations where none of the solutions in X'' are covered by the set X'. Note that both $C\left(X', X''\right)$ and $C\left(X'', X'\right)$ have to be considered since $C\left(X', X''\right)$ is not necessarily equal to $1 - C\left(X'', X'\right)$. Spacing (Zhao & Suganthan, 2010) is a measure of relative distance between consecutive solutions in a non-dominated set Q as follows:

$$S = \sqrt{\frac{1}{|Q|}\sum_{i=1}^{|Q|}\left(d_i - \bar{d}\right)^2}$$

with Q size of the set and $\bar{d}$ average of the d_i, measuring the distance between the solution i and the nearest solution in the objective space:

$$d_i = \min\left\{\sum_{m=1}^{M}\left(f_m^i - f_m^k\right)^2 \mid k \in Q \;\; k \neq i\right\}.$$

A value of this metric equal to 0 would mean that each point of the Pareto front is equally spaced from the other.

RESULTS

As shown from the Pareto frontlines in Figure 3 and the results summarized in Table 4, SACHCA BA found more solutions than the BA for all the tested combinations of parameters. Both the algorithms found 34 day as minimum total lead time, while the minimum value of the total cost 187834680 was found by the SACHCA BA. The minimum total cost value found by the SACHCA BA was always lower than the BA one for all the parameters combinations.

When it comes to the spacing metric, only in combination 3 the SACHCA BA Pareto frontline had a lower value compared to the BA, which means the BA Pareto solutions were better distributed in the front. However, for all the parameters combinations, C(SACHCA,BA) had always a much higher value than the C(BA,SACHCA), which means the SACHCA BA solutions dominated the BA ones.

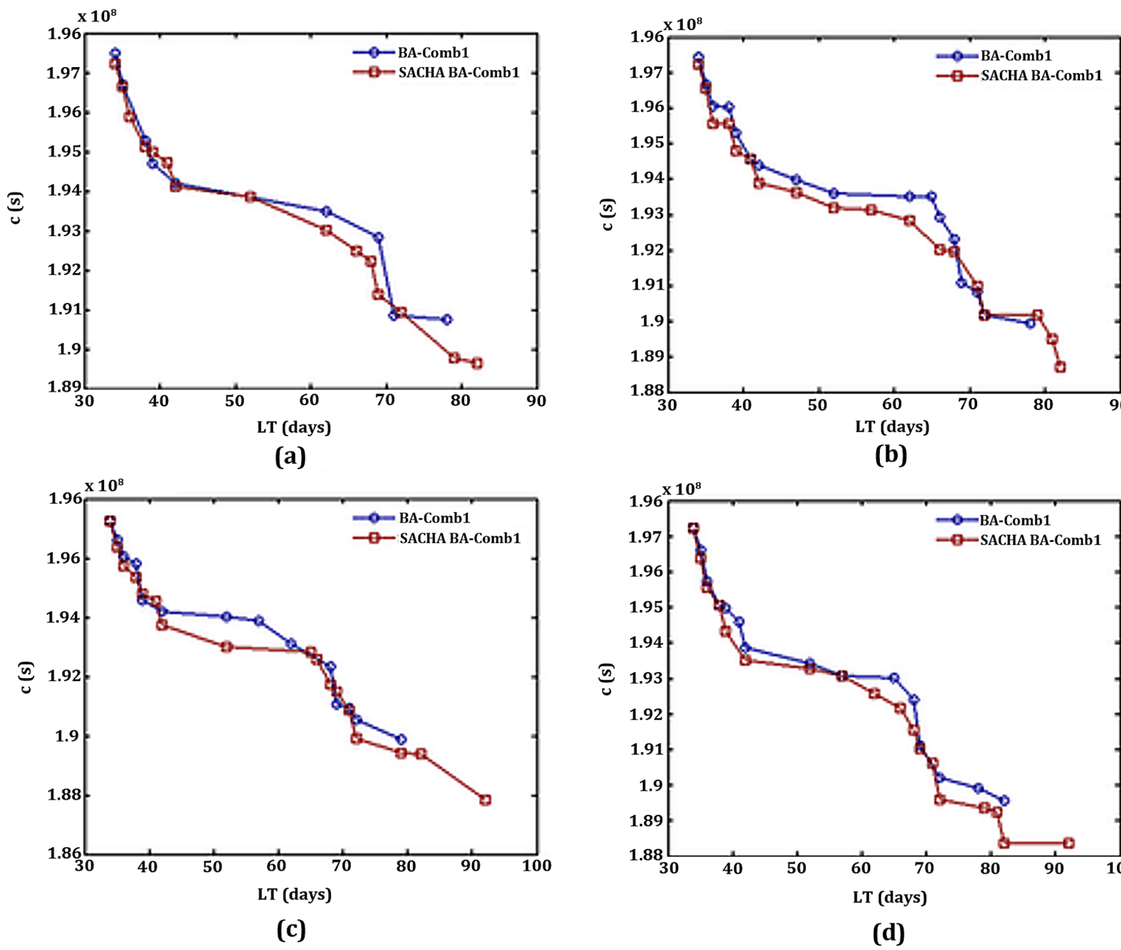
Overall both the algorithms performed very well on the complex combinatorial problem selected, and there was no notable differences in the results obtained with both algorithms for each parameters combination tested.

From a managerial perspective, since the SACHCA BA allowed to find more Pareto solutions dominating the BA solutions, it gives to the decision maker more and better options to choose from regarding the supply network design problem.

Table 4. Results and performance of the two algorithm tested on the notebook supply chain case

Comb	Algorithm	No of Pareto solutions	Min TLeadtime	Min TCost	Spacing	Coverage of two sets
1	Basic BA	9	34	190756440	0.037	**C(BA,SACHCA) 0.067**
	SACHCA BA	15	34	189638280	0.042	**C(SACHCA,BA) 0.667**
2	Basic BA	17	34	189943740	0.031	**C(BA,SACHCA) 0.055**
	SACHCA BA	18	34	188716500	0.025	**C(SACHCA,BA) 0.705**
3	Basic BA	14	34	189867780	0.043	**C(BA,SACHCA) 0.118**
	SACHCA BA	17	34	187834680	0.055	**C(SACHCA,BA) 0.571**
4	Basic BA	15	34	189554940	0.031	**C(BA,SACHCA) 0**
	SACHCA BA	18	34	188369280	0.031	**C(SACHCA,BA) 0.600**

Figure 3. Pareto front lines obtained with the two algorithms tested on the notebook supply chain problem for combination 1 (a), 2 (b), 3 (c) and 4 (d) of the parameters



FUTURE RESEARCH DIRECTIONS

The work presented in this chapter can be developed for future research from different perspectives. One possible development regards testing the proposed SACHCA BA on a more complex supply chain optimization problem considering uncertainty in the parameters such as for instance demand or lead time in order to make the model more fitting to reality.

Another possible future research would consist in comparing the SACHA BA and the BA to other well-known optimization algorithms such as genetic, ant colony, particle swarm and so on. In fact, the SACHA BA has already shown to provide better performance than other well-known algorithm when applied to solve continuous type optimization functions. However it would be interesting to compare their performance on a multi-objective combinatorial problem such as the supply chain design.

Also it would be interesting to use other hybrid version of the Bees Algorithm in order to benefit from the strength of different approaches. Recently, a hybrid Bees Algorithm with Genetic Algorithm has been proposed in order to benefit from a strong random search form the GA and local search from the BA, reducing the inconvenient of stacking in local minima. This hybrid algorithm was applied to solve the single-machine job shop scheduling problem with Earliness and Tardiness optimization (Packianather et al, 2014). The results were very promising and it would be interesting in future to test it on the combinatorial supply chain problem.

Finally, as future research the SACHCA BA could be tested on other well-known operational research problems such as for instance the traveling salesman problem.

CONCLUSION

In this chapter, an enhanced version of the Bees Algorithm with Slope Angle Computation and Hill Climbing Algorithm has been proposed to solve a bi-objective supply chain design problem. The complex combinatorial problem deals with the configuration of a notebook supply network in order to minimize two objective functions simultaneously such as the total cost and total lead-time of the network. The SACHCA-BA and the basic BA have been applied to solve the above mentioned problem with four different combinations of the algorithm parameters in order to find the Pareto solutions. The results showed the efficiency of the proposed algorithm as a more powerful tool to find more and better Pareto solution for the supply chain problem compared with the basic Bess Algorithm. In fact the minimum total cost found by the SACHCA-BA was lower than the one found by the BA for all the parameters combinations tested. Moreover, the coverage of two set C showed how the Pareto solutions obtained with the SACHCA BA dominate the ones obtained with the basic BA. This allows the decision maker to select a supply chain configuration where the tradeoff between cost of final products and delivery time is more balanced. However, the basic BA showed to produce solutions being better distributed in the Pareto front line as confirmed by the lower values of the spacing metric.

REFERENCES

Ahmad, S. A. (2012). *A Study of Search Neighborhood in the Bees Algorithm*. (PhD Thesis). Cardiff University.

Altiparmak, F., Gen, M., Lin, L., & Paksoy, T. (2006). A genetic algorithm approach for multi-objective optimization of supply chain networks. *Computers & Industrial Engineering*, *51*(1), 196–215. doi:10.1016/j.cie.2006.07.011

Ang, M. C., Ng, K. W., & Pham, D. T. (2013b). Combining the Bees Algorithm and shape grammar to generate branded product concepts. *Proceedings of the Institution of Mechanical Engineers. Part B, Journal of Engineering Manufacture*, *227*(12), 1860–1873. doi:10.1177/0954405413494922

Ang, M. C., Ng, K. W., Pham, D. T., & Soroka, A. (2013a). Simulations of PCB Assembly Optimization Based on the Bees Algorithm with TRIZ-Inspired Operators. In Advances in Visual Informatics (pp. 335-346).

Azfanizam, A. S., Pham, D. T., & Faieza, A. A. (2014). Combination of Adaptive Enlargement and Reduction in the Search Neighborhood in the Bees Algorithm. *In Applied. Mechanics of Materials*, *564*, 614–618. doi:10.4028/www.scientific.net/AMM.564.614

Beni, G., & Wang, J. (1989). Swarm Intelligence in Cellular Robotic Systems. *Advanced Workshop on Robots and Biological Systems*, Tuscany, Italy.

Castellani, M., Pham, Q. T., & Pham, D. T. (2012). Dynamic optimization by a modified bees algorithm. *Proceedings of the Institution of Mechanical Engineers. Part I, Journal of Systems and Control Engineering*, *226*(7), 956–971. doi:10.1177/0959651812443462

Che, Z. H. (2012). A particle swarm optimization algorithm for solving unbalanced supply chain planning problems. *Applied Soft Computing*, *12*(4), 1279–1287. doi:10.1016/j.asoc.2011.12.006

Che, Z. H., & Chiang, C. J. (2010). A modified Pareto genetic algorithm for multi-objective build-to-order supply chain planning with product assembly. *Advances in Engineering Software*, *41*(7), 1011–1022. doi:10.1016/j.advengsoft.2010.04.001

Chen, C. L., & Lee, W. C. (2004). Multi-objective optimization of multi-echelon supply chain networks with uncertain product demands and prices. *Computers & Chemical Engineering*, *28*(6), 1131–1144. doi:10.1016/j.compchemeng.2003.09.014

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

Gould, J. L., & Gould, C. G. (1988). *The Honey Bee*. New York: Scientific American Library.

Graves, S. C., & Willems, S. P. (2005). Optimizing the supply chain configuration for new products. *Management Science*, *51*(8), 1165–1180. doi:10.1287/mnsc.1050.0367

Grosan, C., & Abraham, A. (2011). *Intelligent Systems: A Modern Approach*. Berlin: Springer. doi:10.1007/978-3-642-21004-4

Hussein, W. A., Sahran, S., & Abdullah, S. N. H. S. (2014). Patch-Levy-based Initialization Algorithm for Bees Algorithm. *Applied Soft Computing*, *23*, 104–121. doi:10.1016/j.asoc.2014.06.004

Lambiase, A., Mastrocinque, E., Miranda, S., & Lambiase, A. (2013). Strategic planning and design of supply chains: A literature review. *Int. J. Eng. Bus. Manag*, *5*(49), 1–11. doi:10.5772/56858

Liao, S. H., Hsieh, C. L., & Lai, P. J. (2011). An evolutionary approach for multi-objective optimization of the integrated location–inventory distribution network problem in vendor-managed inventory. *Expert Systems with Applications*, *38*(6), 6768–6776. doi:10.1016/j.eswa.2010.12.072

Lien, L. C., & Cheng, M. Y. (2014). Particle bee algorithm for tower crane layout with material quantity supply and demand optimization. *Automation in Construction*, *45*, 25–32. doi:10.1016/j.autcon.2014.05.002

Mahnam, M., Yadollahpour, M. R., Famil-Dardashti, V., & Hejazi, S. R. (2009). Supply chain modeling in uncertain environment with bi-objective approach. *Computers & Industrial Engineering*, *56*(4), 1535–1544. doi:10.1016/j.cie.2008.09.038

Mallipeddi, R., Mallipeddi, S., & Suganthan, P. N. (2010). Ensemble strategies with adaptive evolutionary programming. *Information Sciences*, *180*(9), 1571–1581. doi:10.1016/j.ins.2010.01.007

Mastrocinque, E., Yuce, B., Lambiase, A., & Packianather, M. S. (2013). A multi-objective optimization for supply chain network using the Bees Algorithm. *Int. J. Eng. Bus. Manage*, *5*, 1–11. doi:10.5772/56754

Mastrocinque, E., Yuce, B., Lambiase, A., & Packianather, M. S. (2014). A System of Systems Approach to Supply Chain Design. *Applied Mechanics and Materials*, *496*, 2807–2814. doi:10.4028/www.scientific.net/AMM.496-500.2807

Moncayo-Martínez, L. A., & Recio, G. (2014). Bi-criterion optimization for configuring an assembly supply chain using Pareto ant colony meta-heuristic. *Journal of Manufacturing Systems*, *33*(1), 188–195. doi:10.1016/j.jmsy.2013.12.003

Moncayo-Martínez, L. A., & Zhang, D. Z. (2011). Multi-objective ant colony optimization: A meta-heuristic approach to supply chain design. *International Journal of Production Economics*, *131*(1), 407–420. doi:10.1016/j.ijpe.2010.11.026

Otri, S. (2011). *Improving The Bees Algorithm for Complex Optimization Problems*. (PhD Thesis). Cardiff University.

Packianather, M. S., Yuce, B., Mastrocinque, E., Fruggiero, F., Pham, D. T., & Lambiase, A. (2014, August). Novel Genetic Bees Algorithm applied to single machine scheduling problem. In *World Automation Congress (WAC)*, (pp. 906-911). IEEE. 10.1109/WAC.2014.6936194

Pasandideh, S. H. R., Niaki, S. T. A., & Asadi, K. (2015). Optimizing a bi-objective multi-product multi-period three echelon supply chain network with warehouse reliability. *Expert Systems with Applications*, *42*(5), 2615–2623. doi:10.1016/j.eswa.2014.11.018

Pham, D. T., Afify, A. A., & Koç, E. (2007b). *Manufacturing cell formation using the Bees Algorithm*. 3rd International Virtual Conference on Intelligent Production Machines and Systems (IPROMS 2007), Whittles, Dunbeath, Scotland.

Pham, D. T., Castellani, M., & Ghanbarzadeh. (2007e). Preliminary design using the Bees Algorithm. In *Proc Eighth International Conference on Laser Metrology, CMM and Machine Tool Performance, LAMDAMAP*.

Pham, D. T., Castellani, M., & Fahmy, A. A. (2008). Learning the inverse kinematics of a robot manipulator using the Bees Algorithm. *Proceedings of INDIN*, *2008*, 493–498.

Pham, D. T., Darwish, A. H., Eldukhri, E. E., & Otri, S. (2007d). Using the Bees Algorithm to tune a fuzzy logic controller for a robot gymnast. In *Proceedings of IPROMS 2007 Conference*.

Pham, D. T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S., & Zaidi, M. (2005). *The Bees Algorithm. Technical Report: MEC 0501*. Cardiff, UK: Manufacturing Engineering Centre, Cardiff University.

Pham, D. T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S., & Zaidi, M. (2006a). The bees algorithm-a novel tool for complex optimization problems. In *Proceedings of the 2nd Virtual International Conference on Intelligent Production Machines and Systems (IPROMS 2006)* (pp. 454-459). 10.1016/B978-008045157-2/50081-X

Pham, D. T., Koc, E., Lee, J. Y., & Phrueksanant, J. (2007a). *Using the Bees Algorithm to schedule jobs for a machine*. LAMDAMAP, 8th international Conference on Laser Metrology, CMM and Machine Tool Performance, Cardiff, Euspen, UK.

Pham, D. T., Otri, S., Afify, A., Mahmuddin, M., & Al-Jabbouli, H. (2007c). *Data clustering using the Bees Algorithm*. 40th CIRP International Manufacturing Systems Seminar, Liverpool, UK.

Pham, D. T., Packianther, M. S., Imaguliyev, A., & Yuce, B. (2012). Early Neighborhood Search and Efficiency Based Improved The Bees Algorithm. IMS 2012 Conference, Sakarya, Turkey.

Pham, D. T., Soroka, A. J., Koç, E., Ghanbarzadeh, A., & Otri, S. (2007f). Some applications of the Bees Algorithm in engineering design and manufacture. In *Proc Int. Conference on Manufacturing Automation (ICMA 2007)*, Singapore.

Pinto-Varela, T., Barbosa-Póvoa, A. P. F., & Novais, A. Q. (2011). Bi-objective optimization approach to the design and planning of supply chains: Economic versus environmental performances. *Computers & Chemical Engineering*, *35*(8), 1454–1468. doi:10.1016/j.compchemeng.2011.03.009

Pishvaee, M. S., & Razmi, J. (2012). Environmental supply chain network design using multi-objective fuzzy mathematical programming. *Applied Mathematical Modelling*, *36*(8), 3433–3446. doi:10.1016/j.apm.2011.10.007

Sarrafha, K., Rahmati, S. H. A., Niaki, S. T. A., & Zaretalab, A. (2015). A bi-objective integrated procurement, production, and distribution problem of a multi-echelon supply chain network design: A new tuned MOEA. *Computers & Operations Research*, *54*, 35–51. doi:10.1016/j.cor.2014.08.010

Sazvar, Z., Mirzapour Al-e-hashem, S. M. J., Baboli, A., & Akbari Jokar, M. R. (2014). A bi-objective stochastic programming model for a centralized green supply chain with deteriorating products. *International Journal of Production Economics*, *150*, 140–154. doi:10.1016/j.ijpe.2013.12.023

Seeley, T. D. (1995). *The Social Physiology of Honey Bee Colonies: The Wisdom of the Hive*. London: Hardvard University Press.

Sholedolu, M. O. (2009). *Nature-inspired Optimization: Improvements to the Particle Swarm Optimization Algorithm and the Bees Algorithm*. (PhD Thesis). Cardiff University.

Talbi, E. G. (2009). *Metaheuristics: From Design to Implementation*. John Wiley and Sons. doi:10.1002/9780470496916

Tan, K. C., Lee, T. H., & Khor, E. F. (2002). Evolutionary algorithms for multi-objective optimization: Performance assessments and comparisons. *Artificial Intelligence Review*, *17*(4), 251–290. doi:10.1023/A:1015516501242

Vasant, P. (2010). Hybrid simulated annealing and genetic algorithms for industrial production management problems. *International Journal of Computational Methods*, *7*(02), 279–297. doi:10.1142/S0219876210002209

Vasant, P., & Barsoum, N. (2009). Hybrid genetic algorithms and line search method for industrial production planning with non-linear fitness function. *Engineering Applications of Artificial Intelligence*, *22*(4), 767–777. doi:10.1016/j.engappai.2009.03.010

Vasant, P., & Barsoum, N. (2010). Hybrid pattern search and simulated annealing for fuzzy production planning problems. *Computers & Mathematics with Applications (Oxford, England)*, *60*(4), 1058–1067. doi:10.1016/j.camwa.2010.03.063

Von Frisch, K. (1955). *The dancing bees: An Account of the Life and Senses of the Honey Bee*. New York: Harcourt.

Voudouris, V. T., & Consulting, A. (1996). Mathematical programming techniques to debottleneck the supply chain of fine chemical industries. *Computers & Chemical Engineering*, *20*, 1269–1274. doi:10.1016/0098-1354(96)00219-0

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, *1*(1), 67–82. doi:10.1109/4235.585893

Yeh, W. C., & Chuang, M. C. (2011). Using multi-objective genetic algorithm for partner selection in green supply chain problems. *Expert Systems with Applications*, *38*(4), 4244–4253. doi:10.1016/j.eswa.2010.09.091

Yuce, B. (2012). *Novel Computational Technique for Determining DepthUsing Bees Algorithmand Blind Image Deconvolution*. (PhD Thesis). Cardiff University.

Yuce, B., Mastrocinque, E., Lambiase, A., Packianather, M. S., & Pham, D. T. (2014). A multi-objective supply chain optimization using enhanced Bees Algorithm with adaptive neighborhood search and site abandonment strategy. *Swarm and Evolutionary Computation*, *18*, 71–82. doi:10.1016/j.swevo.2014.04.002

Yuce, B., Mastrocinque, E., Packianather, M. S., Lambiase, A., & Pham, D. T. (2015). The Bees Algorithm and Its Applications. In P. Vasant (Ed.), *Handbook of Research on Artificial Intelligence Techniques and Algorithms* (pp. 122–151). Hershey, PA: Information Science Reference; doi:10.4018/978-1-4666-7258-1.ch004

Yuce, B., Packianather, M. S., Mastrocinque, E., Pham, D. T., & Lambiase, A. (2013). Honey bees inspired optimization method: The Bees Algorithm. *Insects*, *4*(4), 646–662. doi:10.3390/insects4040646 PMID:26462528

Yuce, B., Pham, D. T., Packianather, M. S., & Mastrocinque, E. (2015a). An enhancement to the Bees Algorithm with slope angle computation and Hill Climbing Algorithm and its applications on scheduling and continuous-type optimization problem. *Production & Manufacturing Research*, *3*(1), 3–19. doi:10.1080/21693277.2014.976321

Zhao, S. Z., & Suganthan, P. N. (2010). Multi-objective evolutionary algorithm with ensemble of external archives. *International Journal of Innovative Computing, Information, & Control*, *6*(1), 1713–1726.

Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation*, *8*(2), 173–195. doi:10.1162/106365600568202 PMID:10843520

KEY TERMS AND DEFINITIONS

Bees Algorithm: a honey bees foraging behavior inspired optimization algorithm.
Foraging Behavior: behavior regarding searching food sources.
Hill Climbing Algorithm: a local search optimization method.a local search optimization method.
Honey Bee: insect specialized in production and storage of honey.

Multi-Objective Optimization: optimization problem involving more than one objective function to be optimized simultaneously.

Supply Chain Management: management of the whole productive chain from suppliers to customers.

Swarm-Based Optimization: artificial intelligence technique based on the collective behavior of self-organized systems.

Waggle Dance: method used by the scout bees to communicate information about the food source to the rest of the colony.

This research was previously published in the Handbook of Research on Modern Optimization Algorithms and Applications in Engineering and Economics edited by Pandian Vasant, Vo Ngoc Dieu, and Gerhard-Wilhelm Weber; pages 95-112; copyright year 2016 by Engineering Science Reference (an imprint of IGI Global).