Research Journal of Applied Sciences, Engineering and Technology 6(17): 3234-3243, 2013
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: January 15, 2013 Accepted: March 07, 2013 Published: September 20, 2013

Preliminary Evaluation of Artificial Bee Colony Algorithm When Applied to Multi Objective Partial Disassembly Planning

Gianluca Percoco and Marialuisa Diella
Department of Mechanical Engineering, Mathematics and Management Japigia Avenue, 182 (Italian), 70126 Bari, Italy

Abstract: The aim of this study is the first evaluation of the Artificial Bee Colony Algorithm when applied to multi objective partial disassembly planning. Several methodologies have been proposed by academic and industrial researchers for developing and implementing automated disassembly planning and the research literature is very extensive. In particular, nature-inspired heuristic techniques seem to be very promising and performing well to optimize the disassembly planning problem, among them, the Artificial Bee Colony (ABC) approach, which has not yet been tested. The authors propose the implementation of a discrete ABC algorithm to plan the disassembly sequence of products, following these steps: matrix system modelling, multi-objective function and solution search with an ABC algorithm. In particular the study provides details of the algorithm and heuristic rules, inspired by the behaviour of bees during food search, which is a very efficient natural process. Two case studies have been selected and reported to test the efficiency of the algorithm, while further research is required to compare ABC to other efficient heuristics.

Keywords: Artificial bee colony, disassembly planning, optimization

INTRODUCTION

In recent years, the concept of eco-efficiency has been developed, summarized in the expression "greater profit with lower impact", achieving the aims of economic and environmental excellence. It is therefore necessary to produce smaller quantities of waste and to reduce the use of raw materials, saving money and generating profits (Lee and Shih, 2012). In this context, disassembly tends to be costly, because of the increasing complexity and high variety of products. As a consequence disassembly process planning is very important in minimizing maintenance cost and recovering end-of-life products. Because of the complexity of disassembly, numerical optimization methods are often the best choice to plan this process.

In this study a numerical approach to disassembly sequence planning is proposed, based on the optimization strategy of Artificial Bee Colony (ABC) whose power to search optimal assembly sequencing problems (Brajevic and Tuba, 2012; Akay and Karaboga, 2010) is known. In the case of partial disassembly sequencing, the research space is greater than that of simple permutations of a sequence, because the algorithm must search also for partial disassembly, on the basis of relational constraints. However (ABC) has strong robustness, fast convergence, high flexibility and low number of setting parameters. Premature convergence can be an issue but in industrial applications, such as disassembly sequencing, sub-optimal solution are normally taken into consideration as valid solutions.

This approach, if used in the design phase, would make possible design choices in accordance with the guidelines of Design for Disassembly (DFD) (Mule, 2012) Partial disassembly determines the optimal disassembly level and the optimal sequence to reach this disassembly level (Zhang *et al*., 1997; Tang *et al*., 2002).

In Zussman and Zhou (1999) the optimal disassembly sequence is identified using a Petri-net that incorporated the probabilistic success rate of disassembly operations. In Sarin *et al*. (2006) a traveling salesman problem to achieve optimal or near-optimal solutions is used and in Lambert (2007) an exact method relied on a Bourjault's tree is presented. In Tripathi *et al*. (2009) a self-guided ants method to determine the optimal level of disassembly based on warranty and field service data is presented, in Edmunds *et al*. (2011) a hierarchical GA with an AND/OR graph is used to solve the partial disassembly problem. Other studies report (Meacham *et al*., 1999) a hierarchical product tree representation combined with a linear time algorithm to determine the revenue maximizing disassembly level, a Dynamic Programming (DP) in Teunter (2006).

Corresponding Author: Gianluca Percoco, Department of Mechanical Engineering, Mathematics and Management, Japigia Avenue, 182 (Italian), 70126 Bari, Italy, Tel.: 390805962712; Fax: +390805962788

Disassembly process planning includes various aspects, among which the analysis of the optimal level of disassembly is very important, i.e., the level that makes it advantageous to bear the costs of the process itself. In fact, for each component or sub-assembly, there is a different end-of-life perspective. The disassembly sequence is the main aspect of this planning system, because it has a great impact on efficiency and cost. The difficulty of planning increases with the complexity of the product. In relation to environmental protection, the purpose of disassembly is summarized as follows: obtaining parts, components and subassemblies for reuse in new products; recovery of recyclable materials; removal of hazardous or toxic materials; access to parts or components that may be subject to service operations (repair, maintenance and diagnostics) (Lambert, 1997).

Profitability of disassembly is influenced by the efficiency of the process, product characteristics and the properties of various components. The problem must be addressed considering economic, technological and environmental aspects related to product end of life. Generally, the disassembly planning includes all the problems relating to the dismantling of assemblies (Lambert, 2003): constraints and ways of dismantling, handling of components, collision conditions, generation and optimization of the disassembly sequence and analysis of the optimal disassembly level.

In Karaboga *et al*. (2012), studies about ABC are grouped into three categories: comparisons and modifications, hybrid models and applications; according to this classification the present study deals with a new application.

In Wang and Zhang (2011) disassembly strategy based on bionic disassembly hive model is presented. The disassembly space is modelled as a hive, the judgment of disassembly direction and some other problems are transferred into the evaluation of ant bee groups' migration routes. However this study is dedicated to a clustering more than sequencing problem. In Ilgin and Gupta (2010) a review of research developments in green production and recovery of products, with hundreds of publications divided into four main areas: environmentally conscious product design, reverse and closed loop supply chain, rework and disassembly. In the last area a broad and comprehensive overview of process optimization is presented.

MATERIALS AND METHODS

The approach used in this study for the resolution of the problem of disassembly planning can be divided into the following steps:

- System modelling
- Multi-objective function
- Searching for the best solution

System modelling: The product to be disassembled is considered as a set of independent components. Considering a generic collection of n components, it is possible to number them from 1 to n. The following information can be associated with each item:

- Time of disassembly, required for the removal of the component, which may be resumed by specialized libraries
- Cost of disassembly, for its part, the cost of dismantling operations, to the other, from assessments on the recovery plan at the end of life
- Environmental impact

As regards the feasibility of the sequences, the representation of the proposed system is based on two Boolean matrices used also for assembly sequencing (Dini and Santochi, 1992): the interference and connection matrices.

The first matrix identifies which component can be an obstacle for the removal of other components on a selected direction. For example the case of a product consisting of n components the interference matrix in the X-direction, will be:

$V^X = [v^x_{ij}]$ with j = 1, 2, …, n and i = 1, 2, …, n, with $v^x_{ij} = 1$ if the ith component prevents removal of the j-th component in the X-direction, otherwise $v^x_{ij} = 0$. Similar matrices can be defined for the other directions.

As regards the connection matrix, with m being the number of identified connections in the product, the elements of the matrix will be:

$U = [u_{ij}]$ with j = 1, 2, …, m and i = 1, 2, …, m with $u_{i\,j} = 1$ if the connection i is an obstacle for removing connection j, resulting in a precedence constrain, $u_{i\,j} = 0$ on the contrary.

The multi-objective function: The multi-objective function used in this work has been proposed in Giudice and Fargione (2007); it is expressed through the weighted sum of three factors:

- The disassembly cost, TS, closely dependent on disassembly time for each component
- The cost of the product at end-of-life, C_{eol}
- The environmental impact of the product end-of-life, EI_{eol}

$$\Psi = \varphi_1 \cdot TS + \varphi_2 \cdot (C)_{eol} + \varphi_3 \cdot EI_{eol}$$

φ_1, φ_2, φ_3 are non-homogeneous weights to be normalized. For each component that belongs to the disassembly sequence there is a standardized set of three factors. The best solution is the one with the minimum value of the function ψ.

Considering a product, it is necessary to determine the recovery plan (reuse, recycling or disposal) and

submit it; that plan must balance the cost of disassembly and recovery.
The terms of the ψ function also depend on:

- Material properties and number of components
- The level of disassembly
- The final destination of the disassembled components (reuse, recycle or disposal).

TS, disassembly cost: For the definition of this factor, the execution time of each disassembly task must be known. Considering the ith component, its cost of disassembly is defined as follows:

$$TS_i = C_{smont_i} \sum (ts_i + \delta \cdot cd_i)$$

where the sum is extended to all items that should be removed before removing the ith component, including the ith component:

C_{smont_i} = The cost of labour per hour
ts_i = The disassembly time required for the removal of the ith component
cd_i = A binary value equal to 1 if there is direction change during the disassembly of the product for the ith component, 0 otherwise
δ = A penalty coefficient for direction changes
C_{eol_i} = End-of-life of product

Considering the ith component, its end-of-life cost is defined as follows:

$$C_{eol_i} = \alpha_i \cdot (-C_{prod_i}) + \beta_i \cdot [C_{smal_i} \cdot (1-\xi_i) \cdot W_i + (C_{rcl_i}-R_{rcl_i}) \cdot \xi_i \cdot W_i] + \gamma_i \cdot C_{smal_i} \cdot W_i$$

For each component the coefficients α_i, β_i and γ_i are called recovery planning coefficients and are binary indices taking values 0 or 1, depending on the schedule to which the ith component is intended: recovery or disposal. Their sum is equal to 1 for each component and their value is fixed, depending on strategic recovery choices that may be:

- $\alpha_i = 1$, $\beta_i = 0$, $\gamma_i = 0$ Reuse
- $\alpha_i = 0$, $\beta_i = 1$, $\gamma_i = 0$ Recycling
- $\alpha_i = 0$, $\beta_i = 0$, $\gamma_i = 1$ Disposal

EI_{eol},: Environmental impact of product at end-of-life: This contribution to the objective function quantifies environmental effects associated to the end-of-life phase of each component.

The environmental impact of end-of-life can be calculated as follows:

$$EI_{eol_i} = \alpha_i \cdot (-EI_{prod_i}) + \beta_i \cdot [ei_{smal_i} \cdot (1- \xi_i) \cdot W_i + ei_{rcl_i} \cdot \xi_i \cdot W_i] + \gamma_i \cdot ei_{smal_i} \cdot W_i$$

where,
ξ_i = The recyclable fraction of the ith component
ei_{smali} = The environmental impact of disposal
ei_{rcli} = The environmental impact of recycling of material per unit weight and for the ith component

Solution search: The constraints reduce the complexity of the problem; then it is convenient to use heuristic methods to study complex products, mainly to simplify the problem by introducing additional assumptions and rules, which return sub optimal solutions. However, they can be used in industrial reality (Kishore *et al*., 2005).

The selection among all possible solutions identified as feasible is through the multi-objective function. Each sequence is associated with the value calculated for the cost of disassembly. The algorithm terminates when sequence generation finds the solution with the best value of the objective function.

Discrete artificial bee colony: The algorithm proposed in this study is based on the social behaviour of bees, reproducing the process of seeking nectar,initially proposed for numerical optimizationin (Giudice and Fargione, 2007) and used also for unconstrained and constrained optimization problems (Karaboga, 2005), . Cooperation between insects decreases the cost of finding new sources of food (new viable solutions) and increases the quality of the sources themselves (best solution). A strategy generates various solutions that, through selection criteria, lead to the solution that optimizes the objective function.

The colony of bees chooses the most profitable path, between different sources of nectar. The exchange of information between individuals is the most important event in the formation of collective knowledge. The most important part of the hive, regarding the exchange of information, is the dance floor, where communication between the bees about the quality of food sources, takes place through a procedure called the waggle dance.

A bee colony can extend its search for long distances (more than 10 miles) and in many directions simultaneously to exploit a variety of food sources (Karaboga and Basturk, 2007; Von Frisch, 1976).

Fields of flowers with high amounts of food should be visited by more bees, while the fields with less nectar or pollen should receive fewer bees (Seeley, 1996; Bonabeau *et al*., 1999).

The foraging process begins in a colony with the *Scout Bees* that move randomly from one field of flowers to another (Camazine *et al*., 2003; Wong *et al*., 2009).

When they return to the hive, the scouts who have found a field of high quality flowers, higher than a fixed value, deposit their food and perform the waggle

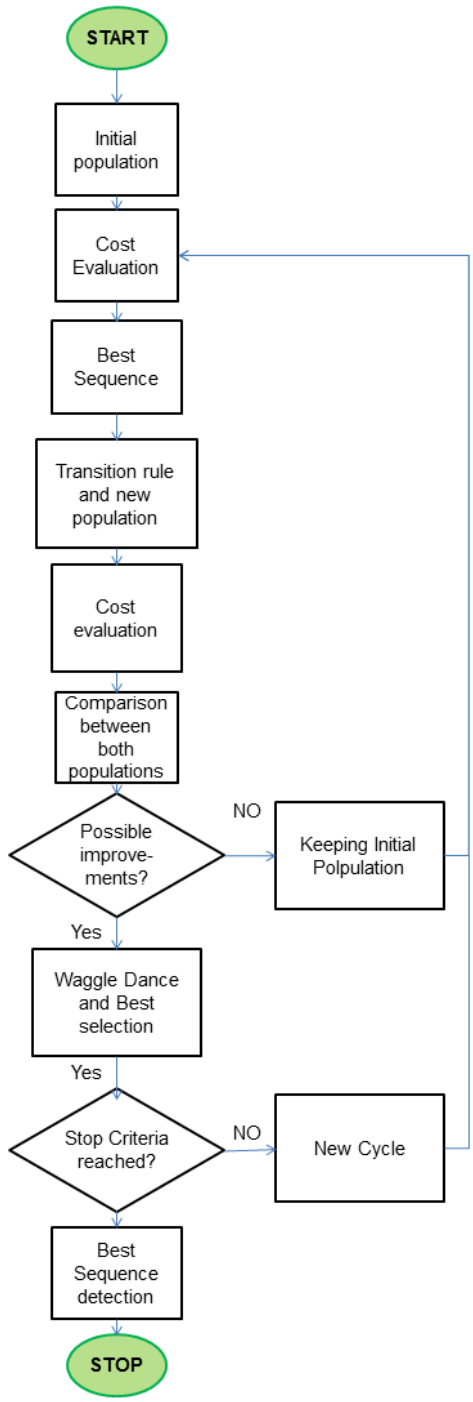

Fig. 1: Discrete ABC algorithm flow chart

dance. This dance enables the colony to evaluate the different fields, both for the quality of the food they provide and the amount of energy required to pick it up. After the dance, the dancer returns to the field of flowers with follower bees that were waiting in the hive to receive information and leave only to the most promising fields. Cooperation between artificial bees produces a rapid discovery of feasible solutions, as well as the discovery of better solutions to the problem.

This study proposes a particular application of ABC: Aimed to solve the problem of finding the optimal or sub-optimal disassembly sequence for an efficient planning process.

As in the case of the travelling salesman problem, the disassembly process planning falls within the category of discrete problems. For this reason, a discrete ABC model has been implemented and applied to appropriate case studies. The algorithm (Fig. 1) can be subdivided into several steps:

Step 1: The first step is the generation of the initial population, the set of bees to start searching, each one associated to a food source, consisting of random solutions matching the geometric and technological constraints of the product.
Step 2: Function evaluation for each sequence
Step 3: Identification of the best sequence
Step 4: Transition rule (Fig. 2), to transform the initial population by generating a new population (new set of bees, each corresponding to a new food source)

The transition is a procedure that allows us to get a new population. Through this rule, "solution-bees" can evolve towards new candidate solutions to solve the problem. The procedure does not apply to solutions that have been abandoned. Given the starting solution that must be transformed, the first component of the new sequence is chosen randomly. Subsequently, the algorithm determines the set of non-assigned components, defined as the set A. When all the components have been assigned, the algorithm proceeds to the examination of the next sequence.

In addition, another set F is defined, containing the next component to be assigned to a given partial sequence, as can be deduced from the best sequence. F will be a set containing a single component, or void if no more elements are available and the complete disassembly has been reached.

Given the sets A and F, if $A \cap F \neq \emptyset$, the component belonging to F will have a greater chance of being selected. The probability of the j-th component, not yet assigned, will be calculated as:

$$P_j = [\lambda]^{\alpha} \cdot [1/C_{ij}]^{\beta} \qquad \text{If } j \in A \cap F$$

$$P_j = [(1-\lambda)/(c-1)]^{\alpha} \cdot [1/C_{ij}]^{\beta} \text{ If } j \in A \wedge j \notin F$$

where c is equal to the number of not yet assigned components, the probability λ has been set to 95% after a trial and error procedure, while C_{ij} is the disassembly time, sequence-dependent, where i is the last component assigned and j the next component to which the probability refers. The weights α and β, defined in the initial part of the algorithm, have been assumed equal to1 and 10 respectively.

On the other hand if $A \cap F = \emptyset$, each component j has the same probability of being chosen:

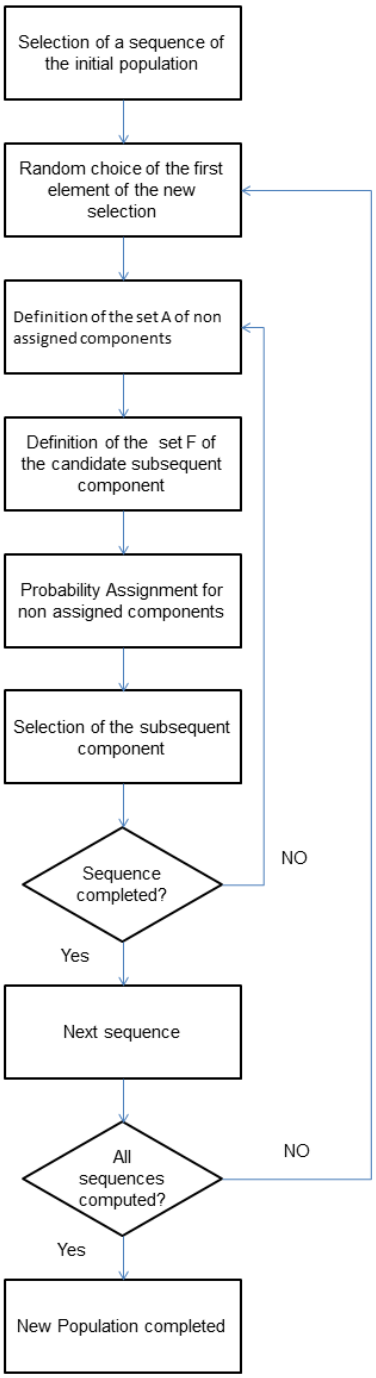

Fig. 2: Transition rule flow chart

$$P_j = [1/\lambda]^{\alpha} \cdot [1/C_{ij}]^{\beta}$$

Subsequently, the probability is normalized by calculating $\overline{P_j} = \frac{P_j}{\sum_{j=1}^{n} P_j}$ with $0 < \overline{P_j} < 1$

Based on these normalized probabilities $\overline{P_j}$, the algorithm randomly chooses the next component in the sequence and the cycle can be repeated again until all the sequences of the new population are completed.

Step 5: The evaluation of the new population. Before evaluating the value of the function associated with each sequence, it must be considered where components were placed, respecting the constraints represented by matrices U and V. There might be only a fraction of the solutions feasible; for a component placed properly one of the possible end of life destinations (reuse, recycling or disposal) must be chosen, while a disposal process can be hypothesized for those not positioned correctly, without any disassembly operation

Step 6: If the rule has led to a sequence with a higher value of the objective function (in the case of disassembly a lower value of TS), the earlier sequence will be replaced by the new one because the new bee will have found a more profitable source of food

Step 7: If in the previous step any modifications in bees have occurred, the next step is the so-called execution of the waggle dance; otherwise, the execution of the algorithm will continue on the previous population, jumping directly to

Step 8: After the execution of the dance, only a part of the bees will continue execution of the algorithm, those with a higher Ψ value than the average of the colony. At the end of this selection process, there will be a new population constituted exclusively of solution-bees able to "survive".

The first phase of the *waggle dance* is the comparison between the old and the new populations. The phase of execution and observation of the dance is the phase where the previous population is replaced by the new, if the new bee is better than preceding one. If the "solution-bee" is improved, the new solution replaces the old and is selected to be a dancer. On the other hand if the "solution-bee" is not improved, the previous solution is preserved.

If at least one solution-dancer is available, the algorithm proceeds with the second phase of the *waggle dance*. The pursuit of the solution indicated by a companion during the dance is the step where the "solution-bee" decides whether or not to abandon the search.

To make that decision, the average profitability of the colony is evaluated. If the cost of a single sequence is lower than or equal to 99% of the average cost of the colony, it will certainly be kept and will continue searching within the algorithm.

As regards all the other solutions that do not satisfy this condition, their fate within the algorithm will

depend in part on their profitability and partly by a random choice.

This last step is regulated by an if-then rule that favours the conservation of sequences with lower cost:

If cost is higher than 1% but lower than 2.5% of the average profitability of the colony, the solution will have the probability of 2% of being abandoned; if cost is higher of 2.5% but lower than 5% of the average cost of the colony, the solution will have 18% probability of being abandoned; if cost is higher than 5% of the average cost of the colony, the solution will have 80% chance of being abandoned. These values have been set using a trial and error procedure.

At the end of this last phase of the *waggle dance*, a new population will be available, ready for the next cycle.

The cycle starts again and repeats the entire execution of the algorithm until the stop condition is verified.

Step 9: When the stop condition is satisfied, the algorithm terminates by identifying the best solution. The solution inside the final population, which presents the best value of the objective function, is taken as the final solution of the algorithm. Otherwise, the cycle repeats with the surviving population, starting from Step 2.

RESULTS

The algorithm has been tested on several case studies; among them two cases has been selected and reported below. These cases were chosen to show how the algorithm behaves on products that reach the minimum of the function when completely disassembled or when the minimum of the function is reached when disassembly is not complete. In this preliminary study the products have been simplified grouping together several components into sub-assemblies.

The first case study: The first case study is shown in Fig. 3. The entire product has been split into ten subassemblies that will be treated as single components.

Matrix modelling: The first step consists in achieving the interference and connection matrices. As shown in Fig. 4, there are several relationships between components. The interference matrix should be compiled in each direction for disassembly. Reducing the problem to only two dimensions, there are two matrices to be filled: one in the vertical direction and the other in the horizontal. It is possible to superimpose the two matrices, thus obtaining a single Boolean matrix, where the presence of the element ' 1 ' indicates that there is a relationship between corresponding elements, in either the horizontal or vertical direction. On the other hand the element ' 0 ' expresses a lack of interference on both directions.

The connection matrix is obtained putting in evidence: the presence of unitary elements in the interference matrix (Fig. 5) shows the existence of a connection between the elements.

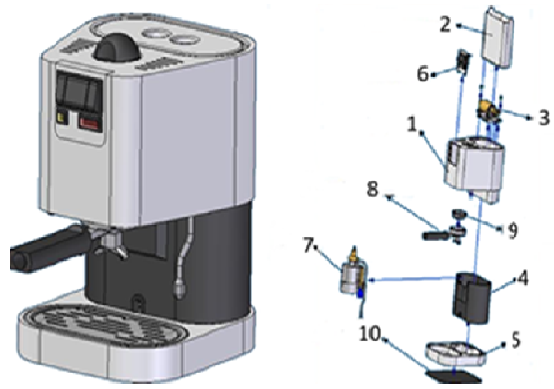

Fig. 3: First case study

COMP	1	2	3	4	5	6	7	8	9	10
1	0	1	0	1	0	1	0	0	0	0
2	1	0	0	0	0	0	0	0	1	0
3	0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	1	0	0	0
5	0	0	1	1	0	0	0	0	0	1
6	1	0	0	0	0	0	0	0	0	0
7	0	0	0	1	0	0	0	0	1	0
8	0	0	0	0	0	0	0	0	1	0
9	0	0	0	0	0	0	1	1	0	0
10	0	0	0	0	1	0	0	0	0	0

Fig. 4: Interference matrix for first case study

CONN	1-2	1-4	1-6	3-5	4-5	4-7	5-10	7-9	8-9
1-2	0	0	0	0	0	0	0	0	0
1-4	0	0	0	0	0	1	0	0	0
1-6	0	0	0	0	0	0	0	0	0
3-5	0	0	0	0	0	0	0	0	0
4-5	0	0	0	0	0	0	0	0	0
4-7	0	0	0	0	0	0	0	0	0
5-10	0	0	0	1	1	0	0	0	0
7-9	1	0	0	0	0	1	0	0	0
8-9	0	0	0	0	0	0	0	0	0

Fig. 5: Connection matrix for first case study

Next Component

Last Component Assigned	COMP	1	2	3	4	5	6	7	8	9	10
	1	0	57.00	62.00	63.45	39.30	27.55	71.30	34.45	68.70	40.00
	2	90.00	0	67.00	68.45	34.30	32.55	76.30	29.45	73.70	35.00
	3	85.00	57.00	0	63.45	39.30	27.55	71.30	34.45	68.70	40.00
	4	85.00	57.00	62.00	0	39.30	27.55	71.30	34.45	68.70	40.00
	5	90.00	52.00	67.00	68.45	0	32.55	76.30	29.45	73.70	35.00
	6	85.00	57.00	62.00	63.45	39.30	0	71.30	34.45	68.70	40.00
	7	85.00	57.00	62.00	63.45	39.30	27.55	0	34.45	68.70	40.00
	8	90.00	52.00	67.00	68.45	34.30	32.55	76.30	0	73.70	35.00
	9	85.00	57.00	62.00	63.45	39.30	27.55	71.30	34.45	0	40.00
	10	90.00	52.00	67.00	68.45	34.30	32.55	76.30	29.45	73.70	0

Fig. 6: Timing of sequence-dependent disassembly: Second case study

Table 1: Disassembly directions and time of each element: First case study

Component	Disassembly time (s)	Orientation	Material
1	85.00	Vertical	Steel
2	52.00	Horizontal	Steel
3	62.00	Vertical	Steel
4	63.45	Vertical	Steel
5	33.30	Horizontal	Steel
6	27.55	Vertical	Steel
7	71.30	Vertical	Steel
8	29.45	Horizontal	Steel
9	68.70	Vertical	Steel
10	35.00	Horizontal	Steel

Table 2: Costs of disposal, recycling and profit derived from the relative weight of each item: First case study

Component	Disposal cost (/kg)	Recyclineg profit (/t)	Wight (kg)
1	0.127	0.0842	1.05
2	0.127	0.0842	0.12
3	0.127	0.0842	0.7
4	0.127	0.0842	0.936
5	0.127	0.0842	0.76
6	0.127	0.1855	0.34
7	0.127	0.0842	1.8
8	0.127	0.1855	0.6
9	0.127	0.0842	0.24
10	0.127	0.0842	0.3

As regards the time of each disassembly operation, each time has been assumed, based on an approach similar to the MTM (Methods and Time Measurement) method. It has been assumed for each direction change there is an average increase of disassembly time equal to 5 sec. These hypotheses and assumptions lead to the data necessary for the assessment of disassembling operation time for each part as summarized in Table 1:

Of course, disassembly times are sequence-dependent and a matrix must be built in order to indicate the disassembly time of each component when starting from each configuration. These times are shown in Fig. 6, where the influence of direction change on single disassembly times is evident.

Objective function: The values used for C_{eol}, hypotisedafter consulting a specialized industrial partner, have been summarized in Table 2 and sorted for each component: cost of disposal, profit resulting from the sale of components for recovery, the weight of the component. To reduce the complexity of the task each item has been considered as if it consisted of a single material. The entire weight of the element is intended for recycling or disposal. Reuse costs have been assumed equal to infinity, because of the small size of the components and a sure non-convenience in reusing.

BEST SOLUTIONS										
1	2	8	10	5	6	9	7	3	4	1
2	2	8	10	5	6	3	4	9	1	7
3	2	8	10	5	6	7	3	4	1	9
4	10	5	2	8	6	9	1	7	3	4
5	2	8	10	5	6	3	4	1	9	7
6	10	5	2	8	6	3	4	1	9	7
7	2	8	10	5	6	4	9	1	3	7
8	2	8	10	5	6	9	7	4	3	1
9	10	5	2	8	6	7	3	4	1	9

Fig. 7: Best solutions in 10 runs: First case study

Table 3: Time, disassembly direction, and material of components: Second case study

Component	Disassembly time (s)	Orientation	Material
1	3.60	Horizontal	Steel
2	12.70	Horizontal	Plastic
3	12.40	Horizontal	Steel
4	116.70	Horizontal	Steel
5	261.30	Horizontal	Steel
6	120.30	Horizontal	Plastic
7	76.40	Horizontal	Plastic
8	81.50	Vertical	Steel
9	15.60	Horizontal	Steel
10	47.10	Horizontal	Plastic

Finding the best sequence: A random initial population was used to start the algorithm, created in the Matlab ® environment, starting with a population of 20 random initial solutions, iterating for 20 cycles. The minimum value of the objective function found by the algorithm is 0.2357 €/piece. Figure 7 shows the best solutions found in 10 runs.

The same minimum value was found for different solutions in 100% of cases (10 launches of the algorithm). The minimum was found in the first 5 cycles in 80% of cases.

The behaviour of solution-bees of several program launches is reported in Fig. 8, considering one single example run.

Starting from random solutions, in this sample run only a few "solution-bees ' (6 of 20) reach the minimum value and terminate 20 cycles. The time necessary for each run is less than 5 sec on a 3.8 Ghz Pentium 4 CPU.

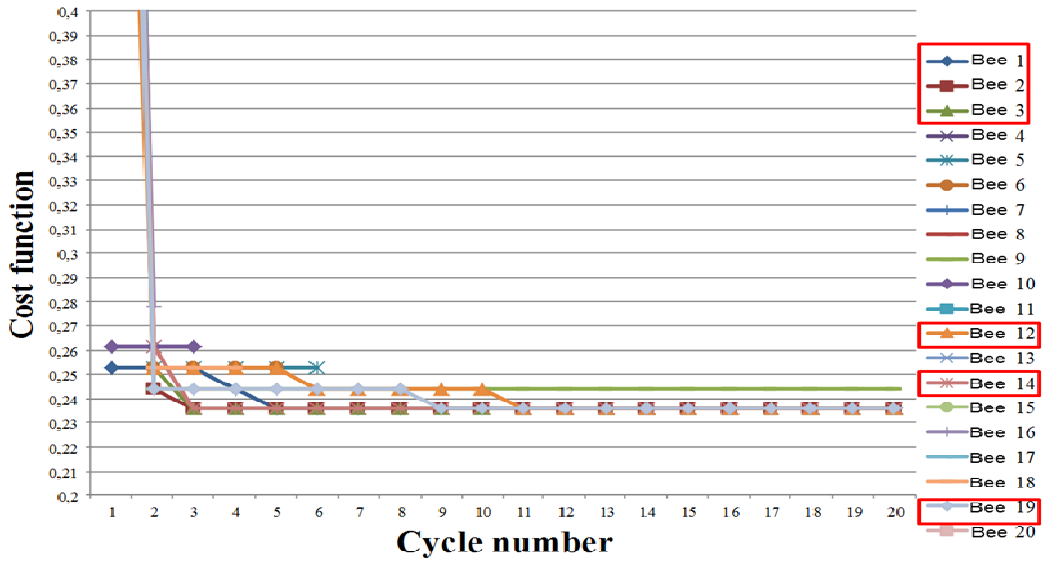

Fig. 8: Evolution example of Ψ for each "solution-bee" in one launch: first case study

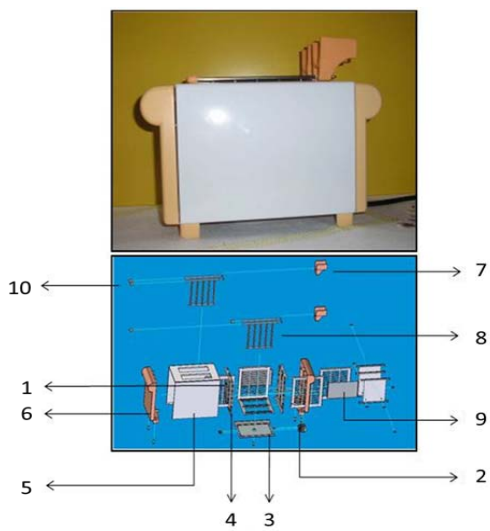

Fig. 9: Second case study

COMP	1	2	3	4	5	6	7	8	9	10
1	0	1	1	0	1	0	0	0	0	0
2	1	0	0	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0	0	0	0
4	0	0	0	0	1	0	0	0	0	0
5	1	1	0	1	0	1	0	1	1	0
6	0	0	0	0	1	0	1	0	0	0
7	0	0	0	0	0	1	0	0	0	1
8	0	0	0	0	1	0	0	0	0	0
9	0	0	0	0	1	0	0	0	0	0
10	0	0	0	0	0	0	1	0	0	0

Fig. 10: Interference matrix for the second case study

COMP	1-2	1-3	1-5	4-5	5-6	5-8	5-9	7-10	6-7
1-2	0	0	0	0	0	0	0	0	0
1-3	0	0	0	0	0	0	0	0	0
1-5	0	0	0	0	0	0	0	0	0
4-5	0	0	0	0	1	0	0	0	0
5-6	0	0	0	0	0	0	0	0	0
5-8	0	0	0	0	0	0	0	0	0
5-9	0	0	0	0	0	0	0	0	0
7-10	0	0	0	0	0	0	0	0	1
6-7	0	0	0	0	0	0	0	1	0

Fig. 11: Connection matrix for the second case study

Next Component

Last Component Assigned / COMP	1	2	3	4	5	6	7	8	9	10
1	0	12.7	12.4	116.7	261.3	120.1	76.4	86.5	15.6	47.1
2	3.6	0	12.4	116.7	261.3	120.1	76.4	86.5	15.6	47.1
3	3.6	12.7	0	116.7	261.3	120.1	76.4	86.5	15.6	47.1
4	3.6	12.7	12.4	0	261.3	120.1	76.4	86.5	15.6	47.1
5	3.6	12.7	12.4	116.7	0	120.1	76.4	86.5	15.6	47.1
6	3.6	12.7	12.4	116.7	261.3	0	76.4	86.5	15.6	47.1
7	3.6	12.7	12.4	116.7	261.3	120.1	0	86.5	15.6	47.1
8	8.6	17.7	17.4	121.7	266.3	125.1	81.4	0	20.6	52.1
9	3.6	12.7	12.4	116.7	261.3	120.1	76.4	86.5	0	47.1
10	3.6	12.7	12.4	116.7	261.3	120.1	76.4	86.5	15.6	0

Fig. 12: Timing of sequence-dependent disassembly: Second case study

Second case study: The second case study is composed by more than one hundred components that have been grouped into 10 subassemblies considered as single components (Fig. 9).

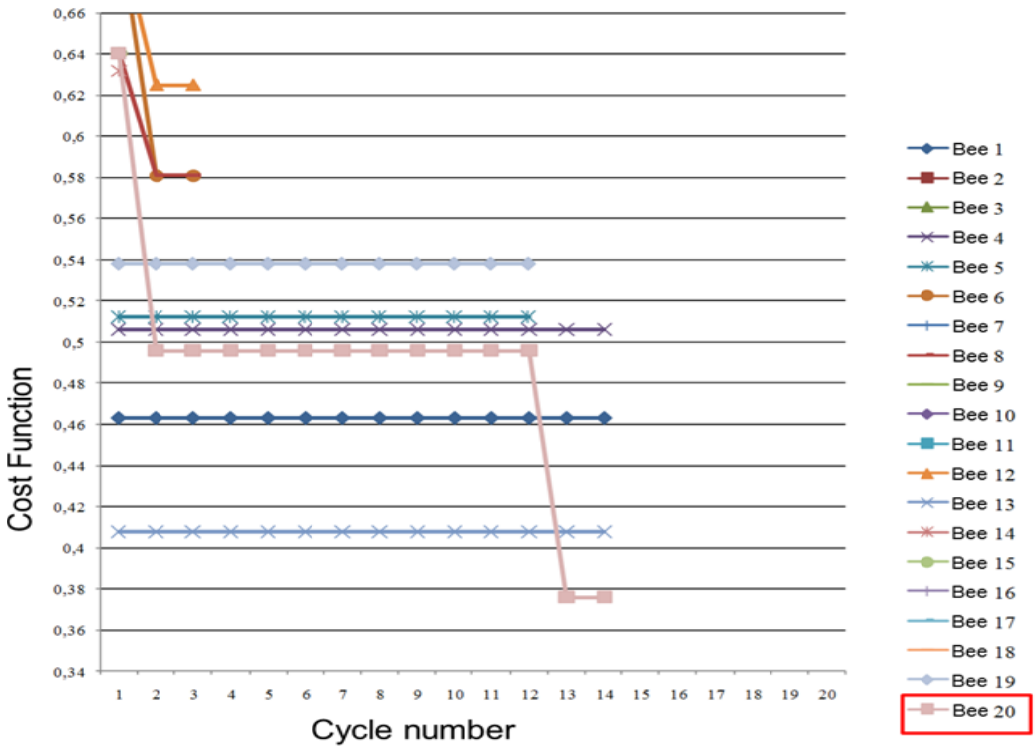

Fig. 13: Evolution example of the multi-objective function for each "solution-bee" in one launch: Second case study

Table 4: Costs of disposal, recycling and profit derived from the relative weight of each item: Second case study

Component	Disposalcost (/kg)	Recyclineg profit (/t)	Wight (kg)
1	0.127	0.0842	0.6
2	0.127	0.1855	0.1
3	0.127	0.0842	0.1
4	0.127	0.0842	0.7
5	0.127	0.0842	1.3
6	0.127	0.1855	0.4
7	0.127	0.1855	0.8
8	0.127	0.0842	0.9
9	0.127	0.0842	0.12
10	0.127	0.1855	0.5

Matrix modelling: Interference and connection matrices are shown in Fig. 10 and 11 for the second case study.

The objective function: time and cost recovery: For this case study two disassembly directions are considered, vertical and horizontal. Estimated data for each element are shown in Table 3.

Sequence-dependent disassembly times are shown in Fig. 12. In this specific case it is supposed that only component 8 needs one direction change and then only component 8 affects sequence-dependent disassembly costs. The choice of a unique direction change has been made in order to have high difficulty in finding global vs local minima to test the algorithm correctly.

Since the materials of the components are just plastic or steel, the data used for C_{eol} are shown in Table 4, following the same assumption as made for the first case study.

Finding the best sequence: Also in this case, results are expressed for algorithm runs related to 20 random initial solutions and 20 cycles. The best sequences have been identified as [9 10 7 3], [3 1 2 7 5 4] and [7 8 5 10], not including a complete disassembly. Further considerations can indicate [3 1 2 7 5 4] as the best solution from an environmental point of view, being the deepest disassembly level obtained. The trend of solution-bees, considering one example run is shown in Fig. 13.

All solution-bees, except 1 and 13, start with high multi-objective function values. This leads to the result that only one bee, 20, manages to reach the optimum value of 0.37562 €/pz. In this case also, the time necessary for each run is between 4 and 5 seconds on a 3.8 Ghz Pentium 4 CPU.

CONCLUSION

The stability of the ABC approach for product disassembly planning must be further enhanced by studying a wider variety of product types and configurations. In both the case studies optimal or near-optimal solution is obtained: in the third generation or 13th generation. Whenever ABC algorithm reaches solution in less than 100 iterations, the algorithm does not have time to properly start searching. However it can be concluded that its speed must be improved to be exploitable and competitive with other metaheuristics

such as Genetic Algorithms or Simulated Annealing. Further research should be addressed to speed up this nature inspired technique to the Disassembly Sequencing Problem.

ACKNOWLEDGEMENT

The authors wish to thank "Globecosrl" (www.globeco.info), specialized in waste management, for their suggestions regarding data shown in Table 2 and 4.

REFERENCES

Akay, B. and D. Karaboga, 2010. Artificial bee colony algorithm for large-scale problems and engineering design optimization. J. Intell. Manuf., 23(4): 1001-1014.

Bonabeau, E., M. Dorigo and G. Theraulaz, 1999. Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, New York.

Brajevic, I. and M. Tuba, 2012. An upgraded Artificial Bee Colony (ABC) algorithm for constrained optimization problems. J. Intell. Manuf., DOI: 10.1007/s10845-011-0621-6.

Camazine, S., J. Deneubourg, N.R. Franks, J.G. Sneyd, Theraula and E. Bonabeau, 2003. Self-Organization in Biological Systems. Princeton University Press, Princeton.

Dini, G. and M. Santochi, 1992. Automated sequencing and subassembly detection in assembly planning. CIRP Ann-Manuf. Technol., 41(1): 1-4.

Edmunds, R., M. Kobayashi and M. Higashi, 2011. Generating optimal disassembly process plans from AND/OR relationships using a hierarchical genetic algorithm interdisciplinary design. Proceeding of the 21st CIRP Design Conference, pp: 16.

Giudice, F. and G. Fargione, 2007. Disassembly planning of mechanical systems for service and recovery: A genetic algorithms based approach. J. Intell. Manuf., 18(3): 313-329.

Ilgin, M.A. and S.M. Gupta, 2010. Environmentally conscious manufacturing and product recovery (ECMPRO): A review of the state of the art. J. Env. Manage., 91: 563-591.

Karaboga, D., 2005. An idea based on honey bee swarm for numerical optimization. Technical Report-TR06. Computer Engineering Department, Engineering Faculty, Erciyes University.

Karaboga, D. and B. Basturk, 2007. A powerful and efficient algorithm for numerical function optimization: Artificial Bee Colony (ABC) algorithm. J. Global Optim., 39(3): 459-471.

Karaboga, D., B. Gorkemli, C. Ozturk and N. Karaboga, 2012. A comprehensive survey: Artificial Bee Colony (ABC) algorithm and applications. Artif. Intell. Rev., Doi: 10.1007/s10462-012-9328-0.

Kishore, K.P., A.J.D. Lambert and S.M. Gupta, 2005. Disassembly Modeling for Assembly, Maintenance, Reuse and Recycling. CRC Press, Boca Raton, Florida, USA.

Lambert, A., 1997. Optimal disassembly of complex products. Int. J. Prod. Res., 35(9): 2509-2523.

Lambert, A., 2003. Disassembly sequencing: A survey. Int. J. Prod. Res., 41(16): 3721-3759.

Lambert, A.J.D., 2007. Optimizing disassembly processes subjected to sequence-dependent cost. Comput. Oper. Res., 34(2): 536-55.

Lee, S.C. and L.H. Shih, 2012. A novel heuristic approach to determine compromise management for end-of-life electronic products. J. Oper. Res. Soc., 63(5): 606-619.

Meacham, A., R. Uzsoy and U. Venkatadri, 1999. Optimal disassembly configurations for single and multiple products. J. Manuf. Syst., 18(5): 311-322.

Mule, J.Y., 2012. Design for disassembly approaches on product development. Int. J. Sci. Eng. Res., 3(6).

Sarin, S., H. Sherali and A. Bhootra, 2006. A precedence-constrained asymmetric traveling salesman model for disassembly optimization. IIE Trans., 38(3): 223-237.

Seeley, T.D., 1996. The Wisdom of the Hive: The Social Physiology of Honey Bee Colony. Massachusetts, Harvard University Press, Cambridge.

Tang, Y., M. Zhou, E. Zussman and R. Caudill, 2002. Disassembly modeling, planning and application. J. Manuf. Syst., 21(3): 200-217.

Teunter, R.H., 2006. Determining optimal disassembly and recovery strategies. Omega, 34(6): 533-537.

Tripathi, M., S. Agrawal, M.K. Pandey, R. Shankar and M.K. Tiwari, 2009. Real world disassembly modeling and sequencing problem: Optimization by Algorithm of Self-Guided Ants (ASGA). Robot. Comput-Integ. Manuf., 25(3): 483-496.

Von Frisch, K., 1976. Bees: Their Vision, Chemical Senses and Language. Cornell University Press, Ithaca.

Wang, W. and J. Zhang, 2011. Research on bionic hive modularization technology of disassembly. Appl. Mech. Mater., 44-47: 106-111.

Wong, L.P., M.Y.H. Low and C.S. Chong, 2009. A bee colony optimization algorithm with the fragmentation state transition rule for traveling salesman problem. Retrieved from: http://kecubung.webfactional. com/a-bee-colony-optimization-algorithm-with-the-fragmentation-state.pdf.

Zhang, H.C., T.C. Kuo, H. Lu and S.H. Huang, 1997. Environmentally conscious design and manufacturing: A state-of-the-art survey. J. Manuf. Syst., 16(5): 352-371.

Zussman, E. and M. Zhou, 1999. A methodology for modeling and adaptive planning of disassembly processes. IEEE T. Robot. Autom., 15(1): 190-194.