Journal of Industrial Engineering and Management
JIEM, 2011 – 4(3):418-435 – Online ISSN: 2013-0953 – Print ISSN: 2013-8423
http://dx.doi.org/10.3926/jiem.2011.v4n3.p418-435

A methodological approach for designing and sequencing product families in Reconfigurable Disassembly Systems

Ignacio Eguia, Sebastian Lozano, Jesus Racero, Fernando Guerrero
University of Seville (SPAIN)
ies@us.es; slozano@us.es; jrm@us.es; fergue@us.es

Received June 2010
Accepted September 2011

Abstract:

Purpose: A Reconfigurable Disassembly System (RDS) represents a new paradigm of automated disassembly system that uses reconfigurable manufacturing technology for fast adaptation to changes in the quantity and mix of products to disassemble. This paper deals with a methodology for designing and sequencing product families in RDS.

Design/methodology/approach: The methodology is developed in a two-phase approach, where products are first grouped into families and then families are sequenced through the RDS, computing the required machines and modules configuration for each family. Products are grouped into families based on their common features using a Hierarchical Clustering Algorithm. The optimal sequence of the product families is calculated using a Mixed-Integer Linear Programming model minimizing reconfigurability and operational costs.

Findings: This paper is focused to enable reconfigurable manufacturing technologies to attain some degree of adaptability during disassembly automation design using modular machine tools.

Research limitations/implications: The MILP model proposed for the second phase is similar to the well-known Travelling Salesman Problem (TSP) and therefore its complexity grows exponentially with the number of products to

disassemble. In real-world problems, which a higher number of products, it may be advisable to solve the model approximately with heuristics.

Practical implications: The importance of industrial recycling and remanufacturing is growing due to increasing environmental and economic pressures. Disassembly is an important part of remanufacturing systems for reuse and recycling purposes. Automatic disassembly techniques have a growing number of applications in the area of electronics, aerospace, construction and industrial equipment. In this paper, a design and scheduling approach is proposed to apply in this area.

Originality/value: This paper presents a new concept called Reconfigurable Disassembly System, which represents disassembly systems with reusability, scalability, agility and reconfigurability features. These features and some specific costs are considered as part of the proposed methodology.

Keywords: linear programming, sequencing, disassembly, reconfigurable manufacturing

1 Introduction

In the last decade, the areas of Environmentally Conscious Manufacturing and Product Recovery (ECMPRO) have gained increasing attention due to growing awareness of conserving energy, material resources and landfill capacity. In many countries waste dump capacity is already limited by legislation, and the estimated amount of obsolete products returning to the manufacturers are increasing exponentially. This environmental awareness and the recycling regulations have put pressure on many manufacturers and consumers to produce and dispose of products in an environmentally responsible manner.

Environmentally Conscious Manufacturing deals with developing methods for manufacturing products that satisfy environmental standards and requirements. The End of Life (EOL) disposal plays a key role in these green principles, and significant among of research has focused on Product Recovery. Recent literature on ECMPRO is organized into four main areas (Ilgin & Gupta, 2010):

- Environmentally conscious product design: methodologies considering certain environmental criteria in the design process (viz., design for

environment, design for disassembly, design for recycling, life cycle analysis and material selection)

- Reverse and close-loop supply chains: methodologies considering all the activities associated with the collection and either recovery or disposal of used products and the simultaneous consideration of forward and reverse flows (viz., network design, network and product design simultaneously, optimization of transportation of goods, selection of used products,…)

- Remanufacturing: methodologies involving the conversion of worn-out products into like-new conditions (viz., forecasting, production planning, production scheduling, inventory models,…)

- Disassembly: methodologies considering the activities of separation of an assembly into its components, subassemblies or other groupings

Disassembly is an important process in material and product recovery since it allows recovering materials with a high degree of purity, reusing components or removing toxic materials. Two important phases may be considered in a disassembly process (Güngör & Gupta, 1997):

- Disassembly planning or sequencing: deals with the problem of determining the best order in which to dismount joints and remove parts

- Disassembly scheduling: consists in determining the time and the amounts of EOL products to process, to fulfil the demand for the parts or components over a planning horizon, minimizing costs

Previously, another two phases must be considered (Tang, Zhou, Zussman & Caudill, 2002):

- Modelling and representation of product disassembly sequences is an important decision for disassembly planning

- Disassembly system design and line balancing: concerns with the determination of the number of resources (stations, machine tools, material handling…) needed to satisfy the disassemble constraints and the demand of products, and also to assign the disassembly tasks to these resources

This paper deals with the disassembly system design and the disassembly scheduling of a new paradigm called Reconfigurable Disassembly System (RDS). A RDS can be defined as an automated disassembly system that uses reconfigurable

manufacturing technologies for fast adaptation to changes in the quantity and mix of products to disassemble.

The majority of the existing disassembly plants are based on the manual labor. The number of obsolete products returning to the manufacturers increases exponentially. The products typically processed on disassembly systems are large and medium-sized electrical products such as TV sets, computers, home electronics or copy machines. As disassembling of electronic products is very expensive, great effort has been made by firms to automate their disassembly processes. To cope with such demand of automation, researches have studied different aspects of disassembly automation in recent years (Seliger, Basdere, Keil & Rebafka, 2002; Duta & Filip, 2008). However expectation on automated disassembly has not yet fulfilled owing to the severe constraints on disassembly technologies, and systems deals with missing product information, varying products, different product conditions and unpredictable product states (Wiendahl, Scholz-Reiter, Bürkner & Scharke, 2001).

The focus of this work is to enable reconfigurable manufacturing technologies to attain some degree of adaptability during disassembly automation design using modular machine tools. By definition, a Reconfigurable Manufacturing System is designed at the outset for rapid change in hardware and software components, in order to quickly adjust production capacity and functionality within a part family in response to sudden changes in demand (Koren et al., 1999). Then reconfigurable manufacturing appears as a response to the need to adapt manufacturing systems to external variations (like product demand) as well as internal variations of the own manufacturing system, in a quick and cost-effective way. Among the technologies that have been developed to obtain a high degree of reconfigurability in manufacturing systems are software technologies that use open architecture controllers that allow the reconfiguration of the devices, and machine tools hardware technologies, with modular blocks that allow the adaptation in capacity and functionality to current needs (Xiaobo, Jiancai & Zhenbi, 2000).

We present a new concept called Reconfigurable Disassembly System (RDS), which represents disassembly systems with reusability, scalability, agility and reconfigurability features. Similar to Reconfigurable Assembly Systems (Yu, Yin, Sheng & Chen, 2003), a RDS can be obtained from an automated disassembly system through the addition of reconfigurable software and hardware systems. The different subsystems that form a RDS are: a) one or more workstations with Reconfigurable Machine Tools (RMT) on which the disassembly tasks are actually

carried out, b) devices for loading the products and unloading the components, and c) a material handling system. RMT are machine tools with a modular structure (basic modules and auxiliary modules) that can be changed to obtain increased functionality or capacity for disassembly a family of similar products. Each RMT has its own basic modules so that the operating capacity is mostly provided by the auxiliary modules attached. That is why in the proposed approach only the auxiliary modules are taken into account. The physical layout of a RDS can be similar to that of a RAS, i.e. in series, carrousel or single station (Jain, Fukuda, Komma & Reddy, 2006).

This paper deals with the design and sequencing of generic RDSs. To achieve this, a methodology has been developed to design the RDS (i.e. to group an amount of products during a time horizon into families and to configure the resources needed to disassembly each product family), and to find the best sequence for all product families (i.e. to find a sequence of the product families trying to minimize reconfigurability and operational costs).

This paper is structured as follows. In Section 2 the methodology for grouping products into families in a RDS (subsection 2.1) and for sequencing the product families (subsection 2.2) is presented. In Section 3 a numerical illustration is presented to validate the methodological approach for the design phase (subsection 3.1) and the sequencing phase (subsection 3.2). Also a sensitive analysis with respect to cost coefficients is developed (subsection 3.3). Finally, in Section 4 main conclusions and further research are pointed out.

2 Methodology for designing a RDS

Different types and quantities of products must be disassembled within a certain time horizon in a Reconfigurable Disassembly System, using Reconfigurable Machine Tools and their available Modules. A Reconfigurable System is configured with the necessary RMT and modules to manufacture (or disassemble in our case) a family of similar products at the same time. Once a family is manufactured (disassembled), the system is reconfigured for manufacturing (disassembling) the following family effectively. In each change, the system incurs in a reconfigurable cost, which depends on the current configuration and the destination configuration.

These features have been considered in the two-phase methodology developed for designing RDSs. The first phase (*Product Families Formation*) in the design of an RDS is the grouping of products to disassemble in families with similar features so that each family will require a different configuration of the RDS. Disassembling

similar products allow a reduction of Setups, Lead-times, Work-In-Process and Material Handling, thus increasing productivity. Then, the most cost-effective sequencing of the families is computed to find the most appropriate configuration.

For the development of an algorithm to group products into families in a RDS, the following data are necessary: a) types and quantities of the products to disassemble within a certain time horizon, b) existing RMT and available modules library, c) operations and processing times required to disassemble each product type, and d) machines and modules required for each disassembly task.

The second phase (*Product Families Selection and Scheduling*) consists on the selection and the disassembly sequence of the product families with the minimum operational costs, that is, reconfiguration costs and the costs of under-utilization of resources. A mathematical model has been developed using information from the first phase plus that related to operation costs.

Figure 1 shows a schematic representation of the proposed two-phase approach, with 9 types of products to disassemble within a certain time horizon (A, B,…, I). As result of the first phase (*Product Families Formation*) 4 families are formed ({A,C,D}; {B,E}; {F,H}; {G,I}). And the minimum cost sequence of families obtained from the second phase (*Product Families Selection and Scheduling*) is {A,C,D}→{F,H}→{B,E}→{G,I}. For each family of products to disassemble the required machines (RMT) and modules configuration and the corresponding total costs are calculated.

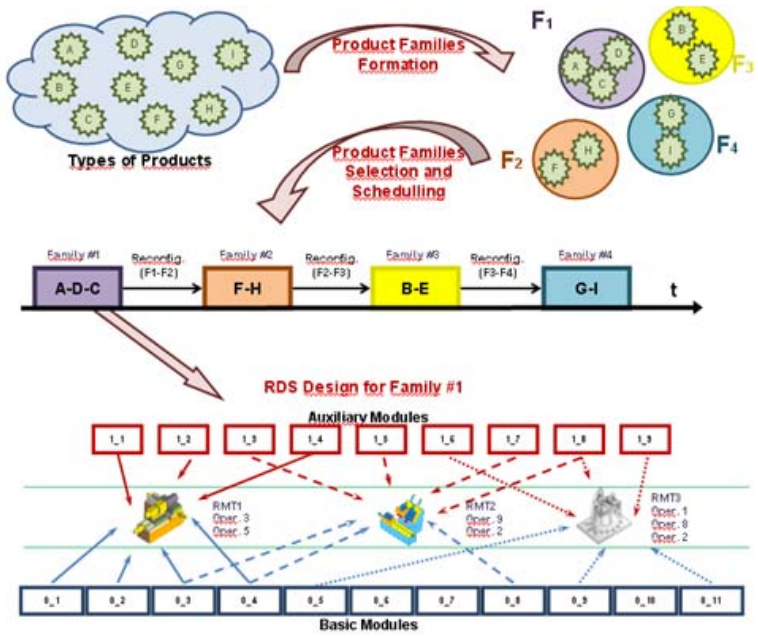

Figure 1. Schematic representation of the proposed approach

Further research will deal with the problem of sequencing the disassembly tasks for each product as well as scheduling the products within each family.

2.1 Product families formation

The proposed methodology is based on hierarchical clustering techniques, which have been commonly used in cellular manufacturing machine grouping and part family formation (Vakharia & Wemmerlov, 1995). These techniques carry out a progressive grouping of products using similarity coefficients that measure the degree of affinity of each pair of products. Two products are considered more similar the longer the time that they share the same system resources.

A specific feature that will be taken into account is the limitation (for technical reasons) on the maximum number of auxiliary modules that a RMT can support. Therefore, products can be grouped within a family provided that the total number of required modules on the corresponding required machines does not exceed the maximum number of modules allowed.

Let

- i: Index of the N types of products to disassemble
- j: Index of the P(i) disassembly operations/tasks for disassembling product type i
- m: Index of the M available machine tools (RMT)
- k: Index of the K available auxiliary modules
- U_m: Maximum number of auxiliary modules supportable by machine m
- D_i: Demand of product i along time horizon considered
- o: Index of the O operation types
- o(j,i): Operation type of disassembly task j of product i
- t_{ij}: Duration of disassembly task j of product i
- δ_{omk}=1, if operation type o requires module k in machine m; =0, otherwise

We assume that each operation type can only be performed on one of the machines. We also assume that for each product the number of modules required

by all its disassembly tasks in a specific machine do not exceed the maximum number of modules that the machine can support.

In order to account for the possibility that the products to disassemble arrive in different conditions so that not all their disassembly tasks may need to be performed we consider as known the following data

- f_{ij}: Relative frequency (i.e. probability) of disassembly task j of product i being required

From the given data the processing time (PT_{imk}) of each product i on each resource (i.e. on each combination of machine m and module k) can be computed as

$$PT_{imk} = D_i \cdot \sum_{j=1}^{P(i)} t_{ij} \cdot f_{ij} \cdot \delta_{o(j,i)mk}$$

Equation 1. Processing time of each product on each resource

The similarity of two products can be measured taking into account the fraction of their overall processing times that are common, i.e. that are performed on the same machine using the same modules. Thus, the proposed similarity coefficient $S_{ii'}$ between two products i and i' is

$$S_{ii'} = \begin{cases} 0 \quad if \quad \sum_{m=1}^{M}\sum_{k=1}^{K} PT_{imk} \cdot PT_{i'mk} = 0 \\ \dfrac{\sum_{m=1}^{M}\sum_{k=1}^{K} Min(PT_{imk}, PT_{i'mk})}{\sum_{m=1}^{M}\sum_{k=1}^{K} Max(PT_{imk}, PT_{i'mk})} \quad if \quad \sum_{m=1}^{M}\sum_{k=1}^{K} PT_{imk} \cdot PT_{i'mk} > 0 \end{cases}$$

Equation 2. Similarity coefficient between two products

Thus, the similarity between two products is zero if they have no common resource (i.e. same machine m and module k) and would be unity if they required exactly the same resources for exactly the same duration. Moreover, it can be shown that this similarity coefficient coincides with the Jaccard similarity coefficient commonly used in cell formation if all the operation processing times and product demands are the unity.

In order to prevent the grouping of products requiring more modules on a machine that it can support, the similarity coefficient $S_{ii'}$ between two products is assigned

a value of -1 if for a certain machine m the number of different modules required by both products exceed U_m.

To form the product families a hierarchical clustering algorithm, namely ALC (Average Linkage Clustering), is proposed (Vakharia & Wemmerlov, 1995). The algorithm first groups the two products with the highest similarity coefficient. Next, in each step, either two non-grouped products are clustered or a non-grouped product is added to an existing group or two groups are merged. In the second case the similarity between a product and product group is computed as the average of the similarity coefficients between the non-grouped product and each product in the product group. In the third case, the similarity between two product groups is analogously computed as the average of the similarity coefficients between the products in each group, i.e.

$$S_{ss'} = \frac{\sum_{i \in s} \sum_{i' \in s'} S_{ii'}}{N_s \cdot N_{s'}}$$

Equation 3. Similarity coefficient between two product groups

where s, s' are the two product groups, $S_{ss'}$ the corresponding similarity coefficient, and N_s, $N_{s'}$ the number of products in groups s and s' respectively.

Every time that adding a product to a group or merging two groups the constraint on the maximum number of modules that each machine can support is checked so that in case the constraint is violated the corresponding grouping is considered infeasible and the its similarity coefficient assigned a value -1.

This hierarchical agglomerative process of selecting the maximum similarity coefficient, carrying out the corresponding grouping step and updating the remaining similarity coefficients matrix is repeated until all the products are grouped in a single family or no additional grouping of products is feasible. The results is a dendogram, which is a an inverted tree that shows the successive grouping steps along with the similarity coefficient levels at which each grouping step was carried out. From this dendogram and selecting a certain level of the tree, the number and composition of the families can be obtained (Galan, Racero, Eguia & Garcia, 2007). Note, however, that in the proposed approach the number of families is not given; the optimal number of families is computed taking into account the different costs incurred. Thus, in the next section a mathematical

programming model is presented for the selection of the number of families and their sequencing that leads to minimum costs.

2.2 Product families selection and sequencing

To select the dendogram level and the sequencing of the corresponding families a minimum costs criterion is proposed. The different operating costs of disassembling products in a RDS can be defined as in (Galan, Racero, Eguia & Canca, 2007), i.e.:

- Reconfiguration costs: costs due to changing the configuration of the RDS (machines and modules) when changing the product family to disassemble. These costs include:
    - Cost of adding or removing from the system a module k from a machine m (α_{mk})
    - Cost of removing from the system a machine m currently not required (β_m)
    - Cost of adding a required machine m (γ_m)
- Underutilization costs: costs for not using at the same level all the resources assigned to the system for the disassembly of all the products in a family. These costs include:
    - Cost per unit time of underutilization of module k in a machine m (ε_{mk})

From the above costs coefficients the reconfiguration costs ($R_{ss'}$) of changing from disassembly the products of family s to the products of family s', both families belonging to the same dendogram level can be computed as the total costs of adding and removing machines and modules

$$R_{ss'} = \sum_{m=1}^{M} \beta_m \cdot \Delta(\sum_{i\in s} \sum_{k=1}^{K} PT_{imk}, \sum_{i\in s'} \sum_{k=1}^{K} PT_{imk}) + \sum_{m=1}^{M} \gamma_m \cdot \Delta'(\sum_{i\in s} \sum_{k=1}^{K} PT_{imk}, \sum_{i\in s'} \sum_{k=1}^{K} PT_{imk}) + \sum_{m=1}^{M} \sum_{k=1}^{K} \alpha_{mk} \cdot (\Delta(\sum_{i\in s} PT_{imk}, \sum_{i\in s'} PT_{imk}) + \Delta'(\sum_{i\in s} PT_{imk}, \sum_{i\in s'} PT_{imk}))$$

Equation 4. Reconfiguration costs

where the indicator functions Δ and Δ' are defined as: Δ(a,b)=1 if a>0 and b=0; Δ'(a,b)=1 if a=0 and b>0.

Also, the resource underutilization costs (H_s) associated to a family s can be computed as the costs of not maximally using all the resources allocated to the family. Thus, compared with the resource with maximum utilization, all the other resources incur costs proportional to the difference in utilization levels, i.e.

$$H_s = \sum_{m=1}^{M}\sum_{k=1}^{K} \varepsilon_{mk} \cdot \left(\underset{\forall(m',k')}{Max} \left(\sum_{i \in s} PT_{im'k'} \right) - \sum_{i \in s} PT_{imk} \right)$$

Equation 5. Underutilization costs

Given the reconfiguration and underutilization costs computed above the problem of finding the optimal dendogram families and their sequencing corresponds to solving a Travelling Salesman Problem (TSP) for each dendogram level and selecting the level with minimum cost. There exist in the literature different MILP formulations of TSP (Lawler, Lenstra, Kan & Shmoys, 1992). In this paper a network-flow MILP model is used based on the three-index VRP formulation of Christofides, Mingozzi and Toth (1981).

Data

- i, j: Indexes of the product families to disassemble
- l: Index of the dendogram levels
- L: Number of dendogram levels (level 1= each product forms a family; level L = all products belong to the same, single family)
- F_l: Set of product families at dendogram level l, $l = 1,..,L$
- N_l: Number of product families at dendogram level l, i.e., Nl=|Fl|
- R_{ijl}: Reconfiguration cost of changing from family i to family j both belonging to dendogram level ($i \in F_l \quad j \in F_l \quad j \neq i \quad l = 1,..,L-1$)
- H_{il}: Resource underutilization cost of family i of dendogram level l ($i \in F_l \quad l = 1,..,L$)

Variables

- T_{ijl} = 1, if products of family i in dendogram level l are disassembled just before those of family j both belonging to dendogram level l ($i \in F_l \quad j \in F_l \quad j \neq i; \quad l = 1,..,L$)

- K_l = 1, if the product families of dendogram level l are selected ($l = 1,..,L$)
- $U_{il} \geq 0$, ancillary variables used to prevent family assignment cycles in each dendogram level ($i \in F_l \quad l = 1,..,L-3$)

Objective function

The objective function is the minimization of the sum of reconfiguration costs between families plus underutilization of resources assigned to each family.

MILP Model

$$Min \quad \sum_{l=1}^{L-1} \sum_{i \in F_l} \sum_{\substack{j \in F_l \\ j \neq i}} R_{ijl} T_{ijl} + \sum_{l=1}^{L} K_l \sum_{i \in F_l} H_{il}$$

s.a

$$(0): \quad \sum_{l=1}^{L} K_l = 1$$

$$(1): \quad \sum_{i \in F_l} T_{ijl} = K_l \quad : l = 1,..,L-1; \quad j \in F_l$$

$$(2): \quad \sum_{j \in F_l} T_{ijl} - \sum_{j \in F_l} T_{jil} = 0 \quad : i \in F_l; \quad l = 1,..,L-1$$

$$(3): \quad \sum_{\substack{j \in F_l \\ j \neq 1}} T_{1jl} = K_l \quad : l = 1,..,L-1$$

$$(4): \quad N_l \cdot T_{ijl} + U_{il} - U_{jl} \leq N_l - 1 \quad : i \in F_l - 1; \quad j \in F_l - 1; \quad j \neq i; \quad l = 1,..,L-3$$

$$(5.1): \quad T_{ijl} = [0,1] \quad : i \in F_l; \quad j \in F_l; \quad j \neq i; \quad l = 1,..,L-1$$

$$(5.2): \quad K_l = [0,1] \quad : l = 1,..,L$$

$$(5.3): \quad U_{il} \geq 0 \quad : i \in F_l - 1; \quad l = 1,..,L-3$$

Equation 6. MILP model

Constraints

0: One and only one dendogram level is to be selected

1: The families corresponding to the selected dendogram level will be sequentially processed once in each processing cycle. The families corresponding to the other dendogram levels will not be considered

2: For each family of the selected dendogram level there must be exactly one predecessor family and another successor family both of the selected level

3: Since the different families selected are sequenced cyclically, the first one can be any of them

4: Anti-cycle constraints

5: Integrality (variables T and K) and non-negativity (variables U) constraints

3 Numerical illustration

In order to validate the proposed methodology, in this section a simple example with five product types and four RMT is presented. Tables 1 and 2 show the weekly product volumes, the corresponding disassembly tasks (together with their duration) and the RMT plus auxiliary modules required by each operation type.

Product	Volume	Operation type (Time in seconds)
P1	80	O1 (140), O2(115), O5(80)
P2	55	O1 (70), O2 (50), O4 (55), O5 (130)
P3	25	O3 (120), O4 (85), O5 (125)
P4	70	O1 (110), O2 (65), O4 (100), O5 (90)
P5	30	O2 (75), O3 (95), O4 (105)

Table 1. Weekly product volumes and disassembly tasks

Operation type	RMT	Auxiliary modules
O1	M4	7, 8, 9, 11, 12
O2	M3	7, 8, 12
O3	M1	9, 11, 12
O4	M2	7, 8, 9, 10, 11
O5	M4	8, 10, 11

Table 2. Resources required by each operation type

The maximum number of auxiliary modules than any of the RMT can support (U_m) is 6. We assume that all units of each product require all the disassembly tasks, i.e. the relative of all disassembly tasks (f_{ij}) is unity. Finally, to simplify the computations let us assume that the cost coefficients are $\alpha_{mk}=1$, $\beta_m=10$, $\gamma_m=10$ y $\varepsilon_{mk}=0.001$ for all machines m and modules k.

3.1 Phase 1: Hierarchical clustering algorithm

Table 3 shows the similarity coefficients $S_{ii'}$ computed for each pair of products using the Equation 2.

	Product			
Product	**P2**	**P3**	**P4**	**P5**
P1	0,406	0,077	0,516	0,053
P2		0,274	0,591	0,299
P3			0,174	0,469
P4				0,196

Table 3. Similarity coefficients for each pair of products

According to the ALC algorithm, initially each product is assigned to its own family. The first grouping step would involve the two products with highest similarities,

which in this case are P2 and P4 (shown in bold in the table). Updating the similarity coefficient matrix after that grouping, using the average similarity criterion (Equation 3), leads to the leftmost matrix of table 4. The highest similarity coefficient in that table corresponds to grouping P3 and P5 into a new family, {P3,P5}. The next grouping step corresponds to the matrix in the center of table 4 and would result in adding P1 to family {P2,P4}. The next and last step is merging both families. Figure 2 shows the corresponding dendogram. Note that in this example no grouping is infeasible in terms of requiring more module on a machine than this can support.

	Family		
Family	**P1**	**P3**	**P5**
P2,P4	0,461	0,224	0,247
P1		0,077	0,053
P3			0,469

	Family	
Family	**P3,P5**	**P1**
P2,P4	0,236	0,461
P3,P5		0,065

	Family
Family	**P3,P5**
P2,P4,P1	0,179

Table 4. Similarity coefficients for successive grouping steps

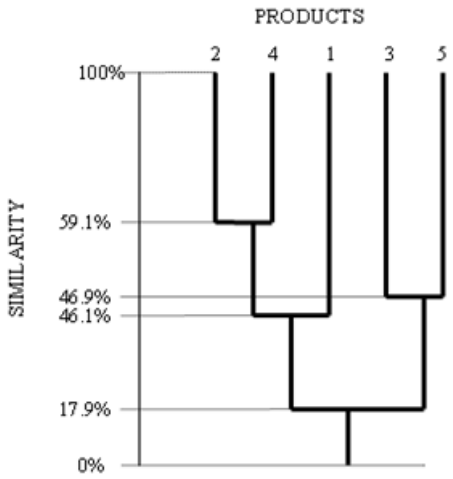

Figure 2. Dendogram representing successive grouping steps

3.2 Phase 2: Family sequencing model

The application of model (Equation 6) to this instance involves 45 binary variables, 9 continuous variables and 53 constraints. The model has been solved using optimization software CPLEX giving an optimal solution of 319,33. The binary variables selected are $K_3=1$, $T_{\{P1\},\{P3P5\},3}=1$, $T\{_{P3P5\},\{P2P4\},3}=1$ and $T_{\{P2P4\},\{P1\},3}=1$. This means that the optimal solution corresponds to dendogram level 3 which, looking at Figure 2, corresponds to three families, namely {P1}, {P2,P4} and {P3,P5}. The variables T_{ijl} indicate the minimum cost order in which the three families must be sequenced. In this case, there are two possible sequences:

{P1}→{P2,P4}→{P3,P5}→{P1} and {P1}→{P3,P5}→{P2,P4}→{P1}

The sequence obtained by CPLEX is the second one. The minimum cost solution of model found by CPLEX has been validated computing, using Equations 4 and 5, the costs of all possible family sequences for all levels of the dendogram of Figure 2.

3.3 Sensitiviy analysis with respect to cost coefficients

The reconfiguration cost coefficiencts (α_{mk}, β_m, γ_m) and the underutilization cost coefficients (ε_{mk}) to be used in Equations 4 and 5 respectively must be estimated based on previous experience (i.e. historic data). The former take into account whether a machine or an auxiliary module is or not required by one family but not by the one that follows it. The latter corresponds to the cost per unit time of not using an auxiliary module that has been included in the configuration for disassembling a given family. In the first dendrogram level families are formed by a single product and therefore, for each family, the RDS will be configured specifically for one product which means that there will not be underutilization costs although there will be frequent reconfigurations. In other words, for low values of the dendrogram level reconfigurations costs are higher and underutlization cost lower. An the contrary occurs for larger values of the dendrogram level. Hence, the lower the values of α_{mk}, β_m, γ_m and the higher the values of ε_{mk} the smaller the families and the higher the number of reconfigurations. Note that the underutilization cost ε_{mk} is different to the other cost coefficients since it is measured per unit of time.

Table 5 shows the results of solving model from Equation 6 (and checking the results by manual computation of the best family sequence for each dendrogram level) for different values of the cost coefficients.

Cost coefficients				Minimum cost solution				Best solution costs				
α_{mk}	β_m	γ_m	ε_{mk}	Sequence	RC	UC	TC	Level 1	Level 2	Level 3	Level 4	Level 5
1	10	10	0.001	P1→{P3,P5}→{P2,P4}	62	257.33	319.33	358.45	358.45	**319.33**	377.33	444.35
0.1	1	1	0.001	P1→{P2,P4}→P3→P5 P1→P2→P4→P3→P5	11.4	244.45	255.85	**255.85**	**255.85**	263.53	348.53	444.35
1	10	10	0.0001	{P1,P2,P3,P4,P5}	0	44.44	44.44	138.45	138.45	87.73	66.53	**44.44**
1	10	10	0.01	P1→{P2,P4}→P3→P5 P1→P2→P4→P3→P5	114	2444.5	2558.5	**2558.5**	**2558.5**	2635.25	3485.25	4443.5
10	100	100	0.001	{P1,P2,P3,P4,P5}	0	444.35	444.35	1384.45	1384.45	877.33	665.33	**444.35**

Table 5. Results for different cost coefficients (RC=Reconfigurations cost, UC=Underutilization cost, TC=Total cost)

4 Conclusions and further research

The actual disassembly plants are mainly based on the manual labor and great effort has been made by firms to automate their disassembly processes. Recent researches have studied different aspects of disassembly automation but expectation on automated disassembly has not yet fulfilled owing to the severe constraints on disassembly technologies. Reconfigurable manufacturing technologies can attain some degree of adaptability during disassembly automation design using modular machine tools. In this paper an innovative methodology for the design of a new automated disassembly system that uses reconfigurable technologies have been presented. The new automated disassembly system has been called Reconfigurable Disassembly System (RDS) and its particular modular structure has been included into the two phase methodological approach.

In the first phase products are grouped into families taking into account the similarities among the products to disassemble. Appropriate similarity coefficients have been proposed using the resources, the processing times required for disassembling each product and the number of units of each product. A hierarchical clustering approach (namely Average Linkage Clustering Algorithm, ALC) has been used, representing in a dendogram the successive possible groupings (i.e. families) that may be formed. The second phase consists in a Mixed Integer Linear Programming (MILP) model that selects the families to consider and their processing sequence. This is done using a minimum cost criterion that considers both the costs of reconfiguring the RDS between two consecutive families (thus adding/removing needed/unneeded machines and modules) and the cost due to the underutilization of the assigned resources.

In order to illustrate and validate the methodology a small-size instance has been solved numerically and with the proposed model using the CPLEX software. The MILP model proposed for the second phase for family sequencing is of Travelling Salesman Problem (TSP) and therefore its complexity grows exponentially with the number of products to disassemble. That is why in real-world problems in which the number of products is usually high it may be advisable to solve the model approximately with a metaheuristic. Further research is required also to complete the methodology with two additional phases: one for determining, for each product, the optimal sequence of its disassembly tasks and another for computing the optimal (i.e. with minimum makespan) intra-family product sequence.

Acknowledgments

This research has been fully funded by the Spanish Ministry of Science and Innovation through grants DPI2008-04788, DPI2010-16201 and FEDER.

References

Christofides, N., Mingozzi, A., & Toth, P. (1981). Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations. *Mathematical Programming*, 20, 255–82.

Duta, L., & Filip, F.G. (2008). Control and decision-making process in disassembling used electronic products. *Studies in Informatics and Control*, 17, 17–26.

Galan, R., Racero, J., Eguia, I., & Garcia, J.M. (2007). A systematic approach for product families formation in Reconfigurable Manufacturing Systems. *Robotics and Computer Integrated Manufacturing*, 23, 489-502. http://dx.doi.org/10.1016/j.rcim.2006.06.001

Galan, R., Racero, J., Eguia, I., & Canca, D. (2007). A Methodology for facilitating reconfiguration in manufacturing: The move towards reconfigurable manufacturing systems. *International Journal of Advanced Manufacturing Technology*, 33, 345-353. http://dx.doi.org/10.1007/s00170-006-0461-2

Güngör, A., & Gupta, S.M. (1997). An evaluation methodology for disassembly processes. *Computers & Industrial Engineering*, 33, 329–332.

Ilgin, M.A., & Gupta, S.M. (2010). Environmentally conscious manufacturing and product recovery (ECMPRO): A review of the state of the art. *Journal of Environmental Management*, 91, 563–591.

Jain, P.K., Fukuda, Y., Komma, V.R., & Reddy, K.V.S. (2006). Performance modelling of Reconfigurable Assembly Line. *International Journal of Simulation Modelling*, 5(1), 16-24. http://dx.doi.org/10.2507/IJSIMM05(1)2.049

Koren, Y., Heisel, U., Jovane, F., Moriwaki, T., Pritschow, G., Ulsoy, G., & Van Brussel, H. (1999). Reconfigurable Manufacturing Systems. *Annals of the CIRP*, 48, 1-14. http://dx.doi.org/10.1016/S0007-8506(07)63232-6

Lawler, E., Lenstra, J., Kan, A., & Shmoys, D. (1992). *The Traveling Salesman Problem*. New York: John Wiley and Sons.

Seliger, G., Basdere, B., Keil, T., & Rebafka, U. (2002). Innovative processes and tools for disassembly. *CIRP Annals – Manufacturing Technology*, 51, 37–40.

Tang, Y., Zhou, M., Zussman, E., & Caudill, R. (2002). Disassembly modeling, planning and application. *Journal of Manufacturing Systems*, 21(3), 200-217. http://dx.doi.org/10.1016/S0278-6125(02)80162-5

Vakharia, A., & Wemmerlov, U. (1995). A comparative investigation of hierarchical clustering techniques and dissimilarity measures applied to the cell formation problem. *Journal of Operation Management*, 13, 117–138.

Wiendahl, H.P., Scholz-Reiter, B., Bürkner, S., & Scharke, H. (2001). Flexible disassembly systems-layouts and modules for processing obsolete products. *Proceedings of the Institution of Mechanical Engineers, Part B: Engineering Manufacture*, 215, 723–732.

Xiaobo, Z., Jiancai, W., & Zhenbi, L. (2000). A Stochastic Model of a Reconfigurable Manufacturing System, Part 1: A Framework. *International Journal of Production Research*, 38(10), 2273-2285. http://dx.doi.org/10.1080/00207540050028098

Yu, V, Yin, Y., Sheng, X., & Chen, Z. (2003). Modelling strategies for reconfigurable assembly systems. *Assembly Automation*, 23(3), 266-272. http://dx.doi.org/10.1108/01445150310486530