Development Overview of an Information Model for Disassembly

Shaw C. Feng[1], Hanmin Lee[1,2], Che Bong Joung[1], and Parisa Ghodous[3]

[1] National Institute of Standards and Technology, Gaithersburg, USA

[2] Korea Institute of Machinery and Materials, Daejeon, South Korea

[3] University of Lyon I, Lyon, France

Abstract

Manufacturing is a fundamental support of an industrialized society. Currently, manufacturing industry is facing major challenges, such as resources depletion, global climate change, environmental pollution, and ensuring the social well being of future generations. Reuse and recycle become necessary for sustainable development. Disassembly of end-of-service-life products is a key operation to separate the product into reusable and recyclable parts. This paper describes an information model for disassembly in the Unified Modeling Language. A literature study of the available literature on design for disassembly, disassembly process planning, and sustainable manufacturing has been conducted. A description of the classes and their relationships on disassembly features, disassembly equipment, disassembly workflow, and operations of separations, cleaning, and inspection is included in the paper. A case study on a car suspension design is conducted to initially test the model.

Keywords:

Disassembly Information Model; Sustainable Manufacturing; Unified Modeling Language

1 INTRODUCTION

Manufacturing is the fundamental support of an industrialized society. Two major challenges that manufacturing industries are facing today is global resource depletion and environmental pollution. To decrease the depletion of natural resources and reduce the release of eco-toxic materials to the environment, recycle and reuse of products is an effective method.

For reuse and recycle of a product at the end of its useful life, disassembly is the key operation to separate the product into reusable, recyclable, and disposable parts. A disassembly process includes the separation of functional parts from the rest of the product for reuse, the cleaning of usable parts, and the inspection for the quality of the separated parts in reuse. Disassembly process planning requires information from design, such as disassembly operation, sequence, disassembly features, their relationships, and subassemblies. Disassembly process planning systems need also to exchange disassembly process plans, including task plan, equipment requirements, resource requirements, disassembly sequence, and cost estimates with other design and process planning systems. Designers require the cost of disassembly and available equipment to determine disassembleability of a design. Figure 1 shows that design for disassembly data need to be exchanged among eco-Computer-Aided Design (ecoCAD) systems and disassembly process planning systems. Disassembly plans need to be exchanged among disassembly process planning systems. Hence, an information model is necessary to share and exchange disassembly information among lifecycle applications, such as design and disassembly planning.

This paper describes an information model for disassembly in the Unified Modeling Language [1]. Section 2 has a study of the available literature on design for disassembly, disassembly process planning, and sustainable manufacturing. Section 3 describes the classes and their relationships in the developed information model. Section 4 provides a case study to test the model. Section 5 concludes with possible future directions for information model development.

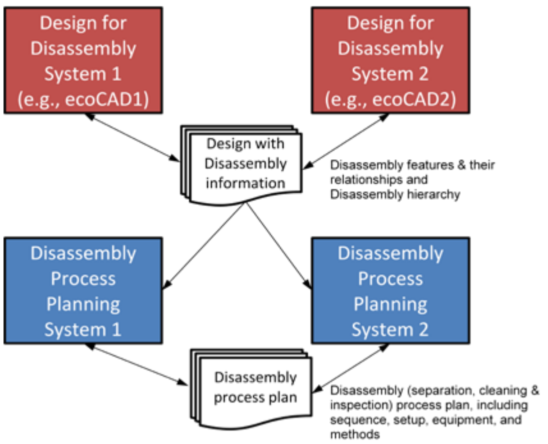

Figure 1: Information sharing among ecoCAD systems and disassembly process planning systems.

2 CURRENT STATUS OF DISASSEMBLY INFORMATION

Many available publications on why disassembly is important, disassembly representation in design, and disassembly process plan representations are reviewed in this section.

Gaps in a comprehensive disassembly information model are discussed at the end of the section.

2.1 Design for disassembly

Disassembleability analysis of a product and the modeling of disassembly sequence has been developed in [2]. An example for possible disassembly path of components of an assembly is analyzed and is shown in [3]. Disassemblability analysis is a base for design for disassembly. Disassembly should be considered during a product design so that the product will be timely and cost effectively disassembled at the end of service life. The development of algorithms has been done to search for the optimal level of disassembly of a product based on the costs and benefits of the extent to which a product should be disassembled, i.e., the number of components in a product should be disassembled [4, 5]. Design analysis can be extended from disassembleability to remanufactureability, reusability, and recycleability [6].

2.2 Modeling strategy for disassembly sequences and process planning

Over the past decade, many modeling strategies have been proposed, e.g., AND/OR graph, Disassembly Petri Net (DPN), and component-fastener graph. Among them, the AND/OR graph is the most popular modeling strategy and forms the basis for much of the later work. The nodes and the hyperarcs in these AND/OR graphs, respectively, correspond to subassemblies and disassembly tasks in which a more complex subassembly is separated into two subassemblies. Homem de Mello and Sanderson [7] presented an AND/OR graph approach to model all the possible disassembly sequences of a system. Lambert [8] used a so-called disassembly graph, which was based on the AND/OR graph and liaison analysis, to derive optimal disassembly sequences.

The DPN approach is applied by Moore et al. [9] for including complex AND/OR relationships in disassembly diagrams. Petri nets for adaptive planners and some associated experimental results are in Zussman and Zhou [10]. An adaptive planner, based on product Petri nets and workstation Petri nets, which modifies the disassembly sequence according to the condition of the items in a batch, is presented by Tang et al. [11].

Kuo et al. [12] proposed a non-directed graph-based heuristic approach for the generation of the disassembly sequence for recycling. A product is modeled by a component-fastener graph. By identifying the “cut-vertices,” the search splits the graph into subgraphs until a disassembly tree is formed. Based on the disassembly tree, disassembly sequences can then be generated. Li et al. [13] proposed a disassembly constraint graph (DCG). Using the DCG, all the possible disassembly operations that are needed for the maintenance of certain components or subassemblies can be deduced.

2.3 Information model for disassembly

There lacks an open, enabling information model of disassembly. Assembly information models can be references for developing a disassembly information model. ISO 10303-44 [14] provides some limited assembly design representations that capture assembly structure and kinematic joint information during the design process. The model focuses on the hierarchy of the product, and on the position and orientation between parts. An ISO working group, TC 184/SC4/WG12 Japan National Committee proposal [15], has proposed several enhancements to ISO 10303-44 assembly representation. In the WG12 proposal, the detailed geometric information is introduced not only for hierarchical relationships but also for peer to peer relationships among component parts via the assembly features.

An integrated NIST Core Product Model (NIST-CPM) [16] has been developed to unify and integrate product or assembly information. The core model focuses on artifact representation including function, form, behavior, and material, physical, and functional decompositions, and relationships among these concepts. The aim of the Open Assembly Model (OAM) [17], which uses the NIST CPM, is to provide a standard representation and exchange protocol for assembly and system-level tolerance information. The assembly information model emphasizes the nature and information requirements for part features and assembly relationships.

2.4 High level Requirements for an information model

Based on the literature review, the following gaps can be identified in modeling information of integrated design for disassembly and disassembly process planning:

1. A comprehensive feature representation for simple, complex, and compound disassembly features and separation between features from two component parts to be disassembled.
2. A representation of disassembly task sequence that is based on a disassembly sequence.
3. A representation of disassembly equipment and methods that is associated with task representation in disassembly process planning.
4. A representation of cleaning and inspection processes following the separation of assemblies of an end-of-service-life product.

3 DISASSEMBLY INFORMATION MODEL

This section describes classes and their relationships in the Feature package, the Equipment package, and the Workflow package in the developed information model.

3.1 Feature

A disassembly feature represents a portion of a part that is of interest in disassembly, including separation, cleaning, and inspection. From the complexity point of view, a simple feature is a baseline feature. From the geometric point of view, a simple feature can be a point, 2D, or 3D feature. Composite and pattern features are defined based on simple feature. Specific feature representations for disassembly applications are in the subpackages. Figure 2 shows the diagram of the feature package.

Simple Form Features

Based on simple 3D features, specific form features are defined for applications in design for disassembly at the end of a product useful life and disassembly process planning.

They are all subtypes of SimpleFeature. Figure 3 shows the diagram of simple form feature classes.

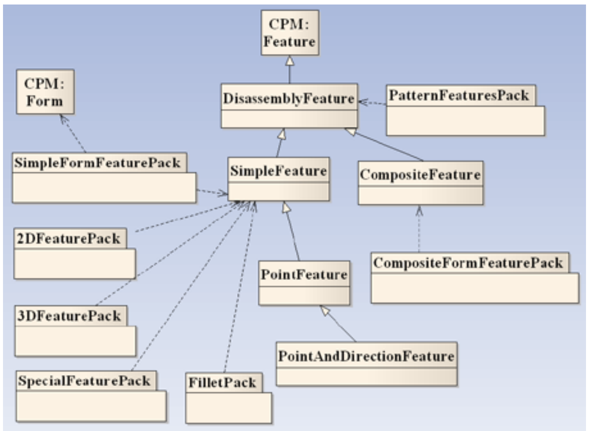

Figure 2: Diagram of the feature package.

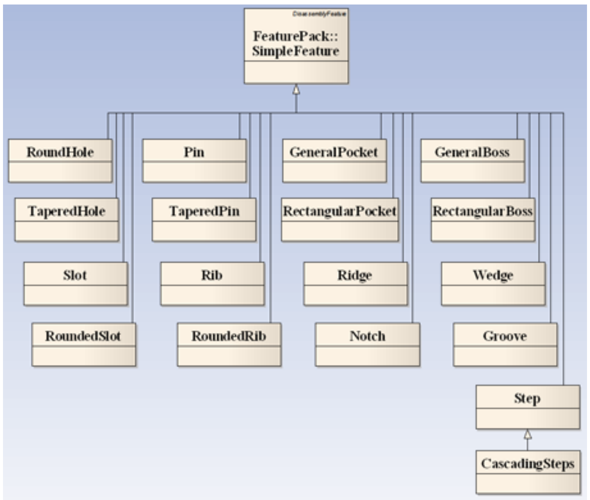

Figure 3: Diagram of simple form feature classes.

Composite Form Features

Composite form features are used for representing form features that are composed of simple form features to create complex form features. Figure 4 shows the diagram of composite form feature classes.

Class Separator is a subtype of Class Connector, defined in the Open Assembly Model, and is used to represent a separator in an assembly that is to be separated at the end of its useful life. It has two attributes. Attribute *feature* represents the feature from that the assembly will be separated into subassemblies, and its data type is SimpleFeature. Attribute *type* represents the type of the separator, and its data type is SeparatorType. Enumeration type SeparatorType is used to define the type of a separator so that proper disassembly tools can be selected. This enumeration type includes *Bolt-Nut-Washer*, *Screw, TaperFit*, *PinFit*, *Rivet*, *SplitFit*, *RubberRing*, *SpringFit*, *Bearing*, *GearMesh*, *BeltMesh*, *Glue*, and *Welding*.

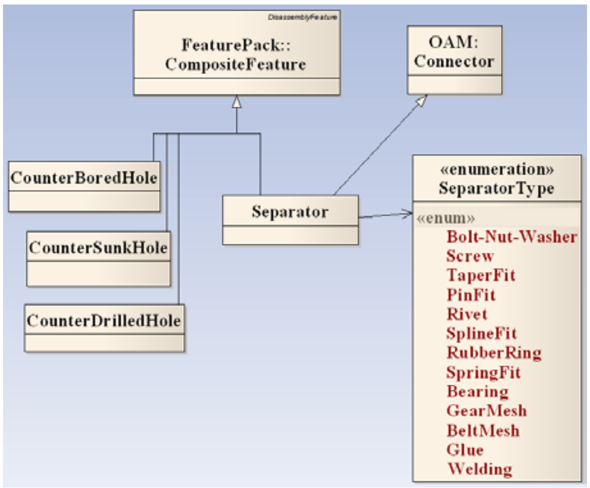

Figure 4: Diagram of composite form feature classes.

3.2 Equipment

The Equipment package includes classes that represent a piece of equipment used in disassembly. The package includes three subpackages. They are separation equipment package, cleaning equipment package, and dimensional measurement equipment package.

Separation Equipment

The Separation Equipment subpackage includes classes that represent a piece of equipment used in separating an assembly into subassemblies or individual parts. Figure 5 shows the diagram of classes in the subpackage.

Class SeparationEquipment is used to represent a piece of equipment used in separating an assembly into subassemblies or individual parts for reuse, recycle, or remanufacturing. The class is a subtype of Equipment and has two attributes. Attribute *tools* represents a set of separation tools, and the attribute's data type is a set of SeparationTool. Attribute *workstation* represents a separation workstation, and the attribute's data type is SeparationWorkstation. Enumeration type SeparationTool-Type is used to list types of tools used in separation. This enumeration type includes *ScrewDriver*, *HandDrill*, and *Knife*. The list can be extended when it is necessary. Enumeration type SeparationWorkstationType is used to list types of workstation used in separation. This enumeration type includes *DestructiveDisassemblyWorkstation*, *Nondestructi-veDisassemblyWorkstation*, *ToolChangingSystem*, and *Cryo-technicalWorkstation*. The list can be extended when it is necessary.

3.3 Workflow

The Workflow package includes classes that represent workflow in manufacturing. Figure 6 shows the diagram of classes and subpackages of Boolean expression and Operation in the package.

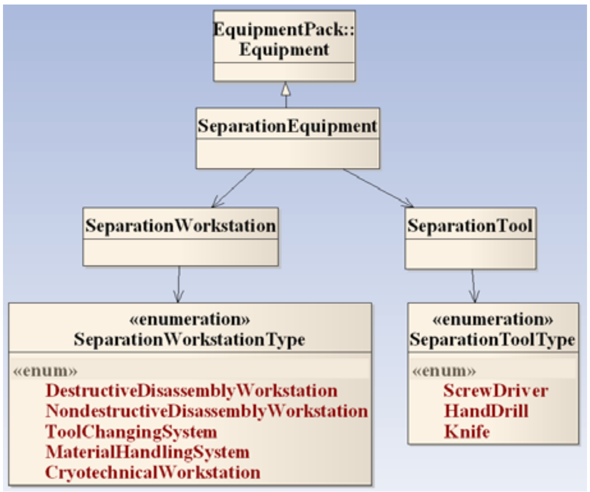

Figure 5: Diagram of classes in the Separation Equipment subpackage.

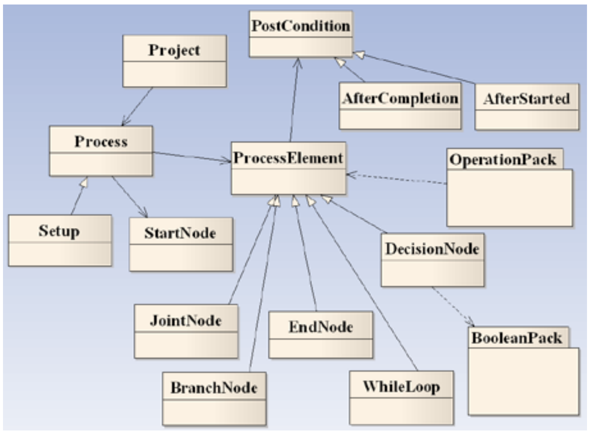

Figure 6: Diagram of the Workflow package.

Operation

The Operation subpackage includes classes that represent operations in disassembly, such as separation, cleaning, and dimensional inspection. Figure 7 shows the diagram of classes in the subpackage.

Class Separation is used to represent a separation operation. The class is a subtype of Operation and has six attributes. Attribute *appliedTo* represents the assembly to which the separation operation is applied, and the attribute's data type is OAM:Assembly. Attribute *cost* represents the cost that is associated with the separation, and the attribute's data type is MeasureWithUnit. Attribute *name* represents the name of the operation, and the attribute's data type is String. Attribute *subassemblies* represents the separated subassemblies as the result of the separation, and the attribute's data type is a set of OAM:Assembly. Attribute *tool* represents the tool used in the separation operation, and the attribute's data type is SeparationTool. Attribute *workstation* represents the workstation used in the separation operation, and the attribute's data type is SeparationWorkstation. Enumeration type MethodType is used to list types of separation operations.

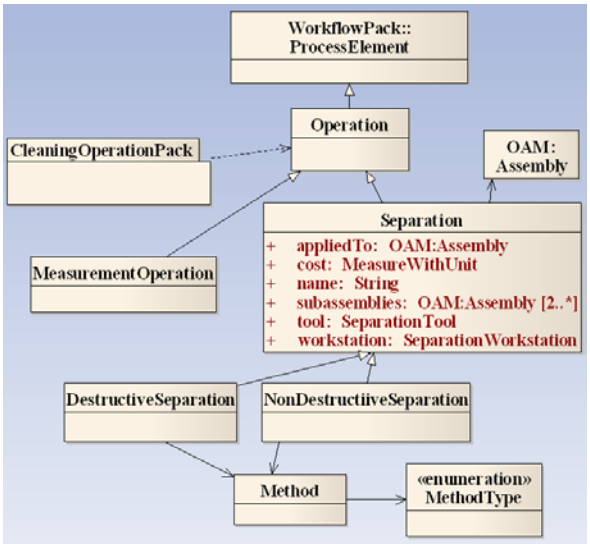

Figure 7: Diagram of classes in the Operation subpackage.

This enumeration type includes *Drilling*, *Unscrewing*, *Knife-cutting*, *Strike-cutting*, *Splitting*, *Shearing*, *Abrasive-cutting*, *WaterJetCutting*, *CryotechnicalSeparation*, *SnapOff*, *Shredding*, and *Sorting*. The list can be extended when it is necessary.

Class CleaningOperation is used to represent a cleaning operation. The class is a subtype of Operation and has four attributes. Attribute method represents a chosen cleaning method used in a cleaning operation, and the attribute's data type is a set of CleaningMethod. Attribute part represents the part or subassembly to which the cleaning operation is applied, and the attribute's data type is OAM:Assembly. Attribute tool represents the tool used in the cleaning operation, and the attribute's data type is CleaningTool. Attribute workstation represents the workstation used in the cleaning operation, and the attribute's data type is CleaningWorkstation.

Class MeasurementOperation is used to represent a dimensional measurement operation. The class is a subtype of Operation and has five attributes. Attribute *consumerRisks* represents the consumer risks that are associated with the measurement results, and the attribute's data type is a list of ConsumerRisk. Attribute *reportRequirements* represents the requirements of reporting measurement results, and the attribute's data type is a list of ReportingRequirement. Attribute *samplingStrategy* represents any specified sampling strategy associated with the measurement operation, and the attribute's data type is optional SamplingStrategy. Attribute *sensors* is used to represent sensors used in dimensional measurement of a feature, and the attribute's data type is a set of *Sensor*. Attribute *toleranceToBeVerified* represents the tolerance to be verified, and the attribute's data type is Tolerance.

4 CASE STUDY

A car suspension module is used to illustrate the disassembly information model. The car suspension module is assumed to be composed of four parts (*1-4*) and two sub-assemblies (*A* and *B*), as shown in Figure 8. The sub-assembly *A* contains two parts (*5* and *6*), and the sub-assembly *B* contains five parts (*7-11*) and a sub-assembly (*C*), which further decomposes into five parts (*12-16*). Figure 9 shows the instance diagram of the disassembly information model which contains assembly hierarchy of the car suspension module. The instance diagram has a hierarchical relationship between assemblies and parts as well as connection relationships between parts.

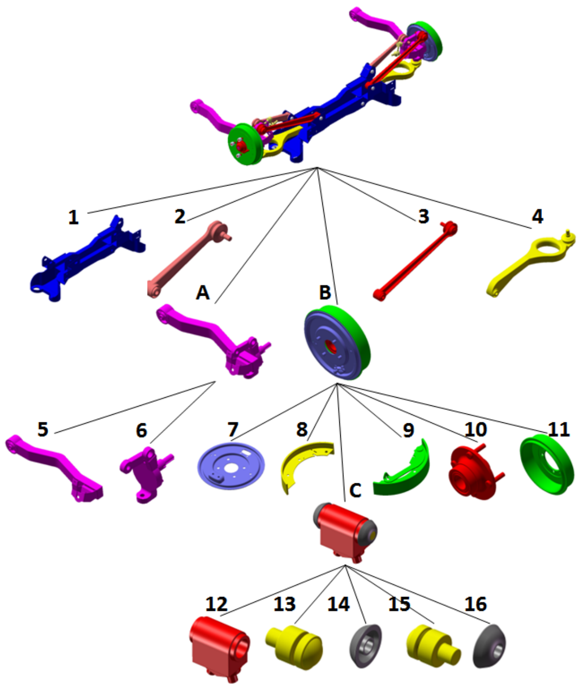

Figure 8: Assembly hierarchy of a car suspension module.

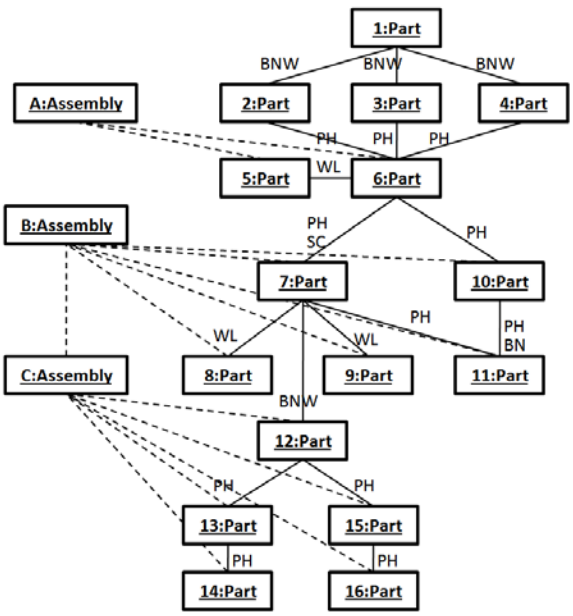

Figure 9: Instance diagram of the car suspension module assembly.

An example of the representation of a disassembly sequence using the disassembly information model is shown in Figure 10. A project is composed of several processes, each process has several sub-processes. For disassembly, a sub-process can be a non-destructive or destructive separation. A separation class has an assembly as input and two sub-assemblies as output, and it needs several tasks to separate the input assembly into sub-assemblies.

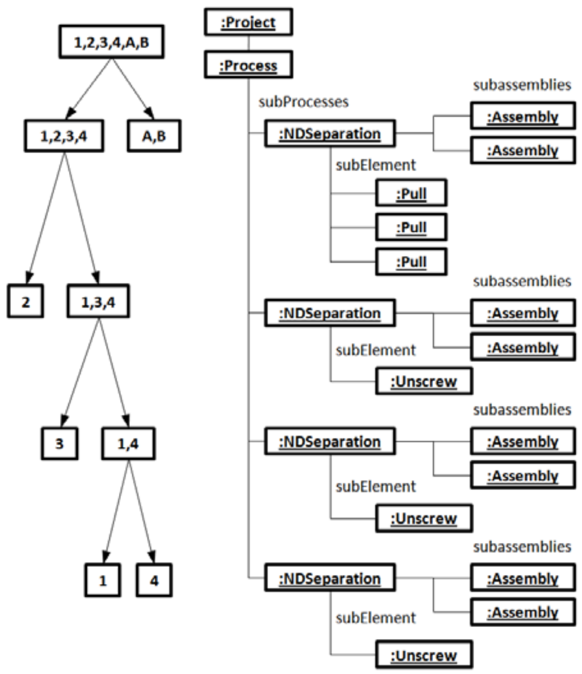

Figure 10: Instance diagram of disassembly sequence of the car suspension module.

5 CONCLUSIONS AND FUTURE WORK

Manufacturing industries are facing the challenge of reuse and recycle of products at their end of service lives. The literature study shows that the number of environmental protection regulations is increasing rapidly. Reuse and recycle are critical activities to alleviate natural resource depletion and save energy to achieve the goal of sustainable development. Disassembly of end-of-service-life products is a key operation to separate the product into reusable and recyclable parts. Information on design for disassembly and disassembly process is critical for decision making in design and manufacturing. An information model for disassembly is, hence, developed using the Unified Modeling Language. The model includes all the classes and their relationships on disassembly features, tolerances, dimensions, destructive disassembly, nondestructive disassembly, disassembly methods, equipment, and disassembly workflow. A study of a case, a vehicle suspension design is conducted to initially test to the validity of model. Specifically, information classes of part and feature relationships of parts or subassemblies to be separated, disassembly sequence, disassembly processes have been tested. The other portion of the model is based on available standards.

Possible future work includes more comprehensive tests with more complicated design needed to test the model. Prototype disassembly database, cost of disassembly analysis software, disassembly process planning systems can be developed using the information model. The investigation on how to use the information model for design for sustainability is also needed. Finally, standard data exchange format for design for disassembly and disassembly process plans can also be developed.

ACKNOWLEDGEMENT

The authors gratefully acknowledge Mr. Gwangsub Chang of Ajou University for providing design models for the case study.

6 REFERENCES

[1] Rumbaugh, J., Jacobson, I., Booch, G., 1999, The Unified Modeling Language Reference Manual, Addison- Wesley, Reading, MA.

[2] Tang, Y., Zhou, M. C., Caudill, R. J., 2000, "An integrated approach to disassembly planning and demanufacturing operation," Proceedings of 2000 IEEE International Symposium on Electronics and the Environment, 354-359.

[3] Lee, K., Gadh, R., 1996, "Computer Aided Design for Disassembly: A Destructive Approach," Proceedings of the 1996 IEEE International Symposium on Electronics and the Environment, 173-178.

[4] Desai, A., Mital, A., 2003, "Evaluation of Disassemblability to Enable Design for Disassembly in Mass Production," International Journal of Industrial Ergonomics, 32, 265-281.

[5] Desai, A., Mital, A., 2005, "Incorporating Work Factors in Design for Disassembly in Product Design," Journal of Manufacturing Technology Management, 16/7, 712-732.

[6] Ijomah, W., McMahon, C., Hammond, G., Newman, S., 2007, "Development of Design for Remanufacturing Guidelines to Support Sustainable Manufacturing," Journal of Robotics and Computer-Integrated Manufacturing, 23, 712-719.

[7] Homem de Mello, L.S., Sanderson, A. C., 1990 "AND/OR graph representation of assembly plans," IEEE Transactions on Robotics and Automation, 6/2, 188-200.

[8] Lambert, A. J. D., "Optimal disassembly of complex products," International Journal of Production Research, Vol.35, No.9, pp.2509-2523, 1997

[9] Moore, K. E., Gungor, R. A., Gupta, S. M., 1998, "Disassembly process planning using Petri nets," Proceedings of 1998 IEEE Conference on Electronics and the Environment, 88-93.

[10] Zussman, E., Zhou, M. C., 2000, "Design and implementation of an adaptive process planner for disassembly processes," IEEE Transactions on Robotics and Automation, 16/2, 171-179.

[11] Tang, Y., Zhou, M., Zussman, E., Caudill, R., 2000, "Disassembly Modeling, Planning, and Application: A Review," Proceedings of the 2000 IEEE International Conference on Robotics & Automation, 2197-2202.

[12] Kuo, T.C., Zhang, H.C., Huang, S.H., 2000, "Disassembly analysis for electromechanical products: a graph-based heuristic approach," International Journal of production research, 38/5, 993-1007.

[13] Li, J.R., Khoo, L,P., Tor, S.B., 2005, "An object-oriented intelligent disassembly sequence planner for maintenance," Computers in Industry, 56, 699-718.

[14] ISO 10303-44, 1994: Industrial automation systems and integration – Product data representation and Exchange – Integrated Generic Resources: Product Structure Configuration.

[15] ISO TC 184/SC4/WG12 N597, JNC Proposal of STEP Assembly Model for Products, 2000.

[16] Fenves, S. J., "A Core Product Model For Representing Design Information," National Institute of Standards and Technology, NISTIR 6736, 2002.

[17] Sudarsan, R., Han, Y. H., Feng, S. C., Roy, U., Wang, F., Sriram, R. D., Lyons, K., "Object Oriented Representation of Electro-Mechanical Assemblies Using UML," National Institute of Standards and Technology, NISTIR 7057, 2003.