Computer-Aided Design and Applications

http://www.cadanda.com

Managing Extended Producer Responsibility using PLM
Part 2: Identification of Joints for End-of-life Treatment Planning

Nikhil Joshi[1] and Debasish Dutta[2]

[1]University of Michigan, njoshi@umich.edu
[2]University of Michigan, dutta@umich.edu

ABSTRACT

Concerns about the environmental impacts of used and discarded products have recently led to enactment of laws that regulate the amounts of hazardous substances and recyclable content in products. The laws also make the Original Equipment Manufacturers (OEMs) responsible for recovery and proper treatment of these end-of-life products. In this two part paper, we present methodologies for OEMs to use the Product Lifecycle Management (PLM) framework to effectively meet the challenges posed by these regulations. In this second part, we outline a methodology that can enable case-by-case selection of the treatment strategy for incoming end-of-life products. To extract product information required for this selection, we develop rule-based heuristics for identification of joints directly from CAD assembly models, along with their type, size, location and orientation.

Keywords: environmental regulations, joints, CAD, PLM.
DOI: 10.3722/cadaps.2008.764-773

1. INTRODUCTION

Extended Producer Responsibility laws, as explained in Part 1 [8] of this paper, are placing a considerable operational and financial burden on the Original Equipment Manufacturers (OEMs) of consumer products. In the first part of this paper, we presented a methodology to account for regulations during the selection of material and processing specifications for components at the early design stages. In this second part, we shall address the second aspect of the problem, namely, the proper treatment of used and discarded products.

Traditionally, the processing of used and discarded products (henceforth referred to as "end-of-life products") has been a small-scale, profit-driven activity restricted to junk-yards and recycling units for specific materials. However, as OEMs are forced to recover and recycle their end-of-life products, there is an emphasis on determining the optimal treatment strategy. The feasibility, costs, and returns of the treatment depend upon a number of variable factors, such as the tools available at the local ATFs, markets for refurbished goods and recycled material, damage to incoming parts, proximity of recycling and disposal facilities, local labor rates, etc. While deciding a treatment plan, OEMs must account for these local and temporal factors, along with technical factors such as product configuration, material composition, hazardous substances disposal regulations, recovery or recycling targets, etc. This undoubtedly requires close collaboration between the OEM, its suppliers, maintenance facilities and Authorized Treatment Facilities (ATFs). We present a systematic methodology that allows consideration of the above-mentioned factors during the selection of the treatment plan for each incoming end-of-life product.

This paper is organized as follows. Section 2 reviews prior work done towards disassembly and treatment of end-of-life products. Section 3 explains the envisaged methodology for case-by-case selection of the treatment plan. Towards facilitating the selection process we develop algorithms for identification of commonly occurring types of joints directly from CAD models. These algorithms are described in section 4. We finally conclude in section 5 by enumerating the advantages of the approach and future directions to extend the approach.

2. BACKGROUND

The determination of an end-of-life treatment plan involves the determination of;

- the disassembly operations required along with the costs, tools, methods, and accessibility requirements to accomplish them,
- the optimal sequence and extent of disassembly operations to be carried out, and
- the end-fates (such as recycling, refurbishment, or disposal) for each separated component or sub-assembly, along with the processing steps (e.g., cleaning, storage, transportation) to be completed before handing over the components to another organization (such as a landfill, a recycling unit, or a used part vendor).

While a holistic approach encompassing all these aspects of end-of-life treatment planning has not been observed in literature, significant portions of the problem have been tackled in the past.

A number of authors have investigated methods to determine the sequence and extent of disassembly operations. Homem de Mello and Sanderson [6] used an AND/OR graph representation to generate a complete set of feasible assembly sequences. The feasibility of a sequence is decided by reasoning on a "relation model", which is created by a human expert to include information about type of connections between parts, their precedence relationships for assembly, etc. The method can be extended to disassembly sequences. Subramani and Dewhurst [14] provide an algorithm to create a disassembly diagram by extracting precedence information from a user defined relation model. A branch and bound algorithm is then used to find the optimal path. Gungor and Gupta [5] use a time-based metric called "Total Time for Disassembly", to measure the efficiency of a disassembly sequence, and also provide a heuristic for generating the best sequence. Several techniques using graphical methods, empirical methods, simulated annealing, or mathematical programming, have been proposed to obtain the optimal disassembly sequence and extent for maximum net revenue. Lambert [9] uses linear programming to determine whether a particular operation should be carried out. Hula, et al. [7] use a genetic algorithm to find the optimal sequence, where the fitness function gives consideration to the different cost structures and environmental impacts in different geographical regions of the world. For these methods, the cost of each disassembly operation, its technical feasibility, precedence relations, and expected returns at each state of disassembly have to be provided by a human expert.

Towards integrating the choice of end-fates into the analysis, Navin-Chandra [11] presents a break-even analysis between the effort put into recovery of components and the effort saved by reusing parts and material using the traveling salesman method. Chen, et al. [2] also present a cost-benefit analysis to determine how much effort should be put into the disassembly and recycling of a product. Bras and Emblemsväg [1] studied the economics of disassembly under uncertain conditions using activity-based cost modeling.

All methods discussed above use more sophisticated representations of the product assembly than is directly available from commonly used CAD software. Graph representations or combined graph and matrix representations are popularly used [13]. The representations incorporate information about part contacts, joints and fastener types, tools and effort required for disassembly, precedence of operations, etc. Thus, information about possible disassembly operations, tools required, accessibility requirements, have to be evaluated and entered by human experts. Owing to the variety of different configurations available and the constant development of newer product models, manually defining this information, although possible, is often repetitive and time consuming. Also, the task of visualizing accessibility of joints is often difficult and unintuitive in the absence of a physical model. Mo, et al. [10] present a virtual reality based disassembly analyzer that assists the user in generating an accessibility graph and a removability graph, which can further be used by some of the methods discussed earlier. Use of virtual reality, however, cannot ensure that all possible methods of accessing the joints have been considered by the user. Thus, to facilitate the end-of-life treatment planning task, there is a need for a method to obtain the required assembly information directly from the commonly used representations of the products, such as CAD assembly models.

In this paper, we present a methodology to systematically accomplish all the planning tasks in an enterprise setting, while accommodating local and temporal factors that affect the suitability of a treatment plan. Towards satisfying the need to extract assembly information, we develop geometric algorithms to infer information about joints directly from CAD assembly models.

3. METHODOLOGY FOR SELECTION OF END-OF-LIFE TREATMENT PLAN

In this section, we describe a methodology for case-by-case selection of the treatment plan for incoming end-of-life products. The methodology assumes the presence of a functional PLM framework that integrates information between the ATF and the OEM along with its network of suppliers and maintenance facilities. The methodology also draws upon the approaches discussed in the previous section.

In order to model the problem, we suggest an efficient, graphical representation, called the “partition lattice” [12]. Consider a product assembly made up of *n* indivisible components or parts, labeled *1, 2, ..., n*. Then, the partition lattice π_n represents all possible ways in which the product can be separated into parts and sub-assemblies. For example, Fig. 1 shows the partition lattice π_4 for a product with four parts. Each node of the lattice is a partitioning (i.e., a set of subsets which have no common elements and includes all elements in the parent set) of the set {*1, 2, ..., n*}. Thus, we can consider each node as representing a state of disassembly, wherein all components in a partition are considered to be connected to form a sub-assembly. Correspondingly, each edge of the lattice represents a disassembly operation, separating one set in the partitioning into two smaller subsets to get a new partitioning. Given this representation, choosing the disassembly operations to be performed and their sequence corresponds to choosing a path on the lattice that begins at the root node (completely assembled state) and ends at any other node on the lattice. The end node of this chosen path represents the final disassembly state (sometimes referred to as disassembly depth), beyond which any remaining sub-assemblies will not be further separated into components.

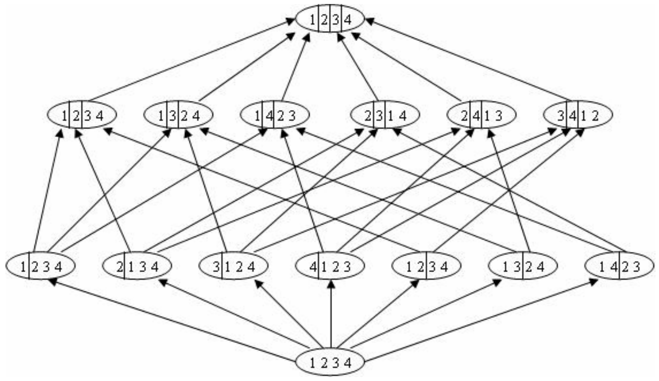

Fig. 1: Partition lattice (π_4) for a four part product.

Any incoming end-of-life product needs to be inspected by the ATF to assess damaged or missing parts, as well as modifications made to the product configuration. Accordingly, the ATF constructs the appropriate partition lattice. For example, if the part labeled ‘3’ in the four part assembly shown above is missing, then a three part partition lattice (for parts 1, 2, and 4) will be constructed; or if instead of the part labeled ‘4’ the product has additional parts ‘5’ and ‘6’, then a five part partition lattice (for parts 1, 2, 3, 5 and 6) will be constructed. Thereafter, the methodology essentially involves adding information to this graphical representation so as to enable selection of the optimal treatment plan (or path in the lattice). The steps involved (as shown in Fig. 2) are explained below:

1. *Determination of possible end-fates and corresponding processing requirements for each possible sub-assembly and component:* Mathematically, for any product assembly with *n* indivisible parts or components, there will be 2^n*-1* possible subsets of parts (not including the null set Φ). If the partial assembly of components defined by a subset of {*1, 2, ..., n*} cannot be realized (e.g., if it contains components that are not connected), then the subset should be marked as “infeasible”. Likewise, a subset is marked “ineligible” if there is a legal requirement to further disassemble the corresponding partial assembly and a single end-fate cannot be defined. Thereafter, for each remaining (feasible and eligible) subsets all alternative end-fates and corresponding processing steps must be determined, such that no regulations (e.g., disposal of hazardous materials) are violated and no further non-destructive disassembly is carried out. This will require complete knowledge of the constituents of each component, prevalent regulations, as well as alternate uses and recycling technologies. It should be noted that such a determination of end-fates would be carried out in advance by component designers and recycling experts, except for cases where unanticipated modifications

or replacement components are present. Nevertheless, the allowable end-fates need to be reconciled to account for local processing capabilities, market for recycled/used parts, etc.

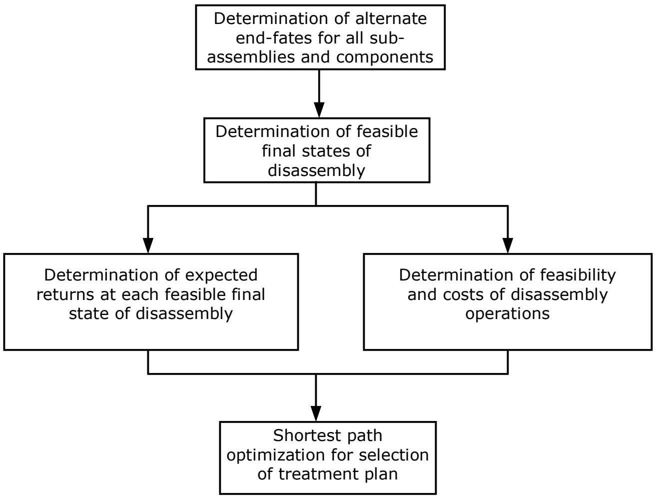

Fig. 2: Tasks in planning end-of-life treatment strategy.

2. *Determination of feasible final states of disassembly:* The next step is to determine feasible final states of disassembly. Each node of the partition lattice defines a state of disassembly, and is composed of a combination of subsets considered in the first step. Thus, any node in the lattice will be a "feasible" final state of disassembly if;
   - No subset in the partitioning for the node is marked as "infeasible" or "ineligible", and
   - There exists at least one combination of available end-fates for the subsets, such that all regulations for minimum recovery of components, minimum recycling of material, and separation and safe handling of hazardous materials, are satisfied.

   Thus, this step requires input about applicable product level regulations and a method to validate the combinations of alternate end-fates against these regulations.
3. *Determination of expected returns at each feasible final state of disassembly:* Once feasible final states of disassembly are determined, one must find out the expected returns if the given node is chosen to be the final state of disassembly. For a given state of disassembly, there may multiple combinations of end-fates of the corresponding subsets that possible. In order to calculate the net expected returns for each combination, the returns from every possible end-fate of its subsets, as decided in step one, should be determined. For this purpose, the costs for processing steps such as degreasing, cleaning, transportation, etc., must be provided by the ATFs. Similarly, expected returns from recycling or refurbishment and costs of disposal must also be calculated. The condition of incoming parts must also be recorded by the ATFs at this stage, since the expected returns may be affected if a part is damaged. Using this information, the combination of end-fates of the subsets, that gives maximum net returns (or minimum net cost) can be calculated for each feasible final state of disassembly and the value will be associated with the respective nodes in the partition lattice.
4. *Determination of feasibility and costs of disassembly operations:* In order to select the optimal path in the partition lattice, one must also determine the feasible edges and the costs associated with them. Each edge in the partition lattice represents a disassembly operation that divides one subset into two smaller subsets. The edge will be regarded as feasible, if the resulting subsets are feasible and if the joint holding the two resulting subsets together is accessible in the original subset. For a feasible operation, the cost of performing the operation needs to be calculated and assigned to the edge. For infeasible operations the corresponding edge is removed from the solution space. It should be noted that consequently, any node that has a sub-assembly that is marked "infeasible" shall automatically be removed from the solution space. This step involves locating joints, determining tools and the method required for disengagement, determining accessibility requirements or obstacles for disengagement, and lastly, the time, effort and costs of disengaging each of the joints.

Information about available tools and capabilities at the ATF, and existing labor rates will also be necessary to accurately determine the feasibility and cost estimates for the disassembly operations.

5. *Optimization for selecting optimal treatment plan:* The final step in the methodology involves optimization to select the optimal treatment plan (i.e., the optimal path and subsequent processing steps corresponding to the final state of disassembly). This essentially involves solving a one-to-many shortest path problem on the partition lattice for paths starting at the root node and ending at any of the feasible final states of disassembly.

The methodology described above affords the flexibility to decide the treatment strategy dynamically taking into account temporal and local considerations, such as prevailing markets, labor costs, facilities available, as well as the condition of incoming products. Obviously, it requires gathering information from various sources across and beyond the enterprise. A Product Lifecycle Management (PLM) system, with its enterprise-wide scope, can provide an ideal framework for automating the collection and organization of this information. Simple algorithms can also be implemented within the PLM system for determining feasibility of the nodes in the graphical representation and the expected returns, once the feasible and eligible subsets and their possible end-fates are determined. The selection of the end-of-life treatment strategy is then reduced to the solution of a shortest path optimization problem. There are various methods discussed in literature for solving such optimization problems.

However, the determination of the feasibility and costs of disassembly operations (step 4) is a non-trivial task. As discussed in section 2, current methods for disassembly planning rely on special representations that record information about joints, disengagement costs, and precedence relationships. These representations have to be generated manually by human experts or using semi-automated methods, such as virtual reality simulations. In the remainder of this paper, we focus on the problem of automating the process of extracting this information from readily available representations of the product, namely CAD models. Specifically, we discuss the algorithms developed for identifying specific types of joints from CAD assembly models and characterizing them with respect to size, location, orientation, etc., so as to enable calculation of accessibility requirements and costs of disengagement.

4. IDENTIFICATION OF JOINTS

In an integrated PLM framework, information about joints will have to be extracted from CAD models, which constitute the primary format for storing product configuration information. Current CAD applications allow the creation of mating constraints and relationships between components in an assembly model. Common practice is to use these constraints to align and position the components with respect to one another. However, information about joints that hold the components in alignment is not explicitly recorded. For example, whether alignment of the holes is achieved by screws or by nuts and bolts is not recorded, nor will the CAD system allow a user to over-constrain a system by specifying alignment on three parallel holes. Efforts are underway to develop schemas to represent assembly information such as connections in CAD [4], and the use of tags to associate such meta-data with the geometry has also been suggested. Nevertheless, deduction of joint information based upon geometry of the mating parts will be useful for detecting designer's intent and thereby assist in recording the joint information. It would be particularly helpful when assembling different configurations of the product, which might include third-party components that are designed to be connected in unintuitive ways. Furthermore, it will help ensure consistency between geometry and stored joint information as the geometry undergoes modifications during the evolution of the design.

There are numerous ways to achieve a joint between two parts. We limit the scope of our research to two of the most commonly used joining methods, namely, threaded fastener joints and snap fits. Often standard joining elements, such as screws or nut and bolts, are not modeled in CAD. Deformation of components, to form snap fits, is also not stored in the model. However, certain geometric features are often present on the modeled components that will betray the existence of such types of joints between them. For example, a single hole on one component being aligned on either side with holes of the same nominal diameter on the other component indicates the likelihood of a pinned joint between the two, whereas an array of holes aligned on one side with a corresponding array of holes on the other component indicates a likely rigid connection using nuts and bolts or screws. Our approach in this research is to determine the required geometric features on the mating components, along their position and orientations relative to each other, to indicate the presence of a particular of type joint. We then implement rule-based, heuristic algorithms to search for these conditions in the CAD assembly models.

We assume that in the CAD assembly model, we have B-Rep representations for each individual component positioned in the assembled configuration in a common coordinate frame. We also assumed that although standard joining elements or deformation may not be modeled, the shape of the parts being joined are modeled completely (i.e., holes to locate the fasteners or engaged geometry of the snap fit is modeled).

4.1 Identification of Threaded Fastener Joints

The two basic types of commonly used threaded fasteners are "nuts & bolts" and "screws". Nuts & bolts require simple aligned through holes on the components being joined. Access from both sides of the component during assembly or disassembly is generally required. A screw passes through a simple hole in one component and fits into a threaded hole in the other. Screws and bolts are variously classified depending upon thread pitch (coarse/fine), hardness grades, type of head (flathead or countersunk, button-head, hexagonal head), type of screw drive (slotted, cross head, hexagonal head, torx head), etc. For the purpose of this research, we only distinguish between "nut & bolt" joints and "screw" joints.

The algorithm to identify threaded fastener connections essentially parses the CAD assembly model to search for mating faces between components. For each pair of mating faces, it searches both the faces for holes (i.e., circular internal loops). If a coaxial pair of holes is detected on the mating surfaces, the holes are marked as a potential location for threaded fastener joint. The program then measures the diameter of the holes, the depth of the holes in each component, and the orientation of the through face at the other end of the hole (if through hole). A "nut & bolt" joint is identified if the coaxial holes are of equal diameter and it is possible to seat a screw-head at the opposite ends of both holes. A "screw" joint is inferred if one hole is slightly larger than the other hole (within a threshold currently defined by a diameter ratio of 1.2), and it is possible to seat a screw-head at the opposite end of the larger hole. The program also infers the exact head location, length of screw/bolt, and orientation of its axis from the geometry of the mating components. We consider that a screw-head can be seated on the end of a hole if;

- the end is open (as opposed to blind), and
- the through face is planar and perpendicular to the hole axis (or conical with standard countersink cone angle and coaxial with hole), and
- the area immediately surrounding the hole (circular area 1.5 times the nominal hole diameter) is not obstructed by another component.

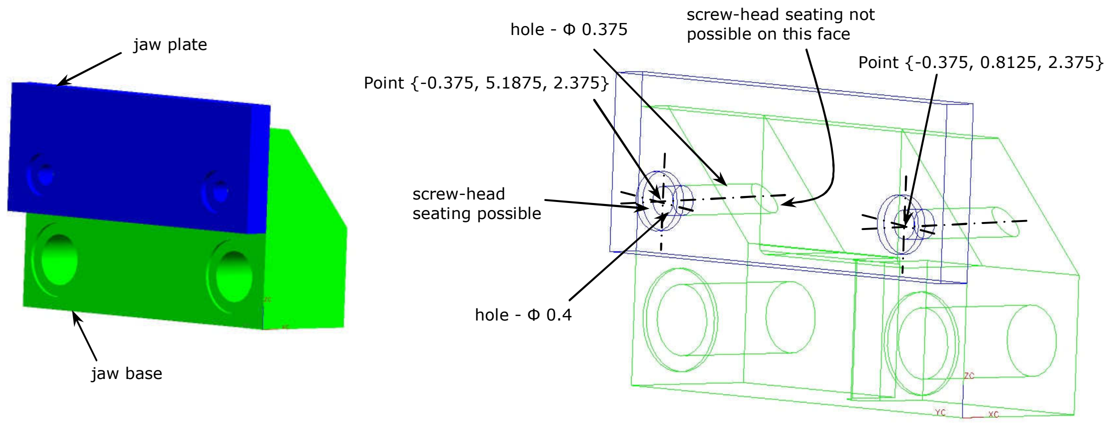

Fig. 3: Example part with "screw" joints.

Fig. 3 shows the fixed jaw assembly of a vise clamp. The assembly consists of only two components, namely, the jaw base and the jaw plate, joined with each other using two screws (which are not modeled in CAD). The figure also shows the "screw" joints identified by the algorithm. The screw head, in both cases, is located on the jaw plate end with the screw axis oriented in the positive Z-axis direction. The screw head cannot be located on the jaw base, since the through face is not perpendicular to the hole axis. The diameter of the screw is correctly determined. The combined length of the coaxial holes is output as the maximum possible length of the screw, based on the assumption that the screw will not emerge out of the threaded hole. Note that while the maximum possible length is calculated for "screw" joints, for "nut & bolt" joints the combined length of the coaxial holes is added to the height of the nut to calculate the minimum required length of the bolt. Other holes (corresponding to guide rails of the vise) are also detected on the mating face of the jaw base, but do not have corresponding coaxial holes on the jaw plate, and are therefore removed from further consideration.

4.2 Identification of Snap Fits

Snap fits are commonly classified into three basic types [15], namely:

1. "Jaw" or "Barbed leg" type fits, which use cantilever deflection for assembly and disassembly.
2. "Cylindrical" snap fits, which employ annular deflection of a cylindrical jaw or lip for assembly and disassembly.
3. "Spherical" or "Ball and Socket" snap fits, which employ deflection of a spherical cup or socket to attach a spherical ball on the mating part.

In addition, various intricate designs have been used to obtain snap fits between components. Although all these joints use similar principles of compliant shapes, the different modified shapes may require fairly different tools and forces to obtain the required deflection to engage and disengage the joint. In this research, we restrict our attention to the basic three types listed above.

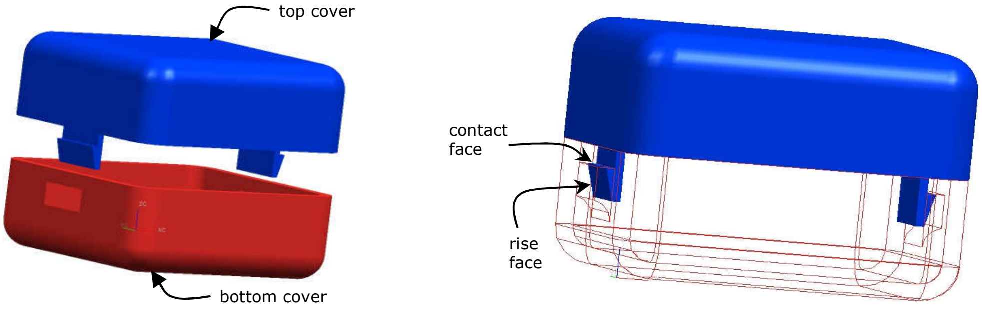

Fig. 4: Example part with "jaw" type snap fit.

4.2.1 "Jaw" Type Snap Fits

Fig. 4 shows the exploded view of a simplified cover assembly (for a fuse box, or battery unit). It has two parts - a bottom cover and a top cover, which is snap fitted into the bottom cover in the assembled condition. The main element of the snap fit geometry, the jaw at the end of the cantilever beam, is made up of a "rise face", which is pushed against the mating part to cause deflection during assembly, and the "contact face", which engages with the mating part in assembled condition.

The algorithm to identify jaw type snap fits parses the assembly model for planar mating surfaces between components. For each planar mating face, the algorithm calculates the normal of each adjoining face. If the normal of the adjoining face makes an obtuse angle with the normal of the original mating face, the algorithm calculates the projection of the adjoining face on the mating face. If the projection of the adjoining face completely covers the mating face, the adjoining face is identified as the rise face. The component is identified as having the jaw for a cantilever jaw type snap fit.

4.2.2 "Cylindrical" Snap Fits

Cylindrical snap fits are similar to jaw type snap fits. A cylindrical protrusion (henceforth, referred to as the shaft) on one component has an annular jaw that engages in a recessed hole on the other part. Thus, the contact face on the shaft is annular and the rise face is conical, such that the projection along the axis covers the annular mating face. Fig. 5(a) shows a hypothetical part with two types of cylindrical snap fits. If the contact faces on the two components are planar, the joint is referred to as "permanent cylindrical snap fit" since it cannot be disengaged by simply pulling the components apart along the axis of the joint. In such cases, the shaft is usually relieved to form multiple prongs, which can be pressed together on the rise faces to disengage the joint. If the contact faces on the components are conical, such that the joint can be disengaged by pulling the components apart, the fit is referred to as a "push-pull type cylindrical snap fit". The shaft may or may not be relieved depending upon the material of the parts. Fig. 5(b) shows contact faces, rise faces, and the disengage direction for both the cylindrical snap fits in the example part.

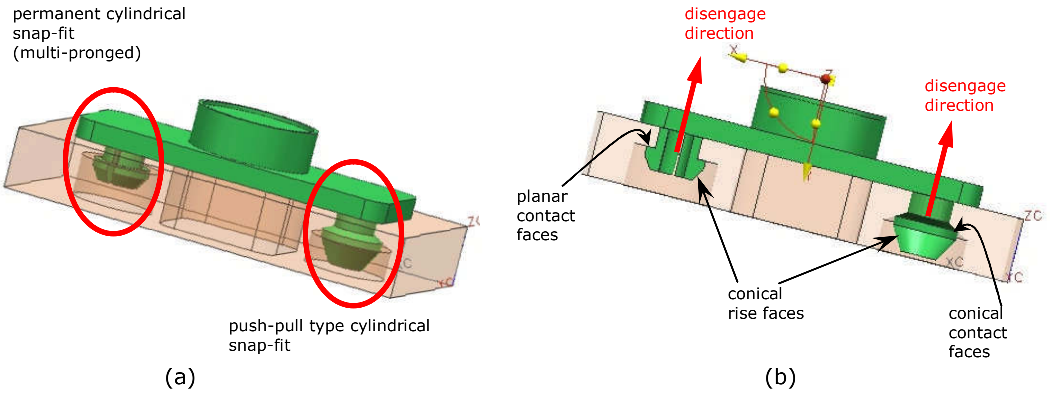

Fig. 5: Example part with "cylindrical" snap fits.

The algorithm to find cylindrical snap fits parses the assembly model to search for planar and conical mating faces with an annular area of contact. The program then establishes which of the components in contact can form the shaft in a cylindrical snap fit by analyzing the geometry of the contact faces. Thereafter, the program searches conical faces on the shaft component. If a conical face is found such that its axis passes through the center of the annular contact area and its projection covers the contact area, it is identified as the rise face. Depending upon whether the contact faces are planar or conical the fit is classified as "permanent" or "push-pull" type. The disengagement direction is opposite to the axis direction of the conical rise face.

4.2.3 "Spherical" or "Ball and Socket" Snap Fits

A "spherical" snap fit is obtained when a protrusion with a spherical tip, called ball, on one component mates with a spherical cavity, called socket, on the other component. Fig. 6 shows an example part with "spherical" snap fits. The socket is often formed by multiple faces in order to flex to allow assembly of the joint as also to allow relative rotation along multiple axes while constraining the translation.

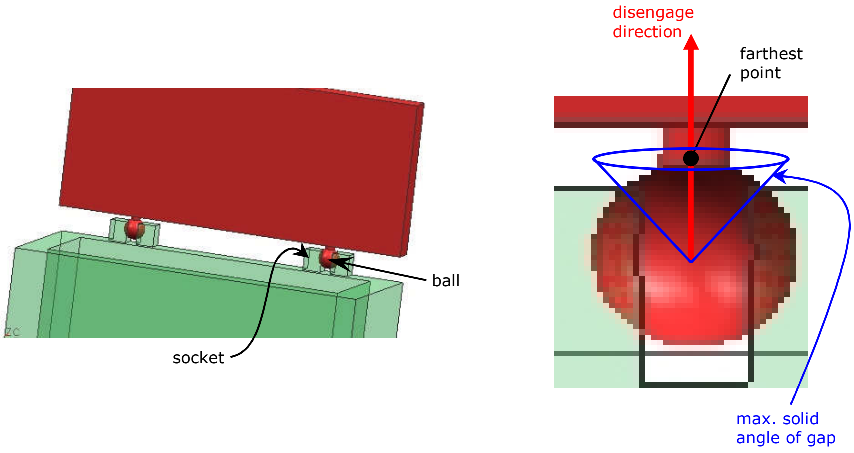

Fig. 6: Example part with spherical snap fits.

The algorithm to find spherical snap fits begins by parsing the assembly model for mating spherical faces. The component containing the ball feature is identified using the material side information and the location and size of the ball is noted. Thereafter, equispaced sample points are created on the surface of the ball and their shortest distance to the faces of the socket component is calculated. If the sample point is covered by a socket face, this distance will be zero. Points that are not covered will have a positive distance from the closest face on the socket. Therefore, the point that has the maximum shortest distance will be at the center of the largest gap in the socket. Thus the direction of disengagement is identified as the vector from the center of the ball toward this maximum distance point. The minimum cone angle of gap can be calculated using the farthest distance, and can be used to calculate the force required for disengagement.

4.3 Implementation

The algorithms discussed above were implemented using C++ and UG/Open API interface. They were tested on the example parts shown in this section. The output from the program for the example part with cylindrical snap fits can be seen in Fig. 7. This output can then be used to populate a tag that stores the joint information in CAD file and further to calculate the accessibility and costs for joint removal.

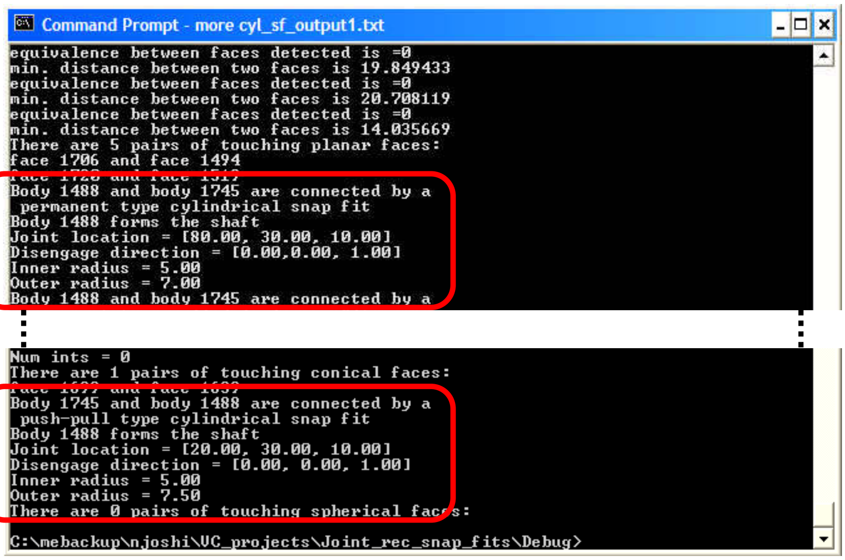

Fig. 7: Program output for part with cylindrical snap fits.

5. CONCLUSIONS AND FUTURE WORK

We have presented a methodology for dynamic, case-by-case selection of the treatment strategy for end-of-life products. The methodology introduces a partition lattice representation to model the selection problem. Although similar to previously used graphical representations, the new representation affords the advantages of being easily automated and able to accommodate different configurations of the product. It also allows temporal, local, and case-based factors to be integrated with known product information.

To enable direct use of CAD models as the source of assembly information, we have developed rule-based heuristics to identify and characterize different types of joints, namely threaded fastener joints and snap fits, in CAD assembly models. Standardized schemas [4] to represent assembly information, such as type of joints, their size, location and orientation will be required to store the information inferred directly in the CAD models. The heuristics developed will also help maintain associations with the feature parameters, such that the joint information can be automatically updated if the geometry of the components is modified.

To calculate the feasibility and cost of removing the joint, one needs to use the joint characteristics inferred to determine the method of disengagement, the tools required and accessibility requirements. This involves determination of a collision free path for the tool to reach the joint, complete the range of motion required to disengage the joint, and return to its original location outside the part along with any joining element, at any given state of disassembly. Clamping and manipulation of parts being separated also needs to be considered. Offline robotic motion planning methods [16], involving use of distance maps, can be used to determine collision free paths for the tool. Exploring these techniques forms the focus of our ongoing and future work.

The methodology presented here, along with the methodology to proactively account for regulatory requirements during the selection of material and processing specifications for components, presented in part 1 of this paper, will improve the ability of OEMs to address the challenges posed by recent Extended Producer Responsibility laws.

6. ACKNOWLEDGMENT

The authors acknowledge the support from the University of Michigan's PLM Alliance. Certain commercial equipment, software, or materials are identified in this paper in order to facilitate understanding. Such identification does not imply recommendation or endorsement by the PLM Alliance.

7. REFERENCES

[1] Bras, B.; Emblemsväg, J.: The use of activity-based costing, uncertainty and disassembly action cards in demanufacture cost assessments, In Proceedings of 1995 ASME Design Engineering Technical Conferences, 82(1), 1995, 285-292.
[2] Chen, R. W.; Navin-Chandra, D.; Prinz, F.: Product design for recyclability: A cost benefit analysis model and its application, In Proceedings of IEEE International Symposium on Electronics and Environment, Virginia, 1993.
[3] Desai, A.; Mital, A.: Evaluation of disassemblability to enable design for disassembly in mass production, International Journal of Industrial Ergonomics, 32, 2003, 265-281.
[4] Fenves, S. J.; Sriram, R. D.; Sudarshan, R.; Wang, F.: A product information modeling framework for Product Lifecycle Management, In Proceedings of PLM Symposium, Bangalore, India, July 2003.
[5] Gungor, A.; Gupta, S.: An evaluation method for disassembly process, Computers and Industrial Engineering, 33(1-2), 1997, 329-332.
[6] Homem de Mello, L. S.; Sanderson, A. C.: A correct and complete algorithm for the generation of mechanical assembly sequences, IEEE Transactions on Robotics and Automation, 7(2), 1991, 228-240.
[7] Hula, A.; Jalali, K.; Hamza, K.; Skerlos, S. J.; Saitou, K.: Multi-criteria decision-making for product disassembly under multiple situations, Environmental Science and Technology, 37(23), 2003, 5303-5313.
[8] Joshi, N.; Dutta, D.: Managing extended producer responsibility using PLM – Part 1: Ensuring compliance in early design stages, Computer-Aided Design & Applications, 4(5), 2007, 719-730.
[9] Lambert, A. J. D.: Linear programming in disassembly/clustering sequence generation, Computers and Industrial Engineering, 36, 1999, 723-738.
[10] Mo, J.; Zhang, Q.; Gadh, R.: Virtual disassembly, International Journal of CAD/CAM, 2(1), 2002, 29-37.
[11] Navin-Chandra, D.: The recovery problem in product design, Journal of Engineering Design, 5(1), 1994, 65-86.
[12] PlanetMath: partition lattices, http://planetmath.org/encyclopedia/PartitionLattice.html .
[13] Santochi, M.; Dini, G.; Failli, F.: Computer aided disassembly planning: State of the art and perspectives, Annals of CIRP, 50(2), 2002, 507-529.
[14] Subramani, A.; Dewhurst, P.: Automatic generation of product disassembly sequences, Annals of CIRP, 40(1), 1991, 115.
[15] Ticona GmbH, Kelsterback, Design calculations for snap fit joints in plastic parts, 2nd edition, 2004.
[16] Ting, Y.; Lei, I.; Jar, H. C.: A path planning algorithm for industrial robots, Computers and Industrial Engineering, 42, 2002, 209-308.