

Life-cycle Clumping of Product Designs for Ownership and Retirement

Matthew D. Marks
Charles F. Eubanks
Kos Ishii

Department of Mechanical Engineering
The Ohio State University
206 W. 18th Ave., Columbus, OH 43210-1107
U.S.A.

ABSTRACT

This paper describes a design strategy for mechanical systems called clumping and its effect on product value during the ownership and retirement phases of its life-cycle. A "clump" is a collection of components and/or subassemblies that share a physical relationship and some common characteristic based upon user intent. Clumping for product ownership values, i.e. reliability and serviceability, aims to increase component accessibility. Clumping for product retirement issues, i.e. reuse, recycling, and disposal, seeks to form aggregations of components for material compatibility, thus decreasing disassembly cost. This paper gives a detailed description of life-cycle ownership and retirement assessment methods and how they can be adapted into an existing computer-based design aid. A refrigerator in-door ice dispenser serves as an illustrative example.

1. INTRODUCTION

Life-cycle design is the process of incorporating various values of a product in the early stages of design. These values include manufacturability, serviceability, recyclability, etc. (Figure 1). A great deal of work has been done in the area of design for assembly (DFA; Boothroyd and Dewhurst, 1983) and design for producibility of components (Priest, 1988). Much of the initial work on life-cycle design issues has focused on pre-ownership values. More recently, attention has focused on design for serviceability (DFS) as a major product ownership value (Makino, et al., 1989; Gershenson and Ishii, 1991). To enhance overall competitiveness and customer satisfaction, designers must address the reliability and serviceability of candidate designs at the layout stage. With the growing concern for the environment, design for recyclability (DFR) has become the newest area of specialization in the DF“X” realm. Although design for recyclability does not address all of the areas of a product’s life-cycle, DFR is a method by which a designer can make immediate and effective decisions about a product’s design. These decisions directly influence how the product will be disposed of at the end of its useful life.

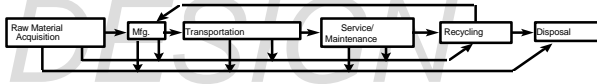


FIGURE 1: PRODUCT LIFE-CYCLE

Life Cycle Assessment (LCA) is a methodology for identifying environmental burdens that arise from a product or process. Groups like the Environmental Protection Agency (EPA, 1993), the Canadian Standards Association (CSA, 1992), and the Society of Environmental Toxicology and Chemistry (SETAC, 1991), have been developing documents that deal with LCA. LCA addresses all life-cycle stages from raw material acquisition to final disposal. The major difficulty with performing a life cycle assessment is the comprehensive nature of the study. Design for Environment (DFE) is a design philosophy and practice whose goal is to minimize the environmental impact of the manufacture, use, and eventual disposal of products without compromising essential product functions or

significantly affecting the life cycle cost in a negative way (Glantschnig and Sekutowski, 1991). Allenby (1991) uses what he calls a Design for Environment Information System (DFEIS) to rank various environmental issues pertaining to each life cycle stage. Glantschnig (Glantschnig and Sekutowski, 1991) reports on DFE at AT&T primarily for waste minimization during manufacturing and end of life considerations.

Our study seeks to bring together different areas of the product life-cycle in order to evaluate designs from various standpoints. Two areas of the life-cycle we are currently addressing are product ownership and product retirement. The product ownership phase is comprised primarily of serviceability and reliability. A product that is reliable and easily serviceable leads to maximum availability and minimum life-cycle service cost, both of which contribute to maximum customer satisfaction. Our initial work focused on automotive hardware (Gershenson and Ishii, 1991; Bryan, et al., 1992), while recent projects address systems such as home appliances. The methodology revolves around Service Mode Analysis (SMA), which focuses on service phenomena in estimating life-cycle service costs. Service modes are the malfunctions of a system as seen through the customer's eyes, extending the concept of Failure Mode Analysis (Dhillon, 1988) to a wider class of service phenomena.

The second area is post-life-cycle intent, which can take on several characteristics. Does the designer intend to have the product discarded into a landfill, or have considerations been made to re-use or recycle part or all of the product? By knowing the post-life-cycle intent of the product, the designer can analyze the product from these standpoints and make iterative changes to improve the design.

A computer tool that helps designers analyze product layouts during the early stages of development would lower the cost of product changes later in the development cycle. This paper focuses on a methodology of grouping components that we call "clumping". Section 2 elaborates on our definition of clumping, while section 3 describes our method to describe system designs. Section 4 gives the details of clumping strategies with examples and section 5 closes the paper with conclusions and future work.

2. ABOUT CLUMPING

We define a "clump" as a collection of components and/or subassemblies that share a physical relationship, and some common characteristic based upon user intent. For recycling, we want all materials and fastening methods in the clump to be compatible with existing reprocessing technologies. An example is a computer housing designed for recycling. Digital Equipment Corporation and GE Plastics have formed a joint venture to buy back Noryl® computer housings. We could define the housing and any attached compatible plastic parts as a clump, and then calculate the disassembly cost to remove the clumps. Figure 2(a) represents a system in which two groups of compatible materials exist. These clumps will not require end-of-life disassembly and can be represented (and analyzed) as single items (Figure 2(b)).

Another characteristic, for both service and recycling, is that the clump must be removable as a unit. To increase accessibility for service, components with known high failure frequencies should be grouped in a clump. This grouping allows for easy removal of the clump from the system and easy removal of the component from the clump. Service cost depends on 1) removal and reinstallation of the clump, and 2) removal and reinstallation of the component (Bryan, et al, 1992.)

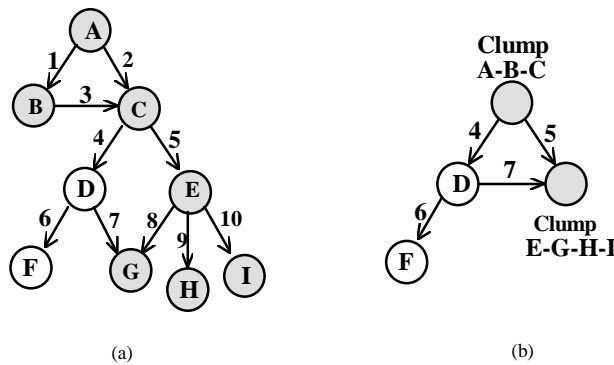


FIGURE 2: ILLUSTRATION OF CLUMPING FOR MATERIAL COMPATIBILITY
 (A) COMPONENTS A,B, AND C ARE COMPATIBLE, BUT DIFFERENT THAN E,G,H AND I.
 (B) ILLUSTRATION OF COMPONENT CONSOLIDATION.

Components can also be grouped for disposal. It is very unlikely that a complex system such as an automobile or refrigerator can be entirely recyclable, but grouping components whose re-use and recycle value is minimal eliminates the disassembly cost associated with the clump. Only the cost of removing the clump from the rest of the system is of interest. Using our layout design description, we intend to show that structural grouping of designs can benefit both serviceability and recyclability.

3. DESCRIBING SYSTEM DESIGNS FOR LIFE-CYCLE EVALUATION

In our computer tools, engineers are asked to describe the design layout using a semantic network representation that we call the "Linker". The computer infers the labor operations and material costs necessary for various service and recycling operations, identifies cost drivers, and indicates areas of improvement.

Consider a mechanical system as an interlinked hierarchically-based network of subassemblies (S1, e.g.) and components (C1, e.g.), as shown in Figure 3. We can envision assembly as creating the relationships, or links, between components, while disassembly consists of breaking component links. Thus, a labor step occurs when a link is either created or broken.

The component relationships in mechanical systems commonly exist in a noun - verb - noun format, such as "flywheel - attaches to - shaft". Likewise, repair operations and labor steps commonly exist in a verb - noun format, such as "remove - flywheel". This notion forms the basis of a semantic network representation. The inference of labor steps becomes a network search that results in a list of links that must be broken in order to disassemble a system, and subsequently reformed to reassemble the system. The nature of the search will depend on whether the disassembly process involves accessing a certain component for service, or if it involves end of life disassembly.

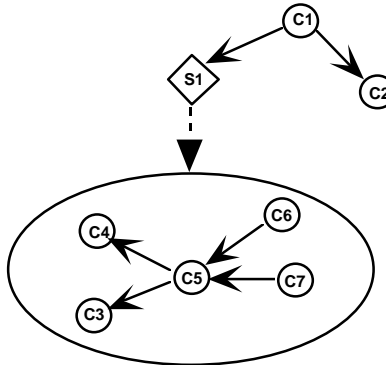


FIGURE 3: HIERARCHICAL NETWORK REPRESENTATION

To accomplish automated reasoning about the design description, we must define both the syntax and the semantics of the network notation (Woods, 1975). We currently use four types of nodes for the design description, defined as follows:

- N1) **COMPONENT:** A design element which cannot be disassembled without permanent damage to the resulting pieces, or loss of intended function following reassembly with the resulting pieces.
- N2) **SUBASSEMBLY:** A design element which can be disassembled into 2 or more other elements and performs its intended function following reassembly with the original elements.
- N3) **FASTENER:** A design element whose intended function or purpose is to maintain an assembled configuration of 2 or more components and/or subassemblies.
- N4) **FASTENING PROCESS:** An action or operation, either physical or chemical in nature, whose function or purpose is to maintain an assembled configuration of 2 or more components and/or subassemblies.

In general, component, subassembly, fastener and process data consist of part or material cost, removal time, installation time, tools and training required to perform the action, the name of the item or process, a user-defined part number or code, and the next higher assembly (if applicable). In addition, fasteners are identified as reusable or non-reusable, with fastener cost added to the repair in the latter case. We currently use five types of links for the design description, defined as follows:

- L1) **COVERS:** No physical connection exists between the two items, but the first item in the link must be removed to access the second. The structural implication is that the cover is attached to or supported by some other item in the system.
- L2) **ATTACHES TO:** This represents a solid connection with no relative motion between the two items during operation. This link is broken by physically removing the first item from the second. The structural implication is that the second item in the link is attached to or supported by some other item in the system.
- L3) **CONNECTS TO:** This represents a solid connection with no relative motion between the two items during operation. This link can be broken (disconnected) without physically removing either of the two items. The structural implication is that the 2 items are attached to or supported by some other item(s) in the system.
- L4) **ENGAGES:** This represents a meshing-type connection with relative motion between the two items during operation. This link can be broken by disengaging either of the two items in the link. The structural implication is that the 2 items are attached to or supported by some other item(s) in the system.
- L5) **SUPPORTS:** This represents a solid connection with no relative motion between the two items during operation. This link is broken by either physically removing the second (supported) item in the link, or by externally supporting the second (supported) item in the link and then physically removing the first (supporting) item. The structural implication is that the supporting item is attached to or supported by some other item in the system.

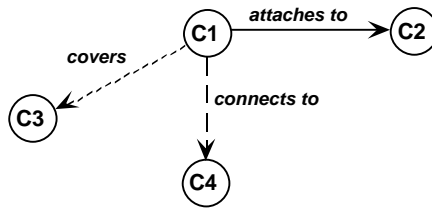


FIGURE 4: SEMANTIC LINKS

Links between design elements represent the same order, or direction, as their semantic representation (Figure 4) ; i.e. "component 1 (C1) attaches to component 2 (C2)", so that C1 sees the link as "outgoing", while C2 sees the link as "incoming". Link direction determines which component receives the required action, if any, and whether one of the components is saved for later examination. If a fastener or fastening process is required to maintain the link, we use a link modifier, called a sublink. Sublink data contains the number of fasteners or process points, clearance around the fastener or process point, tool orientation and, for fasteners, removal and insertion direction (Eubanks, et al, 1992). Figure 5 is a screen dump from our Linker design representation implemented in ToolBook running under Microsoft Windows.

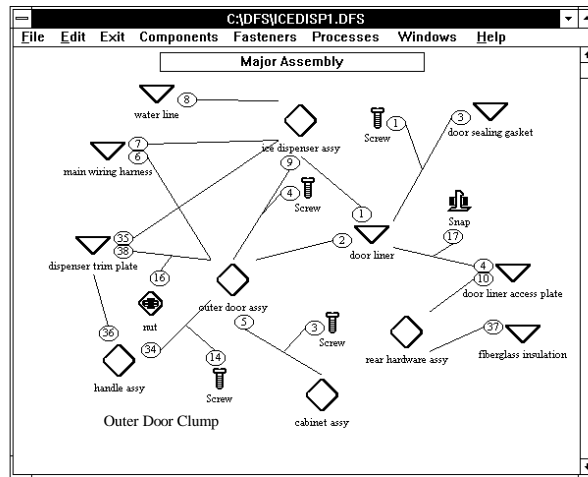


FIGURE 5: LINKER REPRESENTATION FOR A REFRIGERATOR IN-DOOR ICE DISPENSER ASSEMBLY

The "Linker" contains very little spatial information, except for the fact that one component may cover or obscure another. We believe that this level of abstraction is consistent with the type of information available in the preliminary design stage, when components, configurations, degree of modularity, packaging, etc., are not yet finalized.

4. CLUMPING CHARACTERISTICS

4.1 Clumping for Ownership

Serviceability is a primary concern during the ownership of any product. The methodology revolves around Service Mode Analysis (SMA), which focuses on service phenomena in estimating life-cycle service costs (Gershenson and Ishii, 1991). Our current implementation utilizes the linker representation and incorporates the elements of SMA to infer a sequence of labor steps needed to perform each repair operation (Eubanks, et al. 1992). Labor and component costs are calculated for each step and used to calculate life-cycle service costs for repair operations and service phenomena. The labor step cost equation takes the form:

$$LSC = \{[(t_L + p_L) \times c_{LR}] + [c_P + p_P]\} \quad (1)$$

where: t_L = labor time (hours)
 p_L = labor time penalty (hours)
 c_{LR} = labor rate (\$/hour)

c_p = part or material cost (\$)
 p_p = part or material cost penalty (\$)

The labor time is the sum of handling time and either fastening or unfastening time. We add a labor time penalty to account for special tooling requirements, special technician training requirements, fastener clearance and tool orientation. Part replacement adds to the part cost and accesses a penalty based on part availability.

We can now use equation (1) in terms of the life-cycle service cost to yield:

$$LCSC = \sum_{k=1}^n \left[\sum_{j=1}^m \left(f_{R_{j,k}} \sum_{i=1}^l LSC_{i,j} \right) \right] \quad (2)$$

where: $f_{R_{j,k}}$ = frequency of labor operation j associated with service mode phenomenon k
 $LSC_{i,j}$ = labor step cost i associated with labor operation j
 l = number of labor steps associated with labor operation j
 m = number of labor operations associated with service mode phenomenon k
 n = number of service mode phenomena being evaluated (typically 5 to 10)

We compute the labor step costs for a set of repair operations and cost drivers displayed on a summary screen as shown in figure 6. Values displayed are step cost, frequency of occurrence over the entire repair operation set, and total life-cycle cost (frequency x cost).

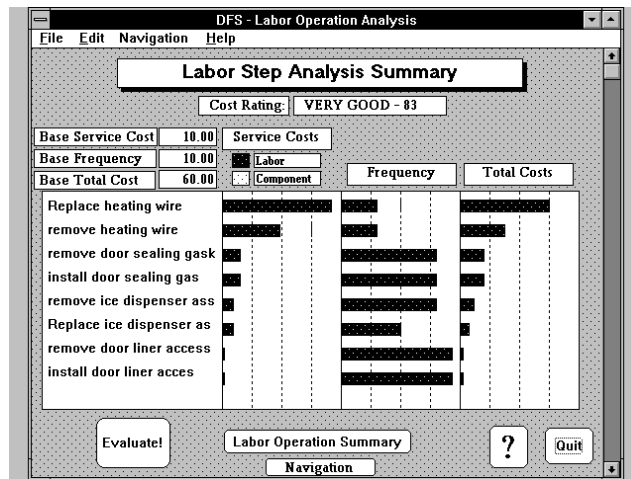


FIGURE 6: LABOR STEP COST SUMMARY

The labor cost computed by this method is directly related to the accessibility of the component in question. Consider the system illustrated in Figure 7(a). Suppose we require access to two sides of G to remove it from the system. Minimal piecewise disassembly would require that components C, D, and H be removed also, for a total of 4 disassembly steps. Since it might not be possible to move the component in question to a geometrically more accessible location, we can explore two clumping options. The number of disassembly step could be reduced to two by 1) clumping obstructing components into an aggregate that is removable as a unit (Figure 7(b)), or 2) combining the component in question into a clump that can be removed as a unit and repositioned to gain access in a previously inaccessible direction (Figure 7(c)).

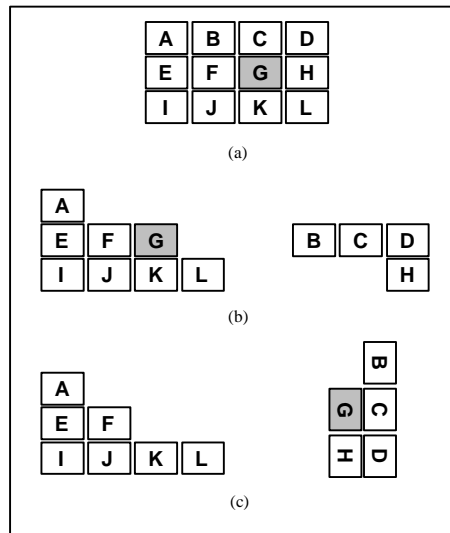


FIGURE 7: ACCESSIBILITY IMPROVEMENT BY CLUMPING.

(A) SYSTEM OF COMPONENTS WITH REQUIREMENT TO REMOVE G; (B) ACCESS GAINED BY REMOVING CLUMP CONSISTING OF B, C, D, AND H; (C) ACCESS GAINED BY REMOVING AND REPOSITIONING CLUMP CONSISTING OF B, C, D, G, AND H.

The labor step inferencing process for a clumped system will change very little from current implementation. The linker representation will identify clumps as single components within the system. If the malfunctioning component is not contained within a clump (Figure 7(b)), the process will proceed as usual. If the malfunctioning component is contained within a clump (Figure 7(c)), the labor step inferencing takes place in two parts: 1) steps necessary to remove the clump (treating it as a component), and 2) step inferencing within the clump itself, starting with the malfunctioning component.

Factors that will influence decisions about clumping for service include: 1) component reliability, since access must be provided for those elements with the lowest reliability, 2) ease of aggregation, which is related to functional communication and structural requirements between elements, and 3) system packaging requirements. We continue to work with our industrial partners to define a set of generic design characteristics that contribute to these factors.

4.2 Clumping for Retirement

Clumping for retirement requires some knowledge of how people deal with the product at the end of its useful life. The product may be clumped for disposal, clumped for re-use, or clumped for recycling. If the product has been designed to be recycled, the clumped components should be manufactured of materials compatible with currently established reprocessing methods. In situations where the average life cycle of a product is more than a few years, it may be difficult to predict government legislation, landfill and raw material availability, and developments in processing technology. In cases like these, the designer needs to base his or her decisions upon current technologies, but should also consider potential developments in technology (see, for example, Nauman and Lynch, *Discover*, 1992) and trends in society.

PC	-											
ABS	2	-										
PPO@/PS	1	0	-									
PPO@/PS/ Nylon	0	0	2	-								
PBT	2	2	0	0	-							
PBT/PC	2	2	0	0	2	-						
ASA	2	0	0	0	1	1	-					
PEI	1	0	0	0	0	0	0	-				
ABS/PC	2	2	0	0	2	2	1	0	-			
Polystyrene	0	0	2	1	0	0	0	0	0	-		
Crystalline Nylon	0	1	1	1	0	0	0	0	0	0	-	
	PC	ABS	PPO@/ PS	PPO@/ PS/ Nylon	PBT	PBT/ PC	ASA	PEI	ABS/ PC	Poly styrene	Cryst alline Nylon	

2 = Compatible
1 = Compatible to a certain level
0 = Not compatible

FIGURE 8: MATERIAL COMPATIBILITY CHART
(GE PLASTICS, *DESIGN FOR RECYCLING*,
DFR ENG. 6M/0392)

Figure 8 shows a material compatibility chart for plastics. On a scale from zero to two, the chart indicates whether two materials can be combined during reprocessing. A "2" indicates that the materials are compatible, a "1" indicates that the materials are compatible to a certain level, and a "0" indicates that the materials are not compatible. This chart, combined with similar information for metals, is part of the material compatibility database used by our program. The data base includes common material names, such as polypropylene or polystyrene, and categorizes the materials as either thermosets or thermoplastics. For metals, the data base includes the name of the material and categorizes the metal as either ferrous or non-ferrous. The information in the material compatibility chart is based upon the judgment of experts. It becomes the responsibility of the user to maintain the data base as new technologies develop. Design engineers typically do not have time to keep track of reprocessing methods for hundreds of materials, however, once the foundation of the matrix has been established, only minor changes should be required.

The designer may wish to clump components that are not compatible, i.e. clumps for service or clumps for disposal. If the post-life cycle intent of the product is for it to be recycled, and materials in the clump are not compatible, then the links between components should be easily broken, i.e. snap fits, press-fits, screws, and screw inserts. Screws should be kept to a minimum and care must be taken not to cross-contaminate the polymers. If the intent of the designer is to clump for disposal, neither the material nor the fastening method is important (aside from being non-hazardous, etc.). The benefit in clumping for disposal is that the clump does not require disassembly; only the boundary links need to be broken.

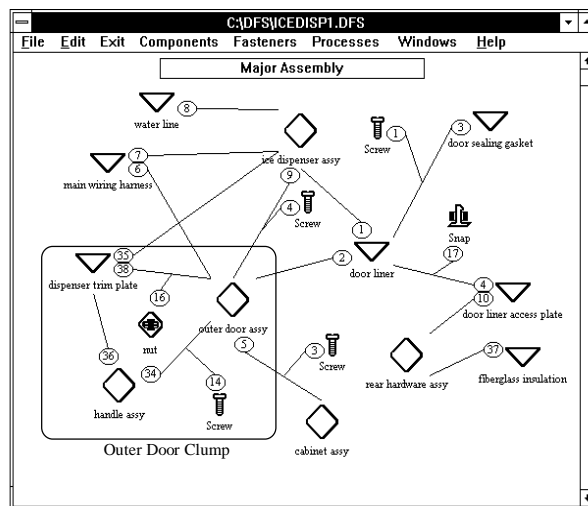


FIGURE 9: CANDIDATE CLUMPING STRATEGY

A boundary link is any physical link (non-"covers" link) connecting the clump to the rest of the system. For example, in figure 9 links 5, 6, 9, and 35 are boundary links. Link 2 is a "covers" link and is not considered a boundary link. The user can visualize the clumped components being consolidated and represented as a single icon. We intend to incorporate component consolidation into the program as an option for the user to better visualize the clumping process (Figure 10).

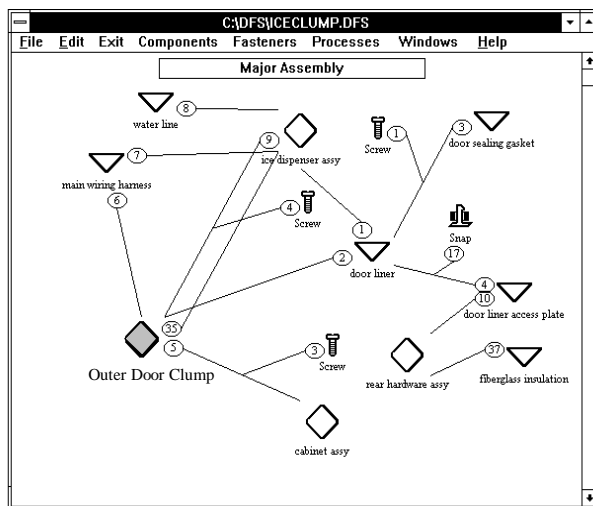


FIGURE 10: CLUMP CONSOLIDATION

After the designer selects the components for clumping, the program analyzes the existing system, individual clumps, and the proposed redesign. The evaluation scheme is based on Design Compatibility Analysis (Ishii, 1991). DCA computes the compatibility index for each link in the system using rules from the knowledge-base. The link index is determined by comparing the components and fasteners used in the link. An example of a rule from the knowledge base follows:

Compatibility-Data:

```

if (material_compatibility is "bad"), and
    (material_type is "polymer"), and
    (fastener is "adhesive")
then (ID=dfrr0002),
    (fastening_compatibility is "bad"),
    (suggestion is "try using snaps,
    separation of dissimilar materials"),
    (comment is "adhesives prevent
    screws, or press-fits")
    
```

The program analyzes individual clumps in the same fashion as the overall system, i.e. all of the "clumped" links are evaluated and a clump index is calculated. The program generates a list, sorted by index, of all non-clumped links and allows the user to select any of these links for clumping (Figure 11). These computer defined, user selected clumps should not require disassembly at the end of the product's useful life, thus reducing system disassembly costs.

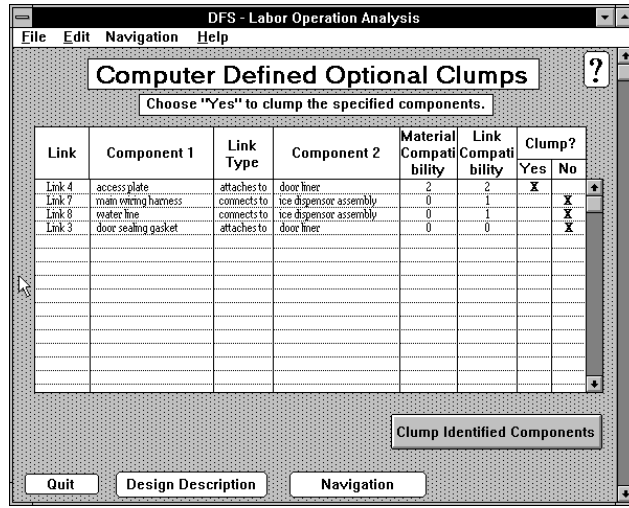


FIGURE 11: OPTIONAL CLUMPING MATRIX

System retirement cost equations are shown below (Note: subscript “c” refers to a clump or a system with clumps):

$$\begin{aligned}
 SRC &= CT_s + D_s + \sum_{i=1}^{dc} (T + L) + \\
 &\sum_{i=1}^{ruc} (M + T - N) + \sum_{i=1}^{rc} (P + T - S)
 \end{aligned}
 \tag{3}$$

$$\begin{aligned}
 SRC_c &= CT_s + D_{sc} + \sum_{i=1}^{dc} (T_c + L_c) + \\
 &\sum_{i=1}^{ruc} (M_c + T_c - N_c) + \sum_{i=1}^{rc} (P_c + T_c - S_c)
 \end{aligned}
 \tag{4}$$

- where:
- SRC, SRC_c = system retirement cost
 - CT_s = collection and transportation (to plant) costs
 - D_s, D_{sc} = system disassembly cost
 - T, T_c = transportation (to landfill, mfg., or OEM) cost
 - L, L_c = landfill cost
 - M, M_c = maintenance / refurbishment costs
 - N, N_c = equivalent cost to manufacture new product
 - P, P_c = reprocessing cost
 - S, S_c = resale value of recycled material
 - dc = number of disposal clumps
 - ruc = number of re-use clumps
 - rc = number of recycling clumps

We focus on system disassembly costs, because the Linker lends itself to a systematic calculation of these costs. Each link in the system has a disassembly time associated with it. Therefore, the total disassembly time for a system with no clumps is calculated by summing the individual disassembly times for each link in the system. If the intent of the designer is to dispose of the entire system, he/she could represent the entire system as a single clump. For a complex design, the system will contain clumps for re-use, recycling and disposal.

In this case, the disassembly calculations include all links except those within a clump. It is important for the designer to make boundary links easily breakable. Figure 12 represents the cost breakdowns for the refrigerator door example shown in Figs. 9 and 10.

The net savings incurred from clumping is shown at the bottom of the figure.

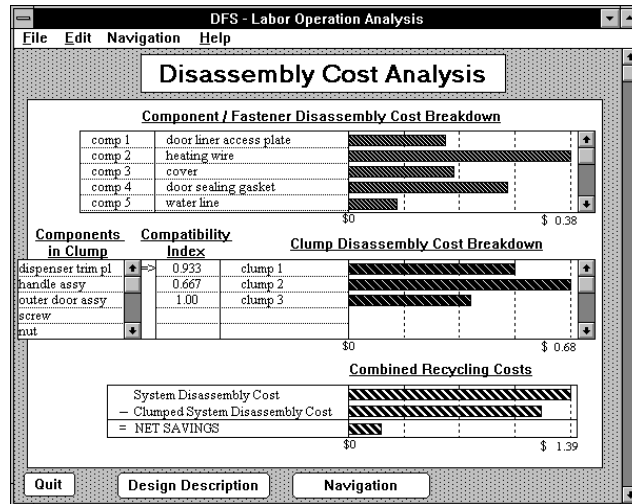


FIGURE 12: RECYCLING COST BREAKDOWN OF ICE DISPENSER ASSEMBLY

4.3 Implementation

The clumping strategies described thus far have led to a program called HyperClumper. The common design representation accommodates the evaluation of several life-cycle issues simultaneously, as illustrated in Figure 13. This clumping simulator allows the user to ask “what if” certain components were grouped for service or recycling.

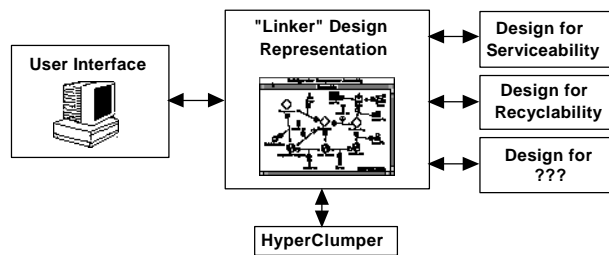


FIGURE 13: USING THE LINKER AS A COMMON DESIGN REPRESENTATION

HyperClumper runs in the Windows environment using Toolbook and Prolog. Our current effort seeks to validate and enhance this program. The validation examples include home appliances and computer peripheral devices.

5. CONCLUSIONS AND FUTURE WORK

This paper presented a design strategy for mechanical systems that we call clumping. Using a semantic net representation called the "Linker", designers can evaluate the effect of clumping on ownership cost (i.e., service and repair cost) and on retirement cost (i.e., reuse, recycling, and disposal costs). There are several future challenges.

1. Trade-off analysis of ownership vs. retirement clumping strategies.

The characteristics which guide clumping for ownership will, in many cases, conflict with those for retirement. The designer will need to know the tradeoffs between these two strategies, thus pointing to a normalized index which provides easy comparisons between the two issues.

2. Obtaining material compatibility data and DFR design guidelines.

General guidelines exist for material mixing during reprocessing, and this type of information is also available for reprocessing metals. However, concise compatibility relationships between various polymer resins remain unavailable. We need to obtain the best available data and use this information to establish more specific guidelines and on-line suggestions regarding materials used in components, fasteners and fastening processes.

3. Integration of clumping evaluation into the current life-cycle design program.

Integration is a key issue in developing a DF“X” environment. By developing methodologies which use a common design

description, we have the potential to combine the evaluation of ownership and retirement. We have started work to integrate clumping routines into the existing serviceability software and modify the labor step inferencing and cost calculation routines to accommodate this analysis. We also need to integrate post manufacturing quality with manufacturability and assembly evaluators.

ACKNOWLEDGMENTS

We deeply appreciate funding and support from the National Science Foundation, General Electric, General Motors, Ford, DEC, and Nissan. The authors express their sincere appreciation for the industrial liaisons and students who have contributed to this work. Special thanks to Sherveen Lotfi for proof reading the manuscript.

REFERENCES

- Allenby, B.R. (1991) "Design for Environment: A Tool Whose Time Has Come." SSA Journal, September, 1991, pp 5-10.
- Boothroyd, G. and Dewhurst, P., (1983), Design for Assembly: a designer's handbook, Boothroyd Dewhurst Inc., Wakerfield, Rhode Island.
- Bryan, C.D., Eubanks, C.F., Ishii, K., (1992), Design for Serviceability Expert System, *Proc. of the 1992 ASME International Computers in Engineering Conference (CIE '92)*, August 1992, San Francisco, CA., ISBN 0-7918-0935-8, Vol. 1, pp. 91-98
- Canadian Standards Association (CSA) (1992) "Environmental Life Cycle Assessment CSA - Z760." July, 1992, Toronto, Ontario.
- Dhillon, B., (1988), Mechanical Reliability: Theory, Models and Applications, National Institute of Aeronautics and Astronautics, Inc., Washington, D.C.
- Discover, (1992), "Mining Plastic", December, 1992, p. 23.
- U.S. Environmental Protection Agency (1993) "Life-Cycle Assessment: Inventory Guidelines and Principles", EPA Report No. EPA/600/R-92/245, Office of Research and Development, Washington, D.C.
- Eubanks, C.F., Bryan, C.D., Ishii, K. (1992), Inference of Service Operations in a Serviceability Design Expert System. *Applications of Artificial Intelligence in Engineering VII*, (D. Grierson, G. Rzevski, R. Adey, eds.), Elsevier Applied Science, New York., ISBN 1-85166-787-3, pp. 373-390
- GE Plastics, (1992), Design for Recycling, The Netherlands, 1992, DFR Eng. 6M/0392, p. 31.
- Gershenson, J. and Ishii, K. (1991) "Life-cycle serviceability design." Proc. of the ASME Design Theory and Methodology Conference, Miami, FL. ASME DE-Vol.31, ISBN 0-7918-0747-9.
- Glantschnig, W.J. & Sekutowski, J.C. (1991) "Design for Environment: Philosophy, Program and Issues." AT&T Bell Laboratories, 1991, Princeton, New Jersey.
- Ishii, K. (1991) Role of computerized compatibility analysis in life-cycle design. *Int. J. of Systems Automation: Research and Applications (SARA)*. Vol.1, pp. 325-345.
- Ishii, K., Eubanks, C.F., Marks, M. (1992) Evaluation Methodology for Post-Manufacturing Issues in Life-cycle Design, *Concurrent Engineering: Research and Applications*. Vol. 1, pp. 61-68.
- Makino, A., Barkan, P., and Pfaff, R., (1989), Design for serviceability, *Proc. of the 1989 ASME Winter Annual Meeting 1989*, San Francisco, CA..
- Priest, J., (1988), Engineering Design for Producibility and Reliability, Marcel Dekker, Inc., New York.
- Society of Environmental Toxicology and Chemistry (SETAC) (1991) "A Technical Framework for Life-Cycle Assessments", January, 1991, Washington, D.C.
- Woods, W. A., (1975), "What's in a Link: Foundations for Semantic Networks," Readings in Knowledge Engineering, (R. Brachman and H. Levesque, eds.), Morgan Kaufman Publishers, Los Altos, CA.